

Styre- og overvåkningssystem for Shell Eco-marathon kjøretøy

Utvikling og test av CAN-bus basert instrumentering og styresystem

Anders Lier Guldahl

Master i teknisk kybernetikk
Oppgaven levert: Juni 2010
Hovedveileder: Jo Arve Alfredsen, ITK

Oppgavetekst

NTNU deltar våren 2010 for tredje gang i Shell Eco-marathon med en bil i Urban Concept-klassen. Hovedformålet med konkurransen er å utvikle en bil som bruker ekstremt lite drivstoff. Bilen som ble utviklet og bygget våren 2008 skal forbedres og målet er å gjøre bilen mer driftsikker og modulbasert. Oppgaven inngår i et samarbeid med øvrige studenter i Shell Eco-marathon-gruppen ved NTNU. Denne delen av prosjektet omfatter utvikling av et nytt styresystem for bilen med fokus på følgende funksjoner:

- Gjøre det mulig for føreren å gi motorpådrag både gjennom gasspedal og cruise control
- Gi fører informasjon gjennom en skjermløsning
- Styre lys, horn og andre elektriske komponenter i bilen
- Overvåke og styre en hydrogen-brenselcelle

Generelt inkluderer oppgaven følgende punkter:

- Utarbeide krav- og designspesifikasjon for ovennevnte styresystem
- Detaljspesifikasjon og implementasjon av dedikert maskin- og programvare
- Test, diskusjon av resultater og dokumentasjon

Oppgaven gitt: 25. januar 2010

Hovedveileder: Jo Arve Alfredsen, ITK

Forord

Jeg vil gjerne benytte anledningen til å takke alle på Eco-Marathon-gruppa for godt samarbeid og innsatsen som alle har lagt ned i prosjektet. Det har vært svært lærerikt, spennende og morsomt å delta på prosjektet.

Jeg ønsker også å takke vår hovedsponsor Det Norske Veritas, samt våre andre sponsorer for å ha bidratt med midler som har gjort det mulig å gjennomføre prosjektet. Andre som også fortjener en takk er Detlef Blankenburg ved IPM for hjelp med bestilling av elektronikk, John Olav Horrigmo ved komponentverkstedet for hjelp og støtte med komponenter, det mekaniske verkstedet for diverse materialer, og ikke minst Jo Arve Alfredsen som har veiledet oppgaven.

Anders Lier Guldahl
Trondheim, 18. juni 2010.

Sammendrag

Shell Eco-marathon er en årlig internasjonal bilprototype-konkurransen mellom studentlag fra høyskoler og universiteter. Våren 2010 skal NTNU delta for 3. gang. Denne oppgaven går ut på å spesifisere krav, designe, lage og teste et komplett instrumenteringssystem for eco-marathon-bilen som deltar i konkurransen i 2010. Modulær oppbygning, samt velprøvd kommunikasjon basert på CAN-bus mellom moduler, danner grunnlaget for et stabilt og enkelt rekonfigurerbart system.

Flere delkomponenter som ble benyttet i bilen foregående år, fungerte ikke ved overtakelse av bilen høsten 2009. På grunn av problemene med tidligere system, kombinert med ønsket om toveis kommunikasjon med alle moduler i bilen, ble det besluttet at et helt nytt system skulle utarbeides. Bilens instrumenteringssystem består i hovedsak av brenselcelleovervåking og styring, kommunikasjon med en el-motordrift, brukergrensesnitt for sjåføren, utvendige lys og signalhorn.

Brenselcellestyringen består av en galvanisk isolert måleløsning for individuelle celledspenningsmålinger, soneinndelt temperaturovervåking, måling av hydrogentrykk og styring av hydrogenventiler, samt et sikkerhetssystem med automatisk avstengning ved gasslekkasje. Forskjellige måter å løse problemet med celledspenningsmålinger er undersøkt. Et fleksibelt og kompakt system er designet og implementert som håndterer funksjonene nevnt ovenfor.

Utvendige lys på bilen med tilhørende driverkretser er basert på energieffektive og kraftige lysdioder som både har mulighet til å lyse opp veien foran bilen, og justeres ned i lysstyrke for å spare strøm. Alt kan enkelt betjenes via et menysystem på en lcd-skjerm i dashbordet. Et utvidbart og fleksibelt menysystem er implementert for å håndtere innstillinger og informasjon fra bilens systemer.

Etttersom systemet er modulbasert, og hver CAN-bus-modul igjen er modulbasert, er det enkelt å rekonfigurere eller endre funksjonaliteten til systemet eller delmoduler. Å hekte på nye moduler for ny funksjonalitet er også veldig enkelt. Systemet kan også benyttes i andre sammenhenger enn bil-instrumentering. Brenselcellemodulen ble for eksempel benyttet på lab hele våren for test og utvikling av styringsalgoritmer for brenselcellestacken. Eksempler på andre bruksområder for deler av systemet kan være hus-automasjon eller styring og overvåking av andre slags lab-oppstillinger.

Selv om konkurranseresultat uteble på grunn av problemer med motorstyringen har elektronikken i styringssystemet stort sett fungert veldig bra under hele prosjektperioden. Det er gjort gode erfaringer med design og bestilling av profesjonelt produserte kretskort, samt innkapsling av elektronikk for å tåle røff håndtering og miljøpåvirkninger. Der problemer har oppstått, spesielt med kretsløsning og de integrerte kretsene benyttet for spenningsmåling på brenselcellestacken, er det nevnt hva som sannsynligvis er skyld i problemene og hva som kan gjøres med det.

Innhold

1	Innledning	1
2	Teori	5
2.1	Modulbasert system	5
2.2	Seriell kommunikasjon	5
2.3	Brenselcellestack	6
2.4	Måling av cellespenninger i stacken	7
2.5	4-20mA strømsløyfe	12
2.6	Ventil og reléstyring	12
2.7	Input/output-expander	13
2.8	Enkodere	14
2.9	LED-driver	16
2.10	Bootloader	17
3	Kravspesifikasjon	19
3.1	Krav og forutsetninger foreslått i prosjektoppgave	19
3.2	Reduksjon av ledningsføring i bilen	19
3.3	Moduler som alenestående enheter	20
3.4	Lys og signalhorn	20
3.5	Sjåfør-grensesnitt	21
3.6	Powerpack	23
3.7	Brenselcellestyring	23
3.8	Kommunikasjon med motorstyring	25
3.9	Opplasting av firmware	26
3.10	Beskyttelse av elektronikken	26
3.11	Diverse ideer	26
4	Design	29
4.1	Systemløsning	29
4.2	Innkapsling av elektronikken	32
4.3	Universalmodul	33
4.4	Frontmodul hardware	35
4.5	Dashbordmodul hardware	40
4.6	Brenselcellemodul hardware	43
4.7	Motormodul hardware	46
4.8	Implementasjon av hardware	49
4.9	Sikkerhetssystem	50
4.10	Viftestyring for brenselcellevifte	50
4.11	Testplan	52
4.12	Sammenkobling av moduler	53
4.13	Bootloader	56

4.14	Generelle softwareteknikker	58
4.15	Software flytskjema	64
4.16	Androidmobil og blåtann	70
5	Resultater	71
5.1	Universal-kretskort	71
5.2	Frontmodul	71
5.3	Dashbordmodul	72
5.4	Brenselcellemodul	73
5.5	Motormodul	75
5.6	Sikkerhetssystem	75
5.7	Sammenkoblet system	75
6	Diskusjon	77
6.1	Brenselcellemodul	77
6.2	Konkurransen	77
7	Konklusjon	81
A	Innhold på CD	85
B	CAN-meldinger	86
C	Skjemategninger	87
C.1	Universal-kretskort	87
C.2	Frontmodul	88
C.3	Brenselcellemodul	89
C.4	Motormodul	90
C.5	Dashbordmodul	91
D	Bootloader	92
E	Bestilling av kretskort	93
F	Brenselcelletilkobling	94

Figurer

1	Flytskjema over arbeidsoppgaver	3
2	SPI-kommunikasjon	6
3	Hydrogenbrenselcelle	7
4	Svitsjet voltmeter	8
5	Motstander og operasjonsforsterkere	9
6	Svitsjet kondensator	10
7	Isolerte analog til digital-konvertere	11
8	Linear Technology batterimonitorkrets	11
9	4-20mA strømsløyfe	12
10	Transistor, relé og diode	13
11	Darlington transistor	13
12	3 bits gray-kodehjul	14
13	Kvadraturenkoderhjul	15
14	To bits gray-kode	15
15	Pulsbreddemodulert signal	16
16	LED-driver prinsipp	17
17	Plassering av moduler og funksjonalitet	30
18	Universal- og spesial-del av moduler	31
19	Elektronikkboks fra Sparkfun	32
20	Universaldel av hver modul	33
21	Universaldel kretskort	34
22	Kobling mellom spesial og universalkort	34
23	Overordnet skjema for frontmodul	35
24	Hvit P4-lysdiode fra Seoul Semiconductor	36
25	Kretsløsning med Zetex lysdiodedriver	37
26	128 stegs absolutt-enkoder for gasspedal	38
27	Oppkobling av absolutt-enkoder	38
28	Oppkobling av signalhorn med relé	39
29	Blåtannmodul montert på frontmodulkretskortet	40
30	Overordnet skjema for spesialdel i dashbordmodulen	41
31	Kvadratur-enkoder til meny-navigering	42
32	Skjema for debounce-kretsløsning	42
33	Skjema for buzzer-høytaler	43
34	Spesialdel i brenselcellemodulen	43
35	Batterimonitorkretser i kaskadekobling	44
36	Digital isolatorkrets	46
37	ULN2003 driverkrets fra Texas Instruments	47
38	Overordnet skjema for spesialdel i motormodulen	48
39	Kretskort dashbord og frontmodul	49
40	Kretskort brenselcellemodul og motormodul	50
41	Hydrogentilførsel og sikkerhetssystem	51

42	Traco DC-DC omformer	52
43	Treplate med ferdige moduler	53
44	Bitvis filtrering av CAN-meldinger i hardware	54
45	Filtrering på enkeltbit	55
46	Filtrering for alle moduler	56
47	Flytskjema for bootloader	58
48	Utgang fra kvadraturenkoder	59
49	Tilstandsmaskin for kvadraturenkoder	60
50	Meny på lcd-skjerm	61
51	Prinsippet for menysystemet	62
52	Flytskjema for software i frontmodul	65
53	Flytskjema for dashbordmodul software	67
54	Flytskjema for brenselcellemodul software	68
55	Flytskjema for motormodul software	69
56	Forkullet motstand	78
57	FLIP3 bootloader	92
58	Kretskort før lodding	93
59	Kobling av flatkabler til brenselcellestack	94
60	Flatkabler mellom brenselcellestack og kretskort	94

Tabeller

1	Generelle krav fra prosjektoppgave	19
2	Individuell testliste	52
3	CAN-bus resetmeldinger	57
4	Eksempel på temperaturomregning	61

Forkortelser

EEPROM	Electrically Erasable Programmable Read-Only Memory
FLIP	FLexible In-system Programmer
IO-linje	Input/Output-linje
LEL	Lower Explosive Limit
NTC	Negative Temperature Coefficient
SPI	Serial Peripheral Interface
SPP	Serial Port Profile
UART	Universal Asynchronous Receive Transmit
UUID	Universally Unique Identifier

1 Innledning

Shell Eco-marathon er en årlig internasjonal bilprototype-konkurranse mellom studentlag fra høyskoler og universiteter. Det arrangeres billøp i hver av de tre verdensdelene på våren. Konkurransen går ut på å kjøre en gitt distanse i løpet av en gitt tid med minst mulig drivstofforbruk. Energikilden kan enten være konvensjonelt drivstoff som bensin, diesel eller naturgass, eller det kan være alternative energikilder som hydrogen, solceller eller bio-brensel. Måleenheten for drivstofforbruk er kilometer kjørt per liter 95 oktan bensin, forbruk og kjørestrekning per liter blir regnet om for de forskjellige energikildene.

Konkurransen har to kategorier; en som heter Prototype, der det meste er lov med tanke på design, sikkerhet og krav til kjøretøyet. Den andre kategorien heter Urban Concept og er den klassen NTNU deltar i. I Urban Concept-klassen stilles det strenge krav til bilen, både funksjonelle krav og sikkerhetskrav. Bilen må i hovedsak ha mange av detaljene en vanlig bil må ha, som lys, fot-betjening av gass og brems og plass til litt bagasje. NTNU deltok første gang i 2008, både da og i 2009 har teamet hatt en tverrfaglig sammensetning. I 2010 arrangeres konkurransen i Lausitz, Tyskland.

Årets team

Årets team består av ti sivilingeniørstudenter som har bilprosjektet som sin avsluttende masteroppgave. Sammen har disse ti en tverrfaglig kompetanse innen produktutvikling og materialer, elkraftteknikk, kjemi og bioteknologi, teknisk kybernetikk og energi, prosess og strømningsteknikk. På grunn av fjorårets erfaringer med mye mediedekning, og det ekstraarbeidet dette medfører, har teamet også fått med seg tre 4.årsstudenter fra medier, kommunikasjon og IT.

Vi samarbeider også med en eksperter i team gruppe og to studenter fra industriell design. Eksperter i team gruppen har ansvaret for å forbedre transporthengeren til bilen og fullføre et grafisk grensesnitt for sanntidsovervåkning av telemetri-data fra bilen. Industriell design har ansvaret for det grafiske designet og lakkering av bilen.

Bilen

Da bilen ble laget i 2008 var fokuset å spare så mye vekt som mulig. Hele karosseriet er laget i karbonfiber og fremdriftssystemet bestod av en elektromotor med tilhørende hydrogen-brenselcelle. Karosseriet har fulgt bilen fra 2008, men fikk mange aerodynamiske forbedringer i 2009. Fremdriftssystemet ble i stor grad byttet ut i 2009-versjonen med ny brenselcelle og spesiallaget elektromotor. Teamets fokus for bilen i 2010 er å redusere bilens prototype-preg, samt forbedre effektiviteten ved å redusere luftmotstand og lage en ny

elektromotor.

Eksempel på prototypedetaljer som trenger forbedring er mangel på dørhengsel og rattoverføringen laget av tauverk. Et annet forbedringspotensiale er driftssikkerheten, bilen ble i utgangspunktet ikke bygget for å tåle veldig mange testkjøringer, framvisninger på messer og lignende. Under løpet i 2009 var det med et nødsrik at bilen klarte å fullføre et godkjent løp.

Nytt instrumenteringssystem

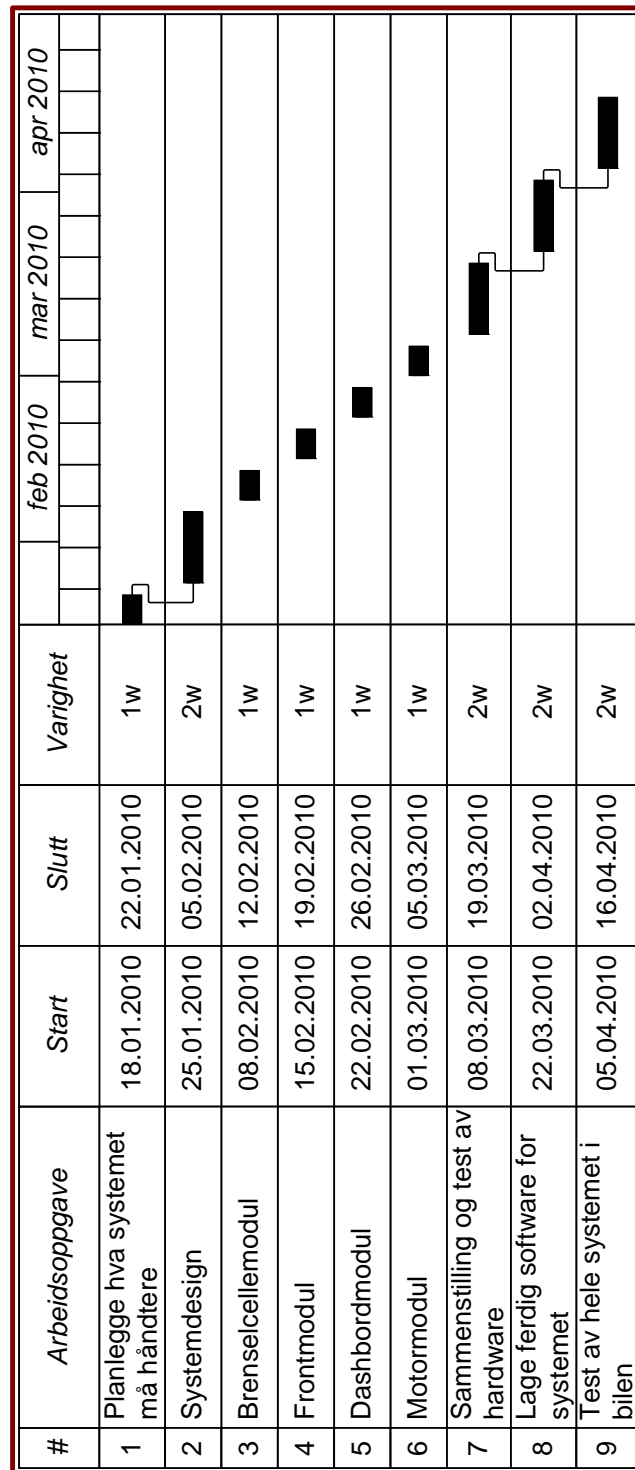
I løpet av høsten 2009 ble en Telemetri-modul for bilen utviklet [1]. For å fullt ut kunne utnytte potensialet til telemetri-modulen er det ønskelig med elektronikk i bilen som er nettverksbasert. Hver elektronikkmodul, enten det er lysstyring, dashbord-indikatorer eller brenselcellestyring må kunne sende og motta data til andre enheter i bilen. I løpet av prosjektoppgaven høsten 2009 ble det bestemt at det nye systemet skulle bli bygget opp rundt samme prinsipper som gjelder for vanlige biler. En nettverksbuss mellom elektronikkmoduler i bilen gjør at alle modulene kan kommunisere med hverandre. Dette gjør systemet enkelt å utvide, og så godt som alt av informasjon er tilgjengelig for alle moduler i bilen. Dermed kan veldig mange endringer og utvidelser utføres ved å kun endre software, eller eventuelt hekte flere moduler på nettverket.

Elektronikk og instrumentering i bilen omfatter alt fra sjåførgrensesnitt og styring av lys og signalhorn, til måling av parametere og styring av en hydrogendrevet brenselcelle. For å gjøre designjobben mindre tidkrevende er mange av designprinsippene fra prosjektoppgaven benyttet videre. For årets bil er det planlagt å ha mulighet til å benytte batteridrift istedenfor brenselcelledrift slik at kjøring og framvisning av bilen utenom konkurransen kan gjøres fra et oppladbart batteri. Instrumenteringssystemet må derfor lett kunne omstilles til batteridrift uten modifikasjoner.

Arbeidsoppgaver

På starten av semesteret ble en arbeidsplan for design og implementasjon av instrumenteringssystemet satt opp. Etersom konkurransen er i starten av mai var det planlagt å bli ferdig i midten av april for å ha en buffer på noen uker. Flytskjema for arbeidet med instrumenteringssystemet er vist i Figur 1.

Alle arbeidsoppgaver fram til det siste punktet i Gantt-diagrammet kan utføres uten at bilen er ferdigbygget. Brenselcellemodulen lages først slik at den kan testes på lab sammen med brenselcellestacken så tidlig som mulig. Ellers kan mange av modulene både designes og produseres i parallell.



Figur 1: Flytskjema over arbeidsoppgaver og planlagt tidsbruk. Ved å planlegge ferdigstilling av bilen til midten av april er det fortsatt en liten buffer igjen til konkurransen i mai.

2 Teori

2.1 Modulbasert system

Et modulbasert system kjennetegnes ved at det er satt sammen av individuelle moduler som sammen utfører en eller flere oppgaver. Hver modul kan utvikles, lages og testes individuelt og samtidig, for så å monteres i det endelige systemet [2]. Moduler kan også byttes ut for å endre funksjonalitet og ytelse. En kan både snakke om modulbasert software og hardware. Eksempel på modulbasert hardware kan være elektronikken i en datamaskin, der individuelle komponenter ofte kan byttes uavhengig av hverandre. For software vil et operativsystem som kan kjøre diverse programmer være et eksempel. Modulbasert design gir også fordeler knyttet til standardisering og volumproduksjon, men kan føre til ytelsestap på grunn av lite optimaliserte grensesnitt mellom moduler. Et eksempel kan være båndbreddebegrensning ved bus-kommunikasjon i datamaskiner, eller tungvinte og lite optimaliserte interface mellom software-moduler i programvare.

2.2 Seriell kommunikasjon

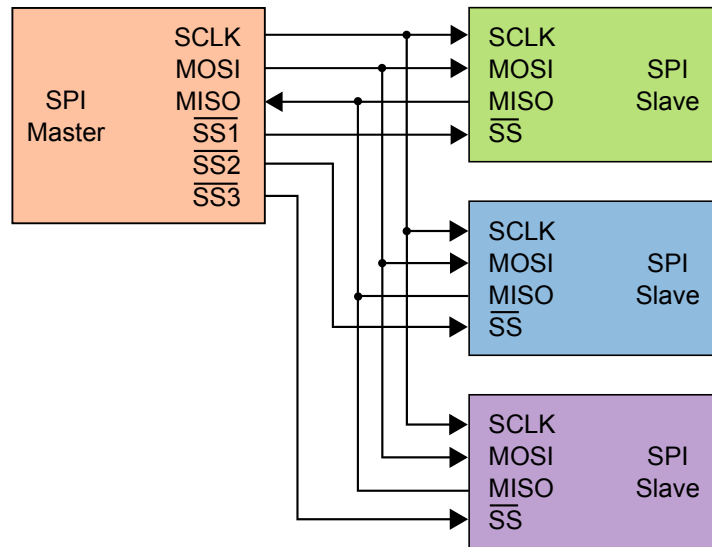
I et instrumenteringssystem med flere moduler, mikrokontrollere og annet hardware er det nødvendig med kommunikasjon mellom de forskjellige modulene. Slik kommunikasjon kan i hovedsak deles inn i parallell kommunikasjon der flere bit blir overført i parallell på flere linjer i bussen. Og seriell kommunikasjon, der hver bit blir overført sekvensielt. Seriell kommunikasjon har den fordelen at færre linjer blir benyttet.

Serial Peripheral Interface

SPI (Serial Peripheral Interface) er en type synkron seriell bus som benytter 4 eller flere linjer til kommunikasjon [3]. En for sending av data, en for mottak av data, en for klokke og en chipselect-linje per enhet for å velge hvilken brikke på bussen som skal være aktiv deltager i kommunikasjonen. I Figur 2 er det gjengitt kommunikasjon mellom en master og tre slaver med tre chipselect-linjer.

RS-232

UART (Universal Asynchronous Receive Transmit) er en fellesbetegnelse på seriell kommunikasjon der mottager og sender ikke utveksler klokkesignal slik som SPI. Det er en form for seriell kommunikasjon der klokken hos sender og mottaker er separat, dette kalles asynkron



Figur 2: SPI-kommunikasjon mellom en mikrokontroller og tre slave-enheter. (Creative Commons)

kommunikasjon. RS-232 standarden definerer elektriske egenskaper, mekaniske tilkoblinger og lignende til en slik asynkron seriell kommunikasjon [3].

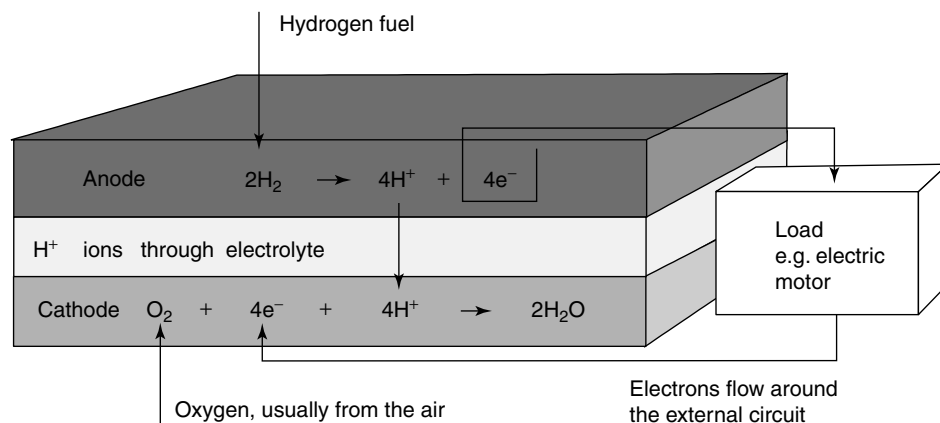
CAN-bus

CAN-bus er en seriell multimaster bus, det vil si at enhver node tilkoblet bussen kan initiere kommunikasjon på bussen. CAN-bus er opprinnelig laget for kommunikasjon i biler og er basert på meldings-id'er framfor adressering av bestemte mottakere. Bussen benytter bitvis arbitrering, som har den effekten at meldinger med lav id får høyere prioritet enn meldinger med høyere id. Bussen støtter hastigheter opp til 1 Mbit/s og benytter to ledninger med differensiell overføring som gir høy støyimmunitet [3].

2.3 Brenselcellestack

Den hydrogenbaserte brenselcellestacken i bilen er en såkalt PEM(Proton Exchange Membrane)-fuelcell som kjennetegnes ved lav arbeidstemperatur (50-100°C). Hydrogenatomer består av et proton og et elektron. Hydrogenbrenselceller fungerer ved katalytisk oksidasjon av hydrogen. Elektronene blir ledet gjennom en elektrisk last og protonene vandrer gjennom proton-bytte-membranen [4]. Oppbygningen av en slik brenselcelle er vist i Figur 3.

Spenningspotensialet over en celle varierer med strøm og temperatur. Spenningen ligger på ca. 1.2V ved liten strømstyrke, og kan synke ned til under 0.5V ved høy last. For å oppnå høyere spenninger kan enkeltcellene seriekobles akkurat som batterier. En brenselcellestack består av mange enkeltceller i serie. Etter at en viss mengde hydrogen har blitt omsatt i brenselcellen vil det bli en opphopning av forurensninger på membranen, dette løses ved å “skylle” stacken med en liten mengde hydrogen. I praksis åpnes en ventil til friluft og en liten mengde hydrogen strømmer gjennom stacken og tar med seg urenheter. Denne prosedyren kalles å purge stacken og må gjøres med jevne mellomrom.



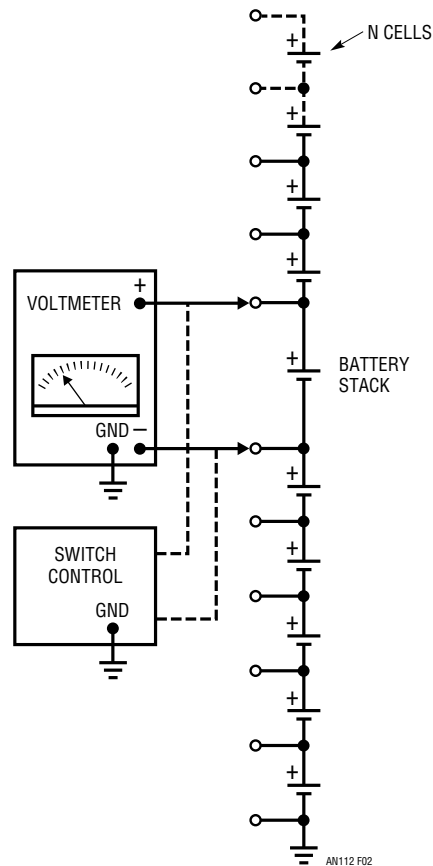
Figur 3: Elektrodereaksjoner og ladningsflyt i en brenselcelle. [4]

2.4 Måling av celledspenninger i stacken

Brenselcellestacken i bilen består av 46 celledspenninger. Dette kan sammenlignes med et batteri bestående av 46 individuelle, seriekoblede celler. Å måle individuelle celledspenninger i et batteri med høy total spenning er et komplisert problem, se [5]. Problemet kalles gjerne “The battery stack problem”. Det finnes flere mulige løsninger på problemet, men noen er vesentlig mer kompliserte og kostbare enn andre.

Voltmeter

Å måle en enkelt celledspenning er enkelt, en kan koble til et voltmeter og lese av spenningen. Voltmeteret kan være batteridrevet, og bryr seg ikke om hvor i stacken det kobles til, da voltmeteret ikke vil “se” hele stack-spenningen. For å bruke denne metoden på hele stacken vil en behøve et svitsjeoppsett som kan koble multimeteret på de forskjellige cellene, uten å kortslutte celler (se Figur 4). Et slikt oppsett for mange celler blir fort veldig komplisert og er derfor sjeldent brukt i praksis [5].



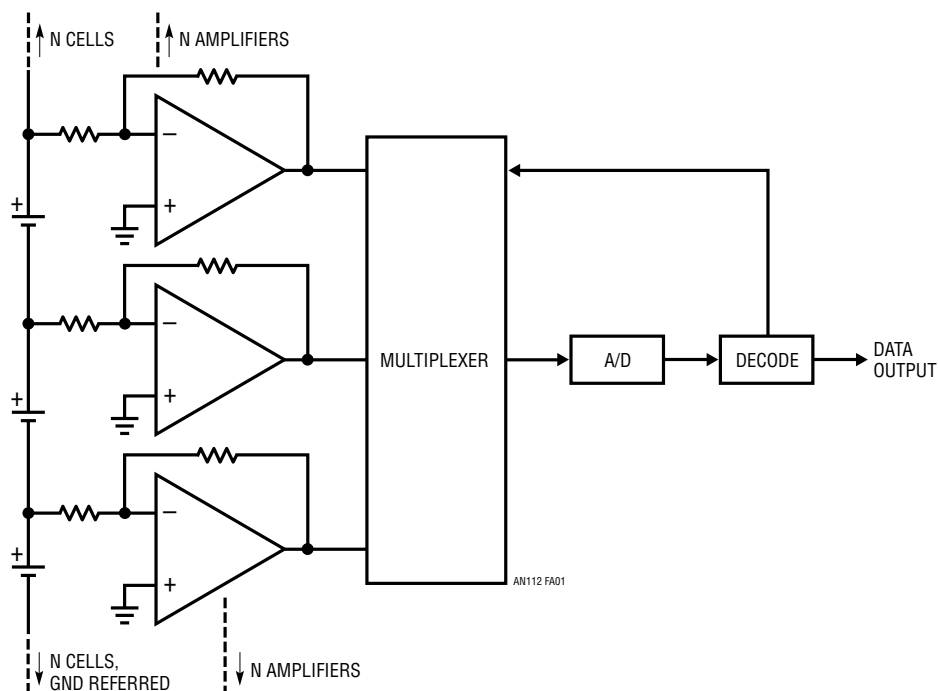
Figur 4: Et voltmeter måler en og en celledspenning og svitsjes mellom enkeltceller. Voltmeteret vil aldri se hele stack-spenningen. (Linear Technology)

Inverterende forsterker

En løsning som unngår problemet med høye spenninger vil for eksempel være en rekke inverterende forsterkerkoblinger med operasjonsforsterkere (se Figur 5). Her vil strømmen i motstandene som kobles til stacken være den målbare variabelen. Problemet med denne metoden er at nøyaktigheten som kreves for motstandene er urealistisk og en vil tappe strøm fra cellene [5].

Svitsjet kondensator

En annen løsning går ut på å sample celledspenninger ved hjelp av en kondensator, som så svitsjes over til en analog til digital konverter (Figur 6). AD-konverteren vil dermed kun se hver enkelt celledspenning uten å måtte takle den høye common-mode-spenningen. Problemet med denne løsningen er at svitsjene fortsatt må tåle høye spenninger, og lekkasje



Figur 5: En inverterende forsterkerkobling på hver celle gjør om høye spenninger til små strømmer som måles. Problemer oppstår som følge av urealistiske krav til nøyaktige motstandsverdier og at strøm trekkes fra stacken. (Linear Technology)

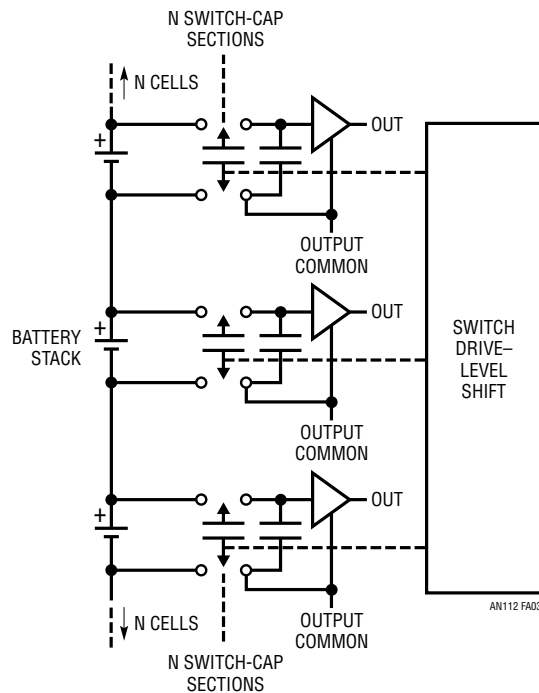
i svitsjene vil gi feil i målingene.

Isolerte AD-konvertere

Ved å koble en analog til digital konverter til hver celle, med isolert dataoverføring og spenningsforsyning til AD-konverteren vil cellespenningene kunne måles (Figur 7). Løsningen kan fungerer bra, og den kan forenkles ved å la en AD-konverter måle flere enn én cellespenning dersom cellespenningene er lave [5]. Men metoden er kostbar og komplisert fordi mange isolerte spenningsforsyninger og datalinjer må implementeres. Det kreves også en rekke analog til digital-konvertere.

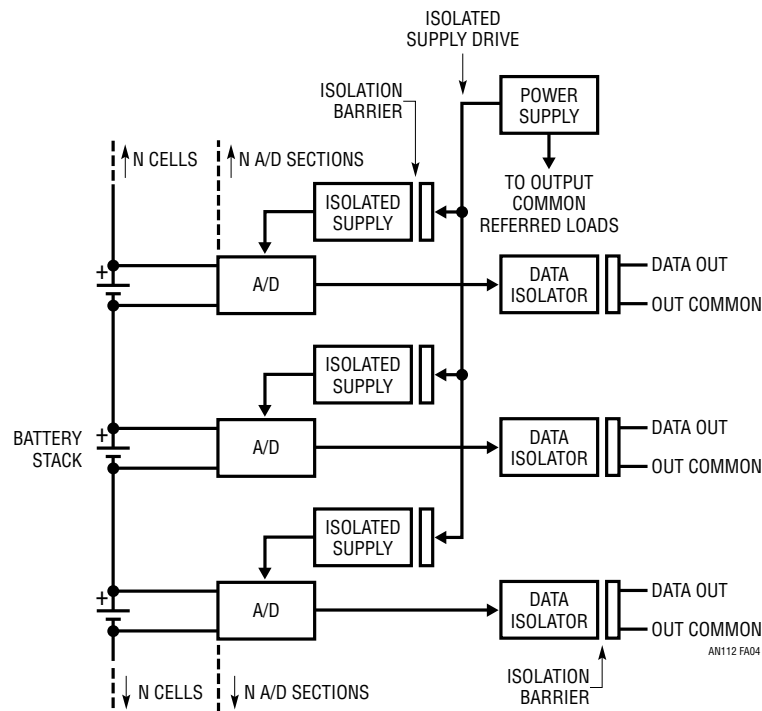
Multicelle batterimonitorkrets

En forenklet versjon av løsningen med isolerte AD-konvertere med intern spenningsforsyning er tilgjengelig i integrerte kretser. Et eksempel på en slik krets er Linear Technology sin LTC6802-1 [6]. LTC6802-1 er egentlig laget for å overvåke og balansere litium-ione battericeller. En krets kan håndtere opptil 12 celler i serie, og hver cellespenning kan være

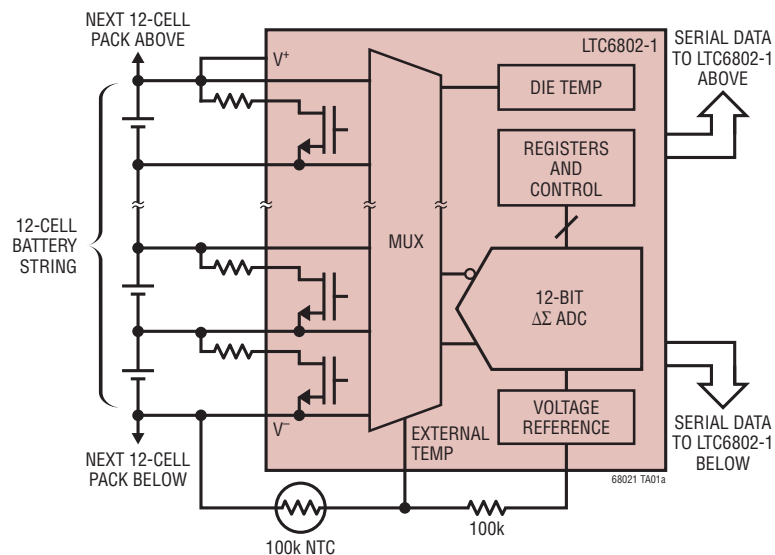


Figur 6: En svitsjet kondensator samler hver cellespenning separat og overfører den til AD-konverteren. Unøyaktighet på grunn av lekkasjestrøm i svitsjene, samt at svitsjene må tåle høye spenninger er problemer med denne metoden. (Linear Technology)

opptil 5V. Det er også mulighet for å koble til temperatursensorer i form av NTC-motstander (Negative Temperature Coefficient). Kretsene kan også kaskadekobles for batteristacker med flere enn 12 celler i serie. Hver krets består blant annet av en multiplexer, en 12 bit AD-konverter, intern spenningsforsyning og et SPI-interface som enten kan kobles til en nabo-krets eller en mikrokontroller. Figur 8 viser den interne oppbyggingen av en slik krets.



Figur 7: En rekke isolerte AD-konvertere måler hver sin del av stacken. Kostbar metode med mange komponenter. (Linear Technology)

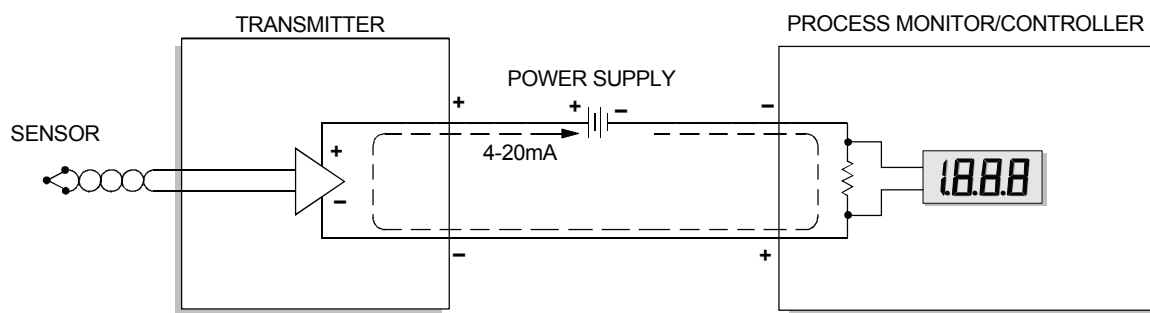


Figur 8: Intern oppbygging av en dedikert batterimonitorkrets. En LTC6802-1 kan håndtere 12 celler, har innebygget strømforsyning og kan kaskadekobles for større batteristacker. (Linear Technology)

2.5 4-20mA strømsløyfe

En 4-20mA strømsløyfe er en vanlig måte å sende data fra sensorer til prosessmonitorutstyr. Sensoren kan være alt fra temperatur, trykk og fuktighetsmålere til mer avanserte strømningsmålere og lignende. Prinsippet baserer seg på å bruke strøm istedenfor spenning som informasjonsbærer. Når sensorsignalet skal gå langt i en ledning vil det nødvendigvis bli et spenningsfall i ledningen. Dersom sensorens signal var en spenning måtte mottakerutstyret hatt veldig høy inngangsimpedans for å få nøyaktig måling. Dette ville igjen gitt problemer med støy på målingene [7].

Ved å sende en strøm over lange avstander vil strømmen kunne holdes konstant selv om det er spenningsfall over lederne. Støyproblemet blir også vesentlig mindre fordi det skal mye til å påvirke en høy strøm som går i ledningen. Et eksempel på en 4-20mA strømsløyfe med en temperatursensor er gitt i Figur 9. Ved å la 4mA bety 0 som målesignal vil en også kunne oppdage brudd i ledningen. Siden det alltid går minst 4mA i sløyfen kan en også la sensoren drives av målestrømmen i sløyfen. Dette gir enkel tilkobling med kun to ledere.



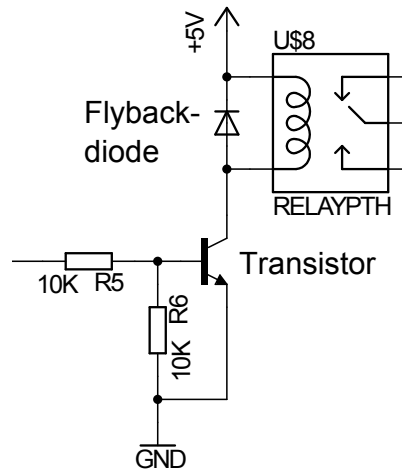
Figur 9: Illustrasjon av en enkel 4-20mA målesløyfe. For å måle strømmen i sløyfen kan en sette inn en enkel motstand i måleapparatet og måle spenningen over denne. (Murata)

2.6 Ventil og reléstyring

I forbindelse med hydrogentilførselen og strømforsyning til motor og hornet på bilen er det nødvendig med hydrogenventiler og elektriske brytere (relé). Dette er induktive laster og det betyr at de vil prøve å motvirke endringer i strømmen som går igjennom. Ventiler og releer fungerer ved å la en liten strøm i en spole lage et magnetfelt som igjen kan vippe over en elektrisk bryter eller en ventil. Styrestrømmen i spolen er vanligvis mer enn en enkel mikrokontroller kan levere, og egne transistorer benyttes gjerne som brytere for spolen i releet, se Figur 10.

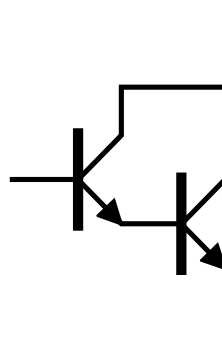
En flyback-diode er også nødvendig for å beskytte transistoren mot høy spenning som oppstår i det transistoren skrur av strømmen. Strømmen i spolen setter opp et magnetfelt

som vil motvirke endring i strøm i spolen. Dette fører til at spenningen over spolen stiger i det strømmen blir brutt. Dioden vil begynne å lede i det spenningen er høy nok, og dermed lede strømmen tilbake og gjennom spolen igjen.



Figur 10: En transistor benyttet som bryter for releet. Dioden beskytter transistoren ved å lede strømmen i spolen tilbake til spolen i det transistoren slår seg av.

Istedenfor en enkelt transistor benyttes ofte to bipolare transistorer koblet opp som vist i Figur 11. Denne transistorkoblingen kalles darlington-kobling og gir mer strømforsterkning enn en enkelt bipolar transistor.



Figur 11: To bipolare NPN transistorer koblet opp som vist på figuren kalles en darlington-kobling. Høyere gain enn en enkelt bipolar transistor oppnås med denne koblingen.

2.7 Input/output-expander

En io-expander, eller input/output-utvider har som oppgave å gi en mikrokontroller flere inngangs- og utgangs-pinner enn den opprinnelig har. io-expanderen er en ekstern krets

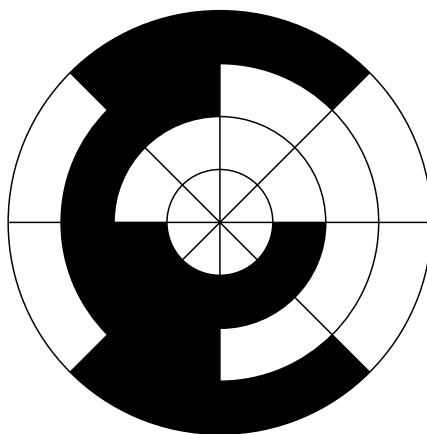
som vanligvis benytter seriell kommunikasjon til mikrokontrolleren, for eksempel SPI. En io-expander har ofte mulighet til å gi et interrupt-signal til mikrokontrolleren dersom en eller flere pinner endrer tilstand. Dermed kan mikrokontrolleren benytte kun en interrupt-pinne for mange innganger. Ofte kan eksterne io-expandere også levere mer strøm enn mikrokontrolleren kan. Mange io-expandere har mulighet for å lagre tilstanden som trigget interruptet, dette er nyttig dersom inngangene endrer før mikrokontrolleren rekker å håndtere interruptet.

2.8 Enkodere

En rotasjons-enkoder er en elektro-mekanisk innretning som benyttes til å måle vinkelen til en roterende aksel. Utgangen kan være et analogt eller digitalt målesignal. Enkodere kan brukes i alle sammenhenger der nøyaktig posisjonsmåling til en aksel er nødvendig.

Absolutt-enkoder

Absolutt-enkoderen har en utgang som alltid vil gi absolutt posisjon til akselen innenfor en hel omdreining. Oppløsningen kan være alt fra noen få skritt til flere hundre per omdreining. Gray-kode benyttes ofte for å lese av vinkelen og kode det digitale utgangssignalet. Dersom vinkelen endrer seg med ett steg vil utgangssignalet i gray-kode alltid endre seg med kun én bit-verdi, se Figur 12 for 3 bit gray-kodehjul. En enkoder som består av mekaniske brytere vil med gray-kode kun endre én av bryternes posisjon ved en vinkelendring på ett steg. Dermed vil det aldri oppstå transiente situasjoner hvor to brytere som skal endre tilstand samtidig ikke gjør det helt likt.

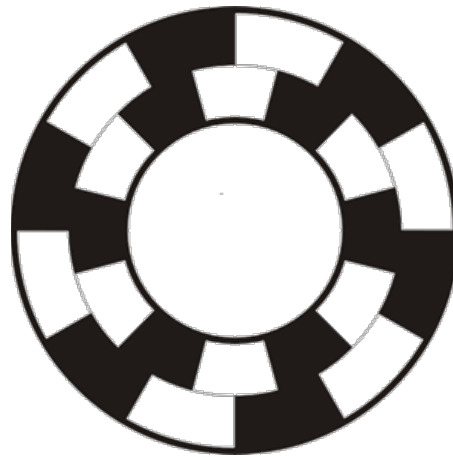


Figur 12: Kodehjul med 3 bit gray-kode. Kan benyttes som absolutt-enkoder med 8 trinn per omdreining.

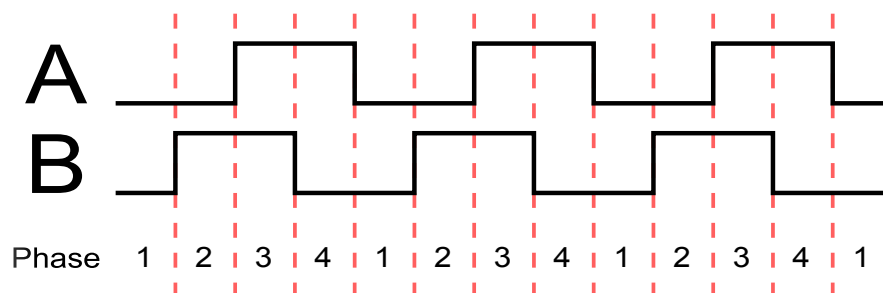
Inkrementell enkoder

En inkrementell enkoder baserer seg på å kun gi ut informasjon om at vinkelen er endret, uten å gi den absolutte posisjonen. En kvadratur-enkoder gir ut en to-bits gray-kode, utgangen er ikke knyttet til en bestemt vinkel, men rotasjonsretningen kan bestemmes. Et eksempel på et kodehjul for en inkrementell enkoder er gitt i Figur 13. En vanligere løsning er enkoderhjul med enkeltspalter og istedet bruke to sensorer med en halv spalte offset. Resultatet blir uansett et utgangssignal som gitt i Figur 14.

Inkrementelle enkodere må ha elektronikk som holder orden på utgangssignalet ved å telle pulser og dermed holde orden på vinkelen. Slik elektronikk kan være batteridrevet for å kunne benyttes i samme applikasjoner som en absolutt-enkoder.



Figur 13: Kodehjul for en kvadraturenkoder, kan sees på som to bits gray-kode.

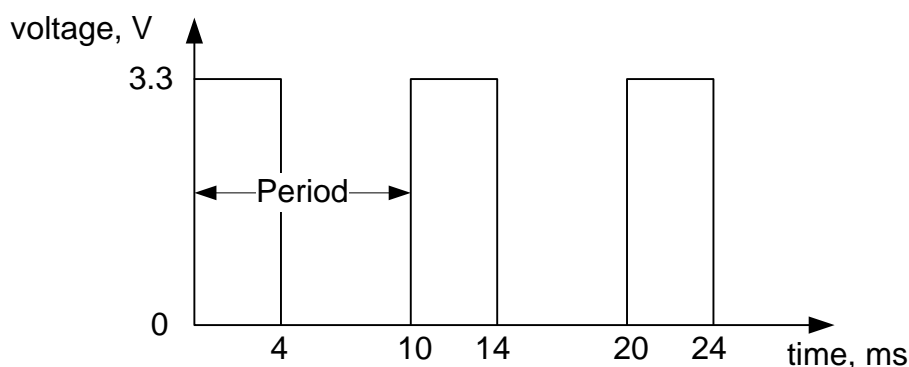


Figur 14: Utgangen fra en inkrementell enkoder. Faseforskjellen mellom A og B bestemmer retningen og antallet pulser bestemmer rotasjonsvinkelen. Signalet kan også tolkes som en to-bits gray-kode.

2.9 LED-driver

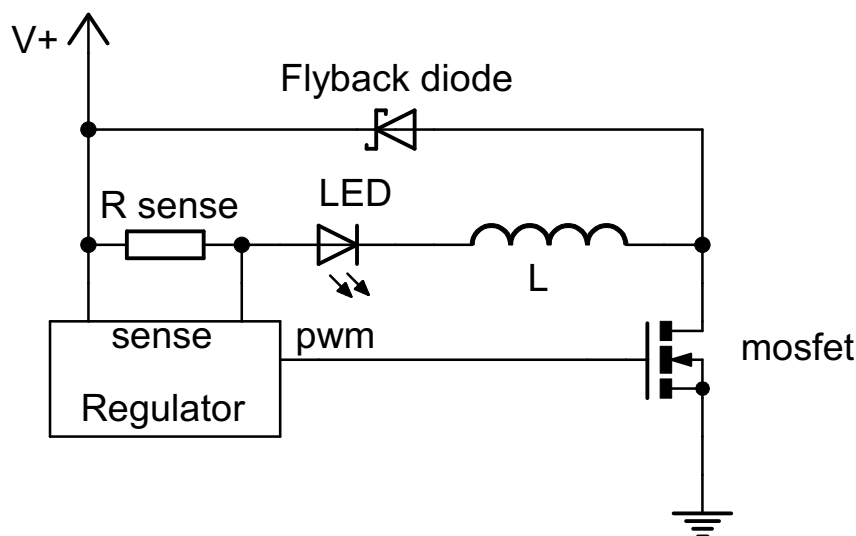
Lysdioder har en eksponentiell strøm-spenning karakteristikk. I spenningsområdet der dioden lyser vil en liten endring i spenning over dioden gi en stor endring i strøm gjennom dioden. Denne ulineære karakteristikken er heller ikke nøyaktig lik for to vilkårlige dioder. Dette betyr at en enkel spenningsregulator ikke kan benyttes til å drive lysdioder uten å ha en måte å regulere strømmen på.

En enkel, men ueffektiv måte å begrense strømmen på er å koble en motstand i serie med dioden. En bedre løsning er en svitsjet strømforsyning som regulerer strømmen gjennom dioden med minimalt tap. Dette er illustrert i Figur 16. Regulatoren benytter pulsbreddemodulering av svitsjetransistoren for å regulere strømmen gjennom dioden. Et eksempel på pulsbreddemodulering er illustrert i Figur 15. Svitsjefrekvensen ligger som oftest i kilohertz-området [8]. En slik løsning har en rekke fordeler, utgangen der dioden er koblet til tåler vanligvis å kortsluttes, og svitsjekretsen har ofte en inngang for å bestemme strømstyrken. Lavere strøm i lysdioden gir mindre lysutbytte, men gir også høyere effektivitet.



Figur 15: Pulsbreddemodulert 100Hz signal med 40% pulsbreddenid.

En måte å regulere lysstyrken på er å regulere den kontinuerlige strømmen i dioden. Problemet med dette er at farge og temperatur på lyset kan endre seg hvis den kontinuerlige strømmen er langt unna det lysdioden er laget for [8]. Løsningen på dette problemet er å benytte pulsbreddemodulering av strømmen i dioden. Ved å sende optimal strømstyrke en del av perioden, og så slå strømmen helt av resten av perioden med høy nok frekvens vil det se ut som dioden lyser kontinuerlig, men svakere. Frekvensen bør ligge over 100Hz for at lyset skal se kontinuerlig ut for mennesker. Kombinert med en pulsbreddemodulert LED-driver vil dette gi lavfrekvent pulsbreddemodulering av referansen til en høyfrekvent pulsbreddemodulert strømregulator.



Figur 16: Prinsippskisse for en konstant strøm LED-driver. Regulatoren prøver å holde spenningen over motstanden konstant ved å pulsbreddemodulere transistoren. Konstant spenning over R_{sense} gir en konstant strøm i lysdioden.

2.10 Bootloader

En bootloader er i denne sammenhengen et lite dataprogram som er det første som blir kjørt hver gang en pc eller mikrokontroller starter opp. Dette lille programmet har som oppgave å gjøre systemet klart for kjøring av hovedprogrammet. Hovedprogrammet vil foreksempel være et operativsystem for en vanlig pc, eller hovedapplikasjonen for en mikrokontroller.

En mikrokontroller kjører vanligvis programinstruksjoner rett fra flashminnet. Flashminnet er som oftest programmerbart via et hardware-grensesnitt som JTAG eller lignende. Hovedapplikasjonen kan i så fall godt ligge alene i flash-minnet og bli kjørt automatisk ved hver oppstart. Bootladeren sin oppgave her kan være å skrive et nytt program til flashminnet hvis brukeren ønsker det, eller verifisere at hovedprogrammet i flashminnet ikke er modifisert eller korrupt.

Selv om et bootloaderprogram for en mikrokontroller ikke er nødvendig, kan det ha mange fordeler. En bootloader vil gi mye større fleksibilitet ved opplasting av ny firmware til flashminnet. En kan for eksempel sende den nye koden til bootladeren over en trådløs-link eller CAN-bus, bootladeren kan så skrive programmet til flash-minnet og starte hovedapplikasjonen. For applikasjoner der programvaren ofte må oppdateres, og mikrokontrollerens hardware-programmeringsgrensesnitt er fysisk utilgjengelig, er en bootloader et uvurderlig verktøy.

3 Kravspesifikasjon

For å designe et system som løser de oppgaver det er ment å løse er det viktig å sette opp spesifikasjonskrav på forhånd. Dette vil gjøre designarbeidet enklere da en vet hva en må forholde seg til. Det er også svært vanskelig å bli ferdig med et produkt dersom kravene endres når detaljdesignfasen er påbegynt. Reglene fra arrangøren angående instrumentering i bilene stiller bare krav til sikkerhetssystemet og bilens utvendige belysning. Resten av kravspesifikasjonen er utarbeidet på bakgrunn av tidligere års erfaringer og ønsker fra sjåfør og resten av gruppen, samt ideer som har dukket opp underveis i arbeidet.

3.1 Krav og forutsetninger foreslått i prosjektoppgave

En del av de overordnede kravspesifikasjonene til instrumentering og elektronikk i bilen ble bestemt i prosjektoppgaven i løpet av høsten [1]. En oppsummering av de viktigste kriteriene som gjelder hele systemet er gjengitt i tabell 1.

- 1 Modulbasert system.
- 2 Enkelt å koble moduler inn og ut.
- 3 Kommunikasjon mellom moduler basert på en automotive standard.
- 4 Mulighet for å bruke moduler som alenestående enheter.
- 5 Energieffektivitet.
- 6 “Plug and play” -funksjonalitet.

Tabell 1: Generelle krav til systemet som ble foreslått i prosjektoppgave

Det som allerede er bestemt og må sees på som fastlagte krav er bus-kommunikasjon basert på CAN-bus, og 24-25V spenningsforsyning fra et batteri. CAN-bus gir en trygg og sikker kommunikasjon mellom moduler i bilen og er allerede en veldig vanlig standard brukt i vanlige biler.

3.2 Reduksjon av ledningsføring i bilen

Ved å basere systemet på en bus-arkitektur vil antallet ledninger som må gå lange avstander i bilen bli redusert til et minimum. Nettopp dette er også ideen bak introduksjonen av bus-basert kommunikasjon i vanlige biler. Et viktig moment ved det nye systemet bør være å plassere moduler i bilen slik at lengden på ledninger som må trekkes til for eksempel lys og sensorer er så liten som mulig. Korte ledninger veier mindre og er lettere å administrere og feilsøke.

3.3 Moduler som alenestående enheter

Både under utvikling og bruk av systemet kan det være ønskelig å benytte eller teste deler av bilen hver for seg. Dette må derfor også elektronikken støtte. Ved å lage systemet slik at delmoduler kan tas ut og brukes alene vil nytteverdien øke, og systemet blir lettere å feilsøke. For eksempel vil brenselcellestacken det meste av tiden stå på en lab og kjøres last-tester og simuleringer på, denne delen av systemet må dermed fungere separat. Ved utvikling av et grensesnitt for sjåføren er det nyttig å kunne benytte skjerm, knapper og lignende separat fra resten av systemet. Å kunne bruke moduler alene er derfor et viktig kriterium.

3.4 Lys og signalhorn

Reglene for konkurransen sier at lys og signalhorn må være tilstede på bilen. I tillegg til det reglene sier er det ønskelig at så mye som mulig på bilen skal være energieffektivt. En mulighet for å regulere lysstyrke og dermed spare strøm er derfor interessant. En oppsummering av hva reglene krever er her gjengitt:

- Minst to kjørellys i front
- To blinklys i front
- To blinklys bak
- Minst to bremselys bak
- Godkjent signalhorn

Godkjent signalhorn betyr at kun signalhornet som kan kjøpes fra Shell Eco Marathon-nettbutikken er tillatt.

Kjørellys og Fjernlys

Ettersom bilen allerede har frontlykt-design som inkluderer plass til 6 lys totalt er det planlagt å ha 4 hvite lys og 2 gule blinklys. De 4 hvite lysene er referert til som kjørellys eller nærlys og fjernlys, men i praksis er det ikke viktig at fjernlys lyser lenger enn nærlysene. Kjørellys og fjernlys bør lyse sterkt nok til være synlig på lang avstand og helst kunne lyse opp litt av veien i front av bilen.

Bremselys

Reglene sier at det holder med røde lys som tennes dersom bilens bremsesystem aktiveres, men vanlige biler har også røde lys bak som lyser svakere ved normal kjøring, og blir mer intense under bremsing. Dette er derfor en interessant utvidelse til det reglene nevner som minimum. Det bør også være mulighet for et høyt plassert bremselys som tennes kun ved bremsing som på vanlige biler.

Blinklys

Bilen må være utstyrt med blinklys eller indikatorlys som blinker foran og bak på høyre eller venstre side avhengig av hvilken vei sjåføren velger å svinge. Et krav til disse lysene er at de har en gul/oransje farge og blinker med passende frekvens. Også blinklysene bør ha justerbar lysstyrke for å spare strøm.

Signalhorn

Reglene stiller krav til hvilket signalhorn som kan benyttes. De viktigste kriteriene for styring av hornet er at sjåføren kan bestemme når og hvor lenge hornet skal være aktivert ved å trykke og holde inne en knapp på rattet. Funksjonen skal være akkurat som på en vanlig bil. Det er viktig at responsen er umiddelbar, det vil si at sjåføren kan trykke veldig fort på tute-knappen og det skal da komme et tilsvarende kort signal fra hornet.

3.5 Sjåfør-grensesnitt

Grensesnittet mellom sjåføren og bilens elektronikk er et viktig aspekt ved designet av instrumenteringssystemet. Som hovedidé er det tenkt at det skal være mest mulig likt som i en vanlig bil. Dermed vil sjåførens jobb være mest mulig intuitiv og enkel å sette seg inn i uten mye opplæring.

Fotpedaler

I bilen vil det bli montert to pedaler, en for gass og en for brems. Gasspedalen må fungere som en vanlig gasspedal og bør helst ha en viss oppløsning i responsen, det vil si at kun av og på ikke er bra nok. Bremsepedalen vil virke direkte på hydraulikken for bremsene og det skal være mulig for sjåføren å bremse ved kun å trykke på bremsepedalen. En sensor i pedalen må derfor monteres for å vite når bremselys skal aktiveres og deaktiveres.

Skjermløsning

For at sjåføren skal vite hva som skjer med bilen til enhver tid er det viktig at sjåføren har enkel tilgang til parametere fra styringssystemet. En måte å gjøre dette effektivt på er å benytte en skjerm for sjåføren. Skjermen må kunne vise tekst og tall, men grafikk er ikke vesentlig. Størrelsen på skjermen bør være såpass at den er lett leselig på en meters avstand. Antallet bokstaver på skjermen bør være relativt høyt for å slippe å følge med på rullende tekst eller lignende. Ca. 4 linjer og 15-25 bokstaver per linje burde holde til det meste. Dette er dobbelt så stort som 2 linjers displayet fra året før. Displayet må ha bakgrunnsbelysning for bruk i mørket, og samtidig være lett å lese i sollys.

Meny-navigering

For å få tilgang flere parametere enn det er plass til på skjermen på samme tid er det ønskelig med et meny-system. Dette vil også gi mulighet for å ha egne menyvalg med innstillinger og valg for systemet. Dersom en meny skal implementeres må det være lett å navigere i denne menyen. En gjennomprøvd løsning fra lignende systemer bør benyttes, og det må være intuitivt å navigere.

Knapper

I tillegg til skjerm og et menysystem må det være egne knapper for vesentlige funksjoner i bilen. Funksjoner som benyttes under kjøring kan ikke være vanskelig tilgjengelig i et menysystem, da det fort kan bli trafikkfarlig å navigere i en meny i fart. De viktigste funksjonene som bør ha knapper er av/på for brenselcelle, indikatorlys og kjørellys, signalhorn og cruisecontrol. I tillegg kan det være greit med knapp for nødblink, og noen ekstra knapper som kan få forskjellige funksjoner etterhvert som behov dukker opp. Dette kan gjøres på samme måte som mobiltelefoner og lignende som ofte har rekonfigurerbare knapper.

Visuell feedback

For sjåføren vil det være viktig å få tilbakemelding fra systemet på en enkel måte. Å få all informasjon via skjermen kan bli tungvint og er lite hensiktsmessig. En optimal løsning vil være lysdioder for bilens viktigste parametere. Foreksempel om brenselcellen er på eller av, eller om kjørelysene er aktive. En interessant idé for statuslys kan være knapper med separat lys i knappen. Statuslysene bør kunne aktiveres uavhengig av knappens posisjon, dermed kan lysene signalisere feil uavhengig av knappeposisjon.

Aural feedback

Visuell feedback krever at sjåførens øyne ofte scanner over instrumentpanelet og får med seg om tekst eller lys har endret seg siden sist. For viktige beskjeder kan det være ønskelig med feedback i form av lyd. Vanlige biler har foreksempel en tikke-lyd dersom blinklys er aktivert. En alarm-lyd for vesentlige funksjonsfeil ved bilen er også interessant. Lyden må være lett hørbar for sjåføren, selv med kjørehjelm og handsfree.

3.6 Powerpack

Eco-marathon gruppen har i fellesskap bestemt at bilen skal kunne kjøres på både batteri og brenselcelle. Denne modulen i systemet skal dermed være lett å bytte. Modulen som inneholder enten brenselcellestaken og hydrogen eller et batteri er kalt powerpack. Styringssystemet må fungere både med og uten brenselcellestaken montert. For sjåføren bør det ikke spille noen rolle hva energikilden er, betjeningen av bilen må være akkurat den samme uavhengig av powerpack'en.

3.7 Brenselcellestyring

Dersom brenselcellen er montert må systemet ta seg av styring av denne. Hovedsakelig består styring av brenselcellen i å overvåke de viktigste parameterene og enten aktivere en kjølevifte dersom levert effekt er høy, eller skru systemet av dersom celledspenninger faller under en kritisk verdi. Dersom levert effekt er høy vil det merkes ved høy temperatur og lav totalspenning over staken.

Detaljer rundt kravene og parameterene til styringen av staken er ikke fastlagt på designtidspunktet, systemet må derfor kunne rekonfigureres til vilkårlige styringsstrategier. Styring og overvåkning av brenselcellen baserer seg først og fremst på å kunne foreta en rekke målinger på staken, og aktivere hydrogen-ventil, purge-ventil, releer, vifte og lignende basert på målingene.

Multicelle spenningsmåling

En av de viktigste parameterene til brenselcellestaken er spenningen til hele staken og til individuelle celler. Å kunne foreta individuelle celledspenningsmålinger ble forsøkt gjennomført de tidligere årene, men uten suksess. Dette er dermed et sterkt ønske for årets styringssystem. Det er 46 celler i staken. Cellene er koblet i serie og spenningen på hver celle vil ligge mellom 0.5V og 1.2V ved normal drift.

Måling at strøm

For å vite når det er nødvendig å purge bør systemet kunne måle strøm gjennom stacken. Strømmen kan ikke bli negativ, da dette vil ødelegge brenselcellen. Ved normal drift vil strømmen ligge mellom 2 og 3 ampere, men den kan komme opp i 5-10A ved akselerasjon. En strømmåling bør ikke medføre vesentlig tap i systemet, og enkel tilkobling er viktig. En induktiv strømgiver som kan monteres rundt en vanlig ledning, uten å måtte koble inn en shunt-motstand, er derfor ønskelig.

Trykksensorer

Hydrogenstrykket inn på stacken bør ligge mellom 1,2 og 1,5 bar ved normal drift [9]. Selve reguleringen av trykket er planlagt løst med en to-trinns mekanisk trykkregulator rett etter hydrogenflasken. Allikevel er det ønskelig å overvåke trykket, og kunne logge dette digitalt. Foregående år var det montert to trykkmålere, en før og en etter hovedventilen. Om dette fortsatt er nødvendig er uvisst, et opplegg for minst to trykkmålere er derfor nødvendig.

Temperaturmåling

En annen viktig parameter for brenselcellen er temperaturen. Ved normal drift vil temperaturen være avhengig av levert effekt og mengden kjøleluft som blåses gjennom. Arbeidstemperaturen kan være alt fra romtemperatur til 70 – 80°C. For å vite om hele stacken oppfører seg likt, og at ikke noen celler er hardere belastet enn andre, er det interessant med en temperaturloggeløsning som er sonedelt. Å se temperaturen på minst 4 soner i stacken, i tillegg til omgivelsestemperaturen, er et greit utgangspunkt.

Ventil og reléstyring

For ventiler og elektriske brytere (releer) som må styres er det ønskelig å legge opp elektronikken slik at noen flere releer og ventiler enkelt kan kobles på hvis det er nødvendig. Samtlige ventiler og relé som er benyttet tidligere år er laget for 24V. Releet og ventilene er valgt i en tidligere masteroppgave på bilen, se Gellein 2009 [9]. Solenoideventilene og releet fungerer bra og skal derfor benyttes videre. Ventilene er de som trekker mest strøm, 170mA. Det er viktig å ta hensyn til at ventiler og releer ikke oppfører seg som en rent resistiv last, men inneholder elektriske spoler og er induktive.

Sikkerhetssystem

Kravene til sikkerhetssystemet er de samme som i 2009. Reglene krever blant annet at hydrogentilførsel og hovedstrøm til motoren skal kuttes dersom nødstoppeknappen inne i bilen aktiveres eller nødstoppen på utsiden av bilen aktiveres. Det skal også være installert en hydrogensensor i motorrommet som har sammen funksjon som nødstopp ved deteksjon av hydrogengass. En oppsummering av reglene for sikkerhetssystemet fra (Gellein 2009 [9]) følger:

- En normalt lukket solenoideventil, montert rett etter flaskeregulatoren. Den skal stoppe hydrogentilførselen ved nødstopp.
- Et lastrelé, type normalt åpen, skal kutte strømmen mellom brenselcelle og motor ved nødstopp.
- En manuelt operert bryter og en sikring, koblet mellom brenselcelle og motor. Sikringen skal være dimensjonert i forhold til brenselcellestacks celleareal.
- En hydrogensensor som aktiverer nødstopp ved en hydrogenkonsentrasjon lavere enn 25% av LEL (Lower Explosive Limit), dette tilsvarer en konsentrasjon på 10 000 ppm.
- En nødstopp-knapp i cockpit.
- En utvendig nødstopp-knapp, som også bryter kretsen til bilens batteri.
- Hydrogensensoren og bilens horn må forsynes av strøm fra bilens batteri, det er også lov å forsyne nød-nedstengningsventilen og lastrelé fra batteriet.
- Dersom man bruker plastslanger, må trykket i dem ikke overstige 0,5 bar overtrykk.

3.8 Kommunikasjon med motorstyring

Motorstyringen er den elektronikkenheten som omdanner likespenningen fra brenselcellen til trefase vekselspanning som er tilpasset motorens turtall. Motorstyringen i bilen er lånt fra Smart Motor og er den samme som ble brukt i 2009. Grensesnittet til motorstyringen består av en vanlig RS-232 serieporttilkobling med en egen kommunikasjonsprotokoll. Styringssystemet må altså ha mulighet til toveis RS-232 kommunikasjon. Kun rx- og tx-linjene ble benyttet i 2009 og det er uvisst om flytkontroll er implementert i firmware på motorstyringen.

3.9 Opplasting av firmware

Etter at all elektronikken er montert i bilen vil det garantert være nødvendig å gjøre endringer i software, som så må overføres til en eller flere mikrokontrollere i systemet. For utvikling av software på kontorpulten går det ofte greit å ha en programmeringsboks (også kalt JTAG) koblet rett til kretskortet med mikrokontrolleren på. Denne løsningen blir imidlertid veldig tungvint dersom mikrokontrollerene sitter spredt rundt i bilen og er vanskelig tilgjengelige.

Kretskortene vil også sannsynligvis bli montert i bokser med kun kommunikasjonsbuss-tilkobling. Det er derfor nødvendig med en enkel og robust måte å overføre ny software til en vilkårlig modul i systemet. Denne operasjonen bør helst kunne foretas ved kun å koble en pc til en lett tilgjengelig tilkobling på bilens kommunikasjonsbuss.

3.10 Beskyttelse av elektronikken

Kretskort og elektronikk er ofte veldig følsomt for miljøpåvirkninger. Fuktighet og statisk elektrisitet er to store og potensielt ødeleggende påvirkninger som kretskort må beskyttes mot. Reglene for konkurransen stiller krav til at hvis elektronikk skal innkapsles må det være i gjennomsiktede bokser, eventuelt bokser med vindu i. Dette er sannsynligvis for å gjøre det vanskelig å gjemme batterier i bilen. Det er ønskelig med en hendig formfaktor på modulene i systemet, små nok til å være enkle å plassere og håndtere, men store nok til å få plass til nødvendig elektronikk. Bokser med ytre mål på ca. 15*15*4cm er et passende utgangspunkt.

3.11 Diverse ideer

Under idémyldrefasen for systemet kom det opp noen forslag til funksjonalitet som ikke er direkte nødvendig, men som kan vise seg å være praktiske og som bidrar til med en ekstra show-faktor.

Blåtannfunksjonalitet

Selv om telemetri-modulen har GPRS-tilkobling kan det være interessant å styre diverse funksjoner i bilen trådløst over blåtann. Foreksempel kan lys og signalhorn betjenes fra en mobiltelefon i nærheten. Status for diverse systemparametere kan monitoreres, og mobiltelefonen kan også brukes som kjøredisplay ved å feste den i ratt eller dashboard foran sjåføren. Mulighet for å koble blåtannenheter til systemet åpner for en rekke interessante ekstrarfunksjoner.

Lysdioder på kretskort

Under utvikling av software, og for raskere diagnose av problemer etter montering i bil kan det være nyttig med lysdioder som er montert på kretskortene. Dette kan benyttes til å gi umiddelbar status fra en elektronikkmodul uten at kommunikasjon med resten av systemet nødvendigvis fungerer. Et kretskort uten lysdioder er tungvint å undersøke om fungerer eller ikke. Det faktum at innkapslingen til elektronikken må være gjennomsiktig gjør det ekstra attraktivt å kun montere små lysdioder rett på kretskortene. Etersom batteriet til styringssystemet er montert i motorrommet kan det være greit å se gjenværende kapasitet på dette i motorrommet, dette kan løses ved å montere lysdioder rett på et kretskort i motorrommet.

4 Design

4.1 Systemløsning

Første del av designprosessen er å bestemme den overordnede oppbygningen av instrumenterings-systemet. Fra kravspesifikasjonen er det fastlagt at systemet skal være modulbasert og antallet ledninger som må trekkes rundt i bilen skal minimeres. Moduler må derfor plasseres i umiddelbar nærhet til det de er ment å styre eller overvåke for å minimere ledningsstrek.

Plassering og funksjon

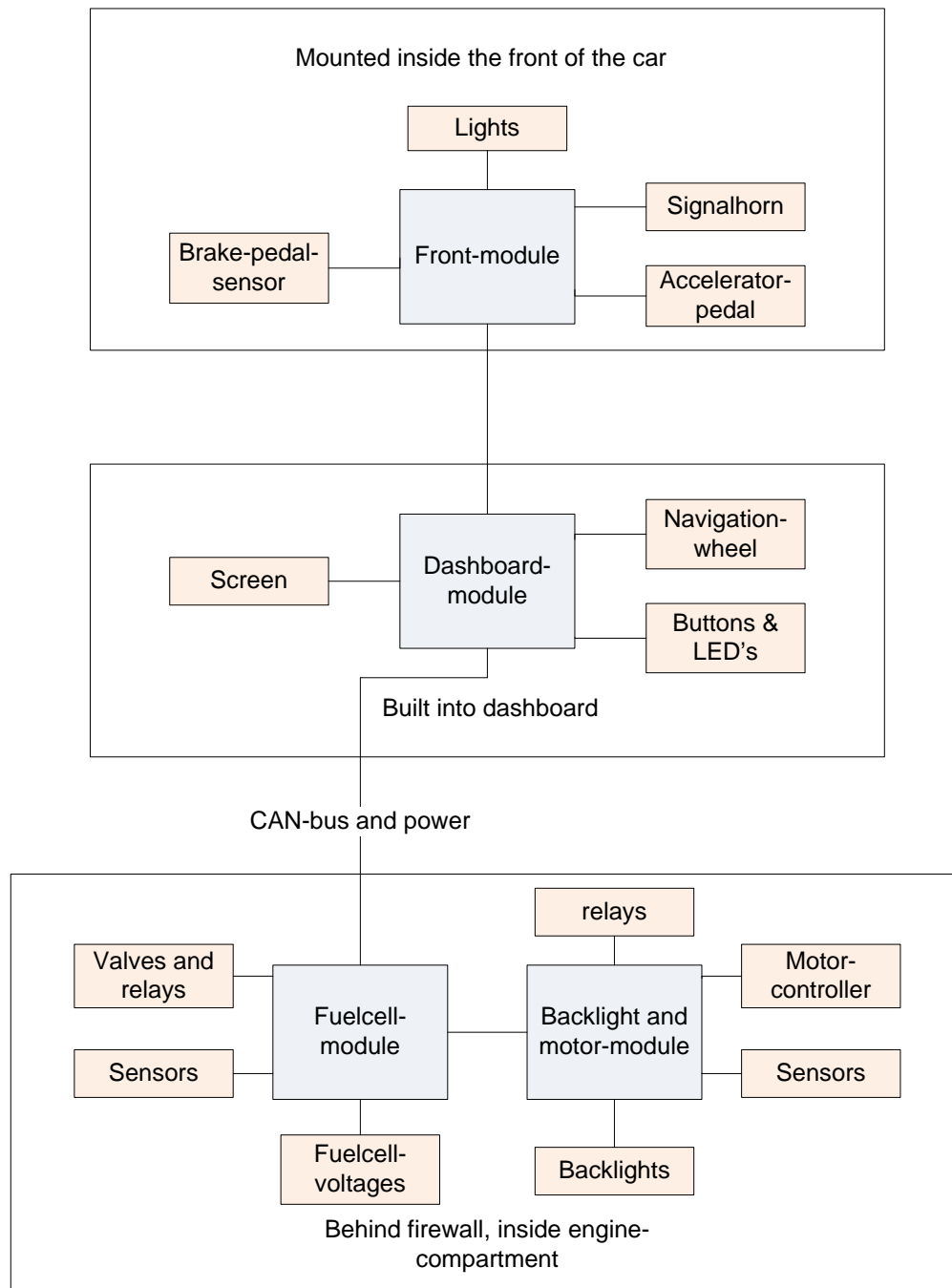
Ved å se på hvor i bilen mesteparten av kontroll og overvåkningsbehovene befinner seg blir det enkelt å dele opp systemet i et passende antall moduler. Helt framme i snuten på bilen er det styring av lys, signalhorn og input fra gasspedal og bremsepedal som er nødvendige funksjoner. Dashbordet trenger en modul for styring av skjerm, input fra knappetrykk og en liten alarm-høytaler. I motorrommet er det både nødvendig med styring av baklys, kommunikasjon med motorstyringen og kontroll av brenselcelle.

Det mest hensiktsmessige for motorrommet vil være å lage en egen modul for brenselcellen. Ideen bak powerpack-prinsippet med utbyttbar energikilde i bilen gjør at brenselcellemodulen må kunne erstattes med kun et batteri for kjøring. Funksjoner som baklys og motorstyring som ikke er relatert til brenselcellen kan derfor ikke plasseres i brenselcellemodulen. En skjematisk oversikt over systemet og modulene slik de er planlagt plassert er gitt i Figur 17.

Modulenes oppbygning

Ettersom systemet skal baseres på CAN-bus er det nødvendig med en CAN-transceiver og prosessering i hver modul. Fra designet av telemetrimodulen ble Atmel sin AT90CAN-128 mikrokontroller vurdert til det beste valget [1]. Den enkleste løsningen for hele systemet er å kopiere kretsløsningen som ble laget for telemetrimodulen. Hver modul bør være bygget rundt samme mikrokontroller for å oppfylle kravet om enkel opplasting av ny firmware til en vilkårlig modul i systemet. Den samme argumentasjonen gjelder for strømforsyning og en eventuell serieporttilkobling for utvikling og debugging. På grunn av dette ble så mye som mulig av funksjonaliteten, som hver modul må inneholde, kopiert fra telemetrimodulen.

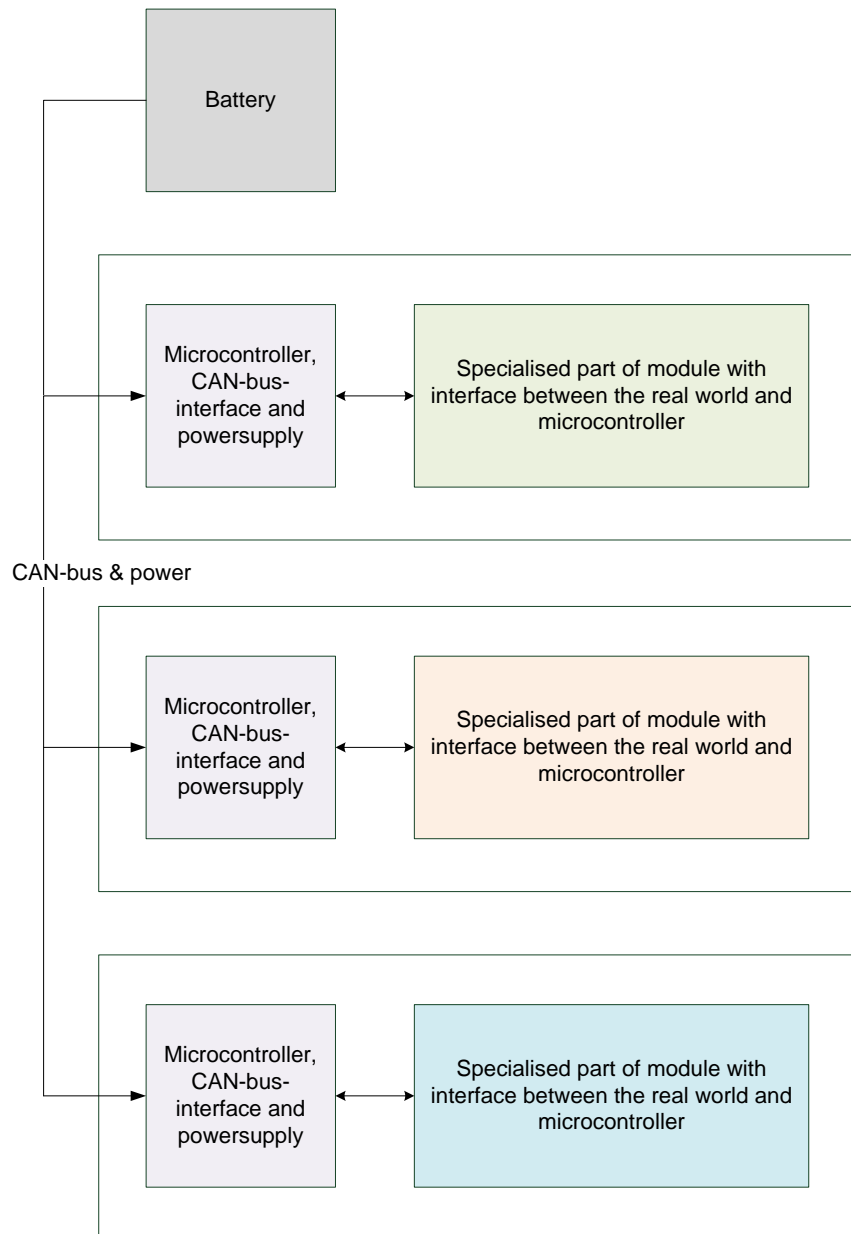
Et moduldesign som baserer seg på et eget universalkretskort med mikrokontroller, CAN-transceiver, USB-tilkobling og spenningsregulatorer i hver modul ble valgt. Det er en rekke fordeler ved å samle mikrokontroller, spenningsforsyning og CAN-bus-kommunikasjonen på et separat kretskort. Det blir mye lettere å utvikle spesial-delen i hver modul og det er lettere å isolere feilen dersom noe ikke fungerer. Å modularisere hver modul er også i tråd



Figur 17: Plasseringen og funksjon til modulene i bilen bestemmes av hva som må styres i den delen av bilen. Brenselcellemodulen kan tas ut av systemet dersom et batteri benyttes som energikilde.

med modulbaseringen av resten av systemet. Funksjonaliteten til hver modul endres ved å kun lage nye tilleggs-kretskort. Ved implementasjon og produksjon er det også gunstig

med mange like kretskort. Prinsippet med en universalmodul og en spesialmodul i hver elektronikk-boks er gjengitt i Figur 18.

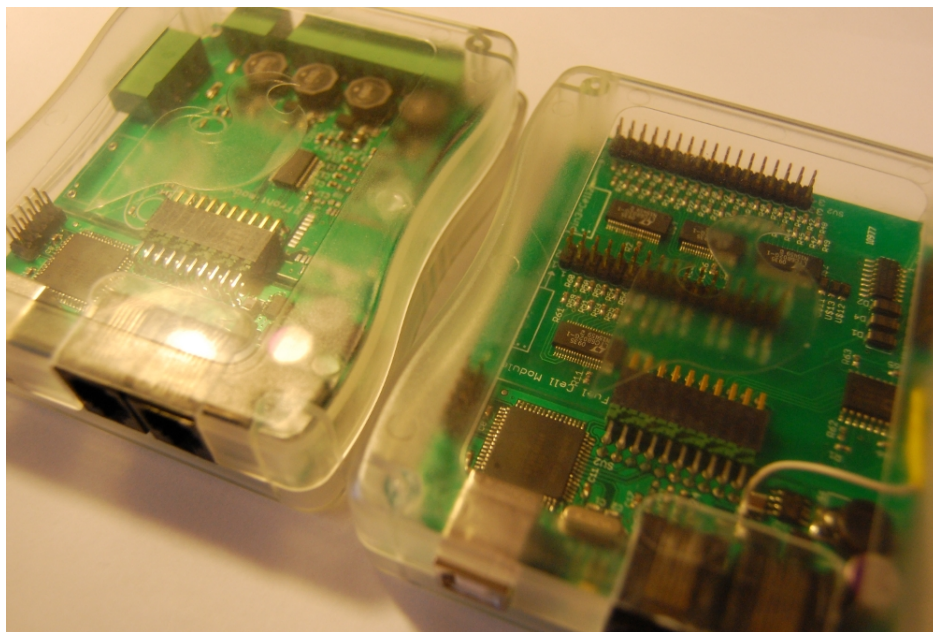


Figur 18: Hver modul i systemet består av en universaldel (lik farge) og en spesialdel (ulik farge). Universaldelen har identisk hardware i alle moduler.

4.2 Innkapsling av elektronikken

Under prosjektarbeidet med telemetrimodulen ble det gjort mange erfaringer med innkapsling av elektronikk. En viktig lærdom var at å finne en egnet boks til elektronikk etter at kretskort er designet er så godt som umulig. Å bestemme formatet på innkapsling og deretter finne en egnet boks eller lignende må gjøres før arbeidet med elektronikken begynner.

Gjennomsiktig boks



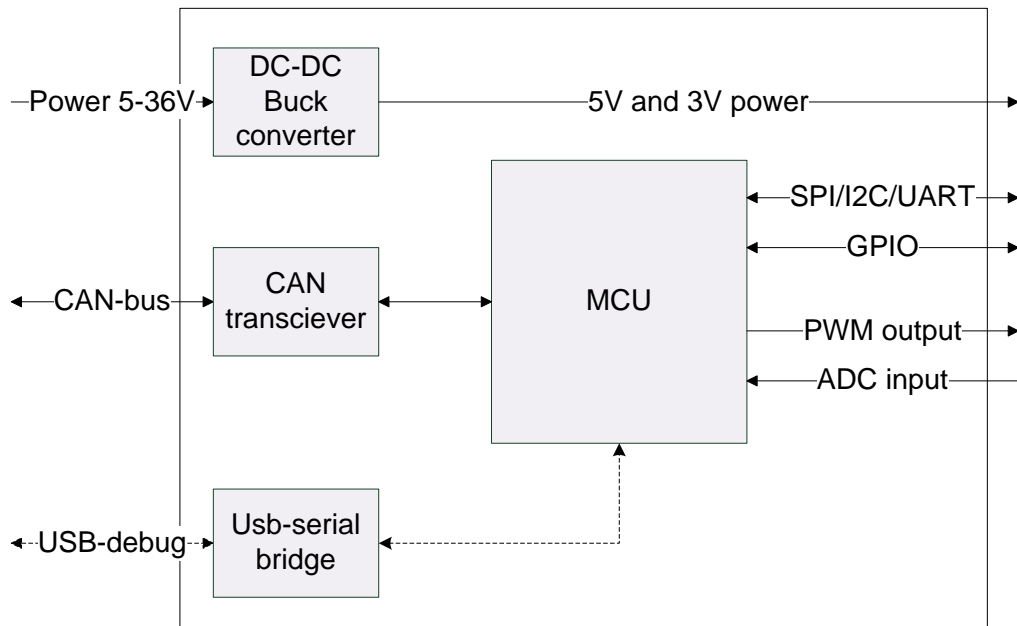
Figur 19: Gjennomsiktig boks til elektronikk fra Sparkfun. Største kretskort som får plass i boksen er 80*100mm.

For telemetrimodulen ble det benyttet en boks funnet på komponentlageret til teknisk kybernetikk. Problemet med boksen var at et vindu måtte monteres, og størrelsen på kretskortet som får plass er ikke veldig stort. En flatere boks med større areal og mer kvadratisk utforming er mer optimalt.

Mange av elektronikk-komponentene til både telemetrimodulen og resten av systemet er kjøpt på www.sparkfun.com. Denne nettbutikken har også en gjennomsiktig prosjektboks som er tilpasset kretskort med størrelse 80*100mm. Til tross for mye leting på internett etter alternativer ble det ikke funnet noen bokser som kom i nærheten av å være like bra egnet som de gjennomsiktige sparkfun-boksene (Figur 19). Boksene og kretskortet holdes sammen med 4 skruer, det er egen bakplate for tilkoblinger og nok plass over og under kretskortet til montering av diverse komponenter på begge sider av kretskortet.

4.3 Universalmodul

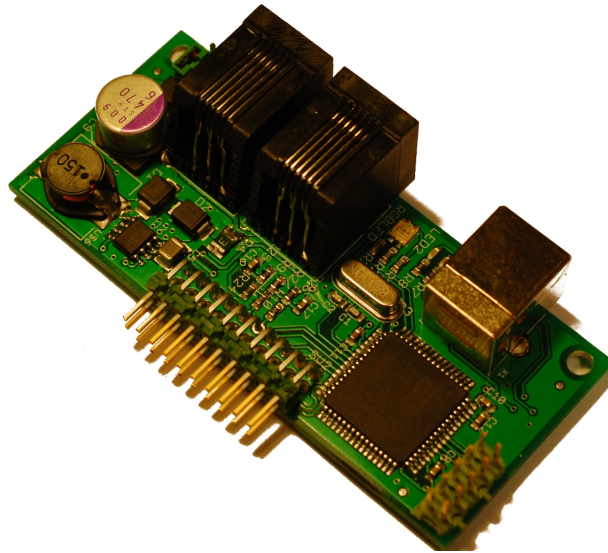
Universalkretskortet er den delen av hver modul i systemet som har lik hardware. Alle de viktigste kretsløsningene på universalmodulen er hentet fra telemetrimodulen og ble testet i løpet av høsten 2009. Overordnet skjemattegning er gitt i Figur 20. Figur 21 viser ferdig kretskort for universalmodulen.



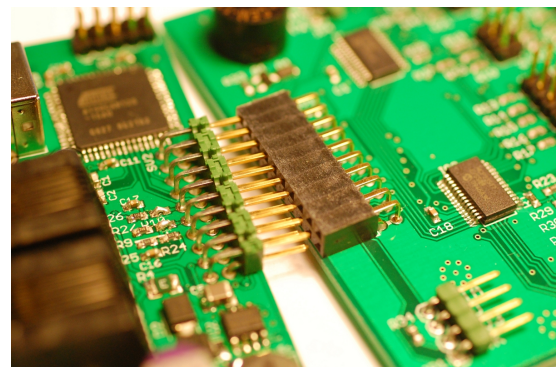
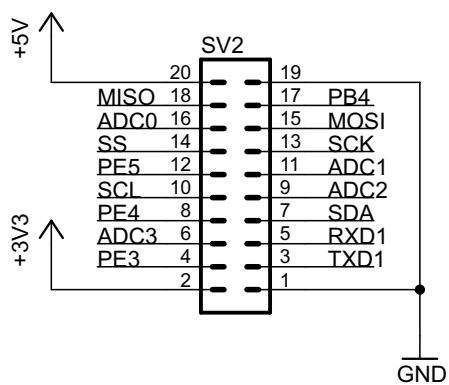
Figur 20: Universaldel i hver modul med grensesnitt mot resten av systemet til venstre og kommunikasjon med en spesialdel til høyre. Grensesnittet på høyre og venstre side er identisk for alle modulene i bilen.

Konnektor mellom kretskort

En standard 90° bøyd pinnekontakt for å koble kretskortet for universalmodulen sammen med spesialmodulen ble valgt. Det er lagt opp 20 linjer til spesialmodulen, de forskjellige linjene som er lagt opp er vist i Figur 22. Kontakten gir både mekanisk stabilitet og tåler mange inn og utkoblinger. Bilde av sammenkoblingen er også gitt i Figur 22.



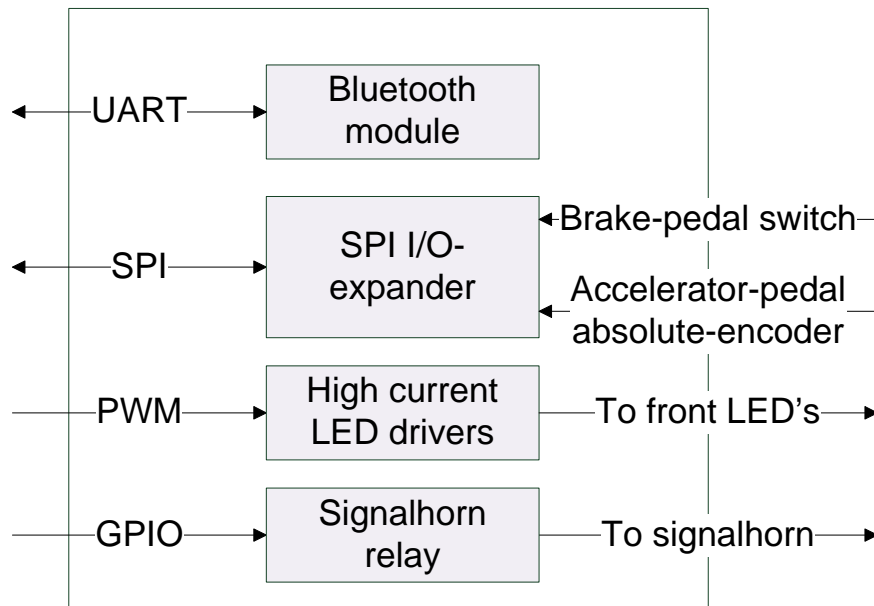
Figur 21: Ferdig kretskort for universaldelen i hver modul.



Figur 22: Sammenkoblingen mellom universalkretskortet og spesialdelen i hver modul. Kontakten gir mekanisk stabilitet, er robust og tåler mange koblingscykler.

4.4 Frontmodul hardware

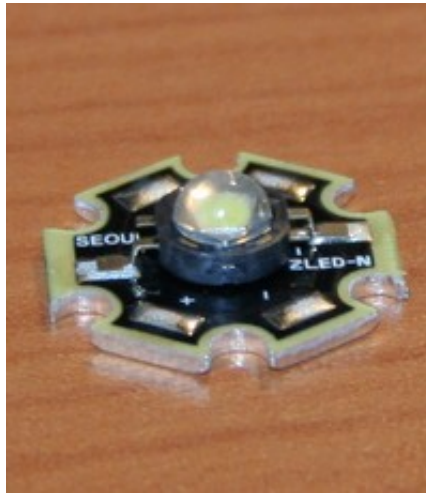
Modulen foran i bilen skal ta hånd om 6 lysdioder til nær-, fjern- og blinklys, signalhorn, input for gasspedalvinkel og bremsepedalbryter. I tillegg til dette ble blåttannmodulen montert her fordi det var plass til overs på kretskortet. En overordnet skjemategning av spesialdelen i frontmodulen er gitt i Figur 23.



Figur 23: Overordnet skjema for spesialdel i frontmodulen.

Valg av lysdioder

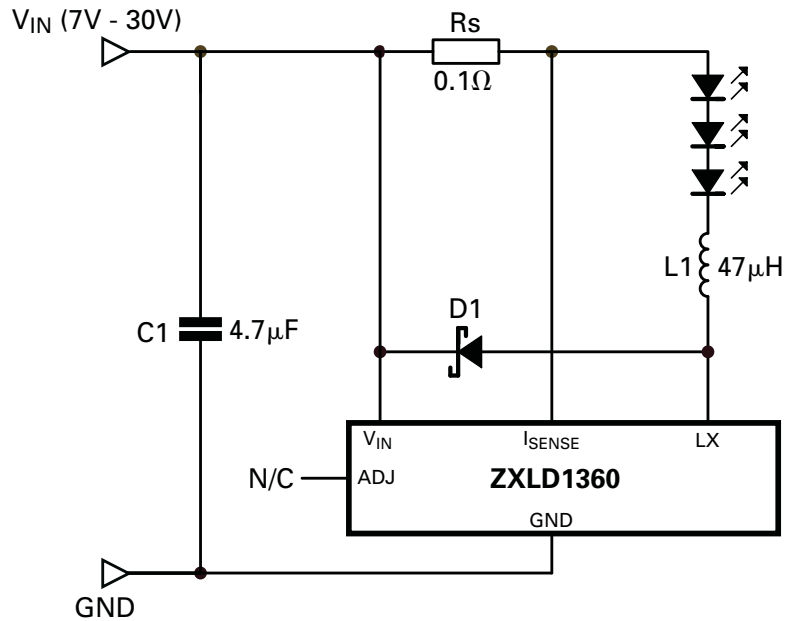
Siden lysdiode-drivere skal lages på nytt ble det besluttet å skaffe nye og kraftigere lysdioder. Lysdiodene som var montert i bilen var laget for strømstyrke opp til 350mA. Hvite lysdioder laget for 1A strømstyrke vil gi vesentlig mer lys. De viktigste kriteriene ved valg av lysdioder var mye lys per watt og at diodene er montert på en liten aluminiumsplate som gjør de enklere å montere i bilens lykter. Valget falt på Seoul Semiconductor sine Z-power-P4 lysdioder (se Figur 24). De har meget høy lysstyrke per watt og kan kjøpes ferdig montert på en aluminiumsplate som fungerer som en liten kjøleribbe. En fordel med akkurat denne lysdiode-typen er at den finnes i både orange og rød utgave i tillegg til hvit [10]. Dette gjør at samme type diode kan brukes til blinklys og bremselys uten fargefilter. De fargede versjonene har maksimal strømstyrke på 350mA.



Figur 24: Hvit P4-lysdiode fra Seoul Semiconductor, montert på en stjerneformet aluminiumsplate som både fungerer som kjøleribbe og gjør monteringen i lyktene enklere.

Drive lysdioder

For å drive lysdiodene må strømmen gjennom dem reguleres, se teori (2). En kretsløsning der det er enkelt å styre intensiteten er ønskelig. Det enkleste vil være å benytte en pulsbreddemodulert linje fra mikrokontrolleren direkte til lysdiode-driver-kretsen. ZXLD1360 fra Zetex er en 1A og 30V stepdown lysdiodedriver med PWM-input og få nødvendige eksterne komponenter. At den tåler en spenning opp til 30V betyr at flere lysdioder kan kobles i serie og total levert effekt kan være opp mot 30W. En enkel kretsløsning er gitt i Figur 25.



Figur 25: Kretsløsning basert på Zetex lysdiodedriver. Strømreguleringen fungerer ved at kretsen forsøker å holde spenningen over R_s konstant, og dermed vil strømmen gjennom diodene være konstant. Adjust-inngangen kan styres enten med en analog spenning, eller et PWM-signal. (Zetex)

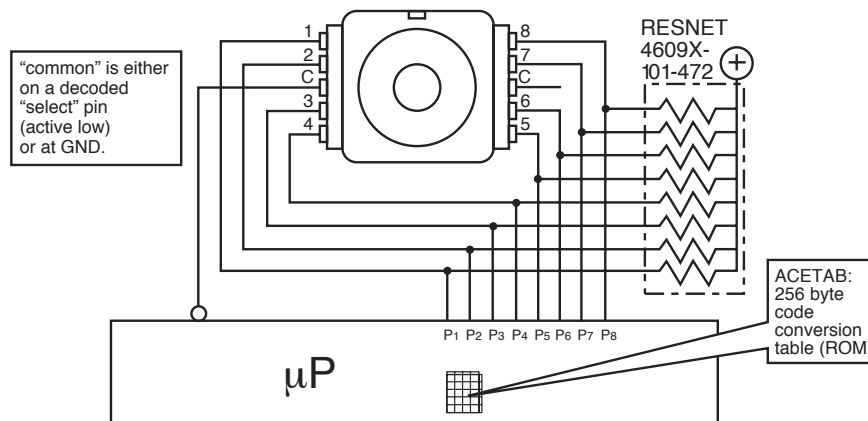
Gasspedal

Ettersom variabelt gasspådrag er ønskelig er en vinkelsensor nødvendig for å måle vinkelen til gasspedalen. De tidligere løsningene baserte seg på en variabel motstand. Erfaringene med dette var blandet, hovedproblemet var at det var støyfølsomt. Et alternativ til et potensiometer er en enkoder, en digital vinkelsensor. Ved å velge en absolutt-enkoder vil vinkelen til gasspedalen kunne leses av direkte uten å måtte følge med på endringer slik man må hvis man benytter en kvadratur-enkoder. En absolutt-enkoder ble derfor valgt.

Høyeste oppløsning som var mulig å oppdrive i fornuftig størrelse og pris på absolutt-enkodere var 128 steg per omdreining, se Figur 26. Dersom denne kobles direkte til gasspedalen vil det gi ca. 5-15 steg i oppløsning gitt vinkelendring på 15-45 grader. Vinkel gasspedalen kan trykkes inn var bestemt av mekanikk og plassbegrensninger foran i bilen. Oppløsningen er i minste laget, men å øke oppløsningen for en absolutt-enkoder viste seg vanskelig og mekanikk for utveksling var heller ikke aktuelt. Det ble besluttet å gå for løsningen med 5-15 stegs oppløsning da bilen allikevel skal kjøres med cruise-control i konkurransen. Enkoderen som baserer seg på elektrisk kontakt er koblet opp som vist i Figur 27.



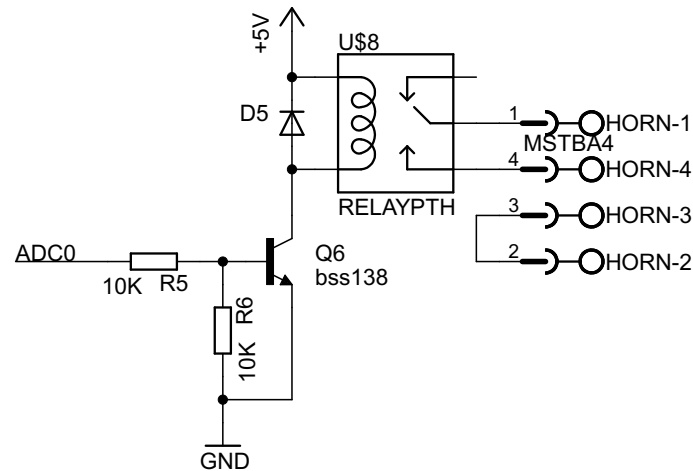
Figur 26: Absolutt-enkoder benyttet til gasspedal. Enkoderen har 128 steg per omdreining og baserer seg på elektrisk kontakt mellom stifter og sleperinger. (Bourns)



Figur 27: Skjema for hvordan absolutt-enkoderen kan kobles til en mikrokontroller. På høyresiden er pull-up-motstander vist. Vinkelen fåes ved å lese inn de 8 pinnene som én byte og foreta et tabelloppslag i software. (Bourns)

Signalhorn

Signalhornet som må benyttes er laget for 12V, siden batteriet i bilen skal være 24V må spenningen til hornet reduseres. En enkel løsning er å koble hornet i serie med en motstand. Siden hornet ikke skal brukes lenge av gangen vil effekttapet i motstanden ikke bety så mye for strømforbruket fra batteriet. Hornet trekker 3A ved 12V, det gir en motstand på 4Ω , se ligning (1). Effekten motstanden må tolerere blir 36W, ligning (2). For å skru hornet av og på er det mest hensiktsmessig med et rele på grunn av den høye strømmen som trekkes rett fra batteriet. Oppkobling er gitt i Figur 28.



Figur 28: Oppkobling av relé og transistor for signalhornet. Spenningsforsyning kobles til terminal 1 og 2 og hornet i serie med motstand kobles til 3 og 4. Polariteten for hornet er likegyldig.

$$\frac{U}{I} = R \quad (1)$$

$$\frac{12V}{3A} = 4\Omega$$

$$R \cdot I^2 = P \quad (2)$$

$$4\Omega \cdot 3^2 A = 36W$$

Input/output-expander

SPI-io-expanderen som er vist i Figur 23 håndterer gasspedalkoder og bremsepedalbryter. I tillegg driver io-expanderen 6 status-lysdioder på kretskortet. Io-expanderen som er valgt er en MCP23S17 fra Microchip (datablad [11]). Denne kretsen ble valgt fordi den fungerer ved 3.3V, den kan lagre interrupttilstand for pinner og har et enkelt SPI-grensesnitt.

Blåtannmodul

Ettersom kravspesifikasjonen nevnte blåtann som en interessant ekstrarfunksjonalitet og det var plass til overs på kretskortet for front-modulen ble en blåtannmodul plassert her. En enkel løsning ble valgt, en ferdig modul med antenne og alt innebygget som kan

loddet rett på kretskortet, se Figur 29. Modulen er fra Sparkfun og eneste tilkobling til mikrokontrolleren er 3.3V UART rx- og tx-linjer. Blåtannmodulen er laget for å erstatte serieportledninger og krever ingen driver eller lignende på mikrokontrollersiden. Etter installasjon av blåtanndriver på en pc med blåtannadapter dukker den opp som en vanlig COM-port.



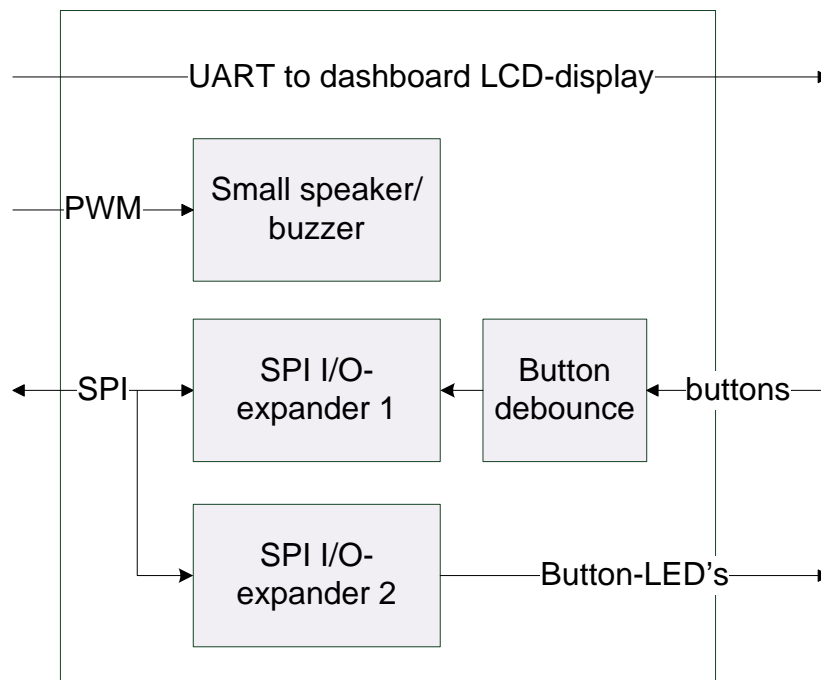
Figur 29: Blåtannmodul fra Sparkfun montert på frontmodulkretskortet, jordplanet under antennen er fjernet for å få bedre rekkevidde.

4.5 Dashbordmodul hardware

Dashbordmodulen må ta hånd om skjerm, navigering i meny, knappe og statuslysdioder. I tillegg skal modulen kunne lage signallyder til sjåføren. Et overordnet diagram er gjengitt i Figur 30.

Display

For å gjøre elektronikken for displayet så enkel som mulig ble det valgt en LCD-modul med serieporttilkobling og innebygget LCD-driver. Dermed kan mikrokontrolleren sende bokstaver og kommandoer over en enkel Tx UART-linje. LCD-modulen har 4 linjer, 20 bokstaver på hver linje og ble kjøpt på Sparkfun.com. Spenningsforsyningen til displayet er 5V, og det er integrert belysning i displayet. Displayet er også lett å lese i sollys på grunn av reflektivt belegg bak LCD-elementene.



Figur 30: Overordnet skjema for spesialdel i dashbordmodulen. Modulen har to io-expandere, en for knappetrykkinput og en for lysdioder i knappene.

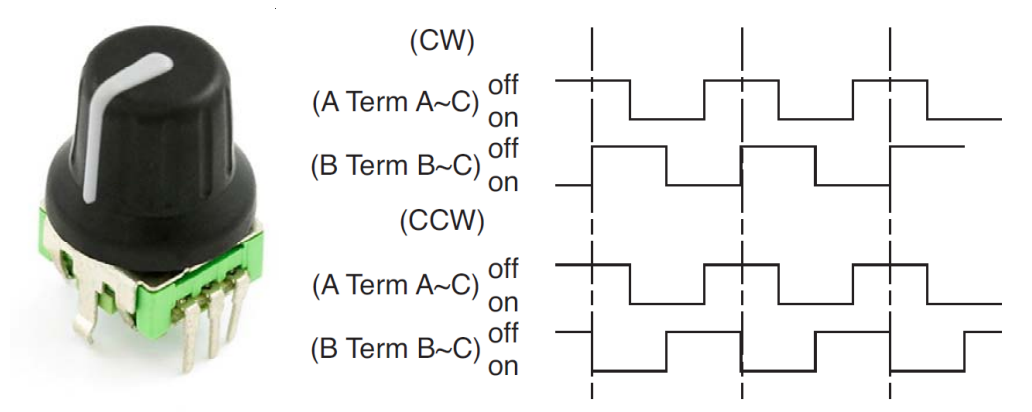
Menynavigasjon

Navigasjon i et menysystem på LCD-skjermen kan løses med knapper eller joystick, men en mer intuitiv navigasjon med et hjul ble tidlig satt opp som beste alternativ. En kvadratur-enkoder med 12 skritt per omdreining og merkbare taktile trinn ble valgt som navigasjonsinput, se Figur 31. Ved å ha en enkoder med taktil feedback til brukeren blir det lettere å navigere et visst antall skritt i menyen. Enkoderen kan i tillegg trykkes inn som en knapp. Kvadratur-enkoderen har tre pinner som må betjenes av mikrokontrolleren i dashbordmodulen, to stykker er enkoder-linjene og den tredje er til knappetrykk.

Knapper og lysdioder

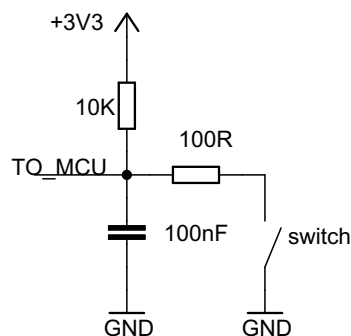
For å ta imot input fra knapper og navigasjonshjul og drive lysdioder ble samme type io-expander som er i frontmodulen benyttet. Ved å bruke to stykker gir det opptil 32 linjer til knapper, lysdioder og navigasjonshjul. Io-expanderen gjør det også mulig å gi et interrupt til mikrokontrolleren dersom en eller flere knapper blir trykket. Dette bidrar til ryddig og effektiv kode.

For å begrense fenomenet “button-bounce” ble knapper koblet til io-expanderen via et



Figur 31: Kvadratur-enkoderen som er benyttet til meny-navigering. Det faseforskjøvede utgangssignalet er vist til høyre, rotasjon med klokken øverst, og mot klokken nederst.

RC-filter som vist i Figur 32. Button-bounce kommer av at knapper ikke lager et rent av/på signal i det de trykkes inn/ut, signalet vil slå seg raskt av og på akkurat i det kontaktflatene møtes. Verdiene til motstander og kondensator er hentet fra samme type filter som ble benyttet i styrings-modulen i 2009, [12].

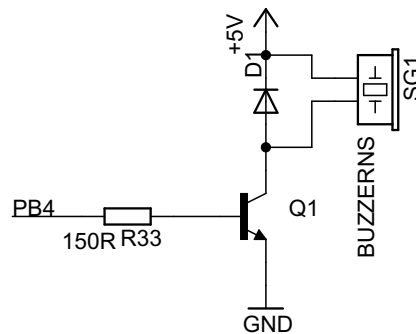


Figur 32: Et RC-filter er benyttet for å dempe button-bounce som kommer av at knapper ikke lager et rent av/på signal i det de trykkes inn/ut.

Lydsignal

For å lage et lydalarmsignal til sjåføren ble en liten og enkel pipetonehøytaler benyttet. Det finnes to typer små høyttalere for å lage pipelyd, den ene fungerer som en vanlig høyttaler med spole og magnet. Den andre benytter piezo-elektrisk effekt. En vanlig høyttaler-type ble benyttet for å få litt større frekvensomfang og dermed ha mulighet til å lage tikkelyder for blinklys og lignende, ikke bare høye pipetoner. Høytaleren som ble valgt har høyest volum ved pipetoner rundt 2kHz. Om høyttaleren er hørbar ved lavere frekvenser og tikkelyder

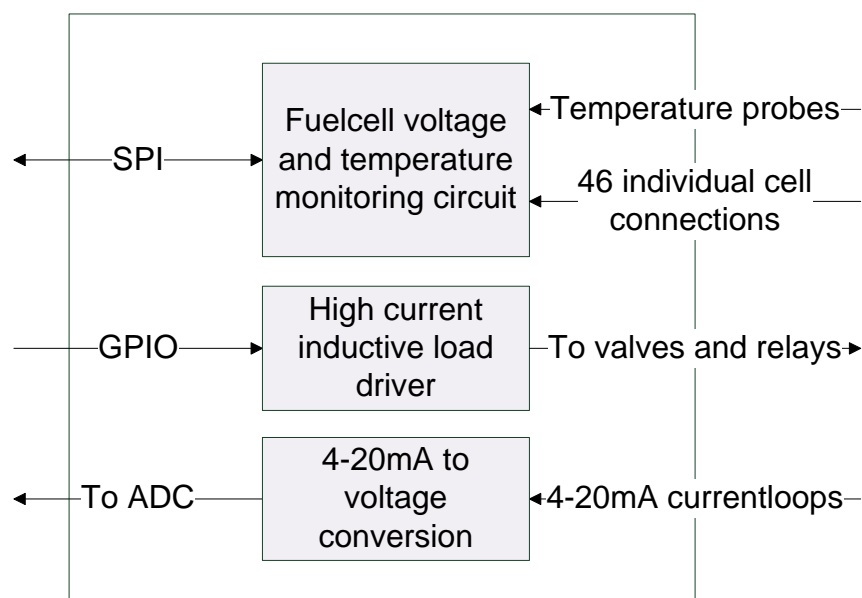
må testes etter montering i bilen. Høyttaleren er koblet til mikrokontrolleren som vist i Figur 33.



Figur 33: Skjema for tilkobling av buzzer-høyttaler med transistor og flyback-diode.

4.6 Brenselcellemodul hardware

Brenselcellemodulen tar seg av en rekke viktige funksjoner for brenselcellestyringen. Den mest omfattende oppgaven er å måle alle cellespenningene. Et overordnet skjema over modulen er gitt i Figur 34.

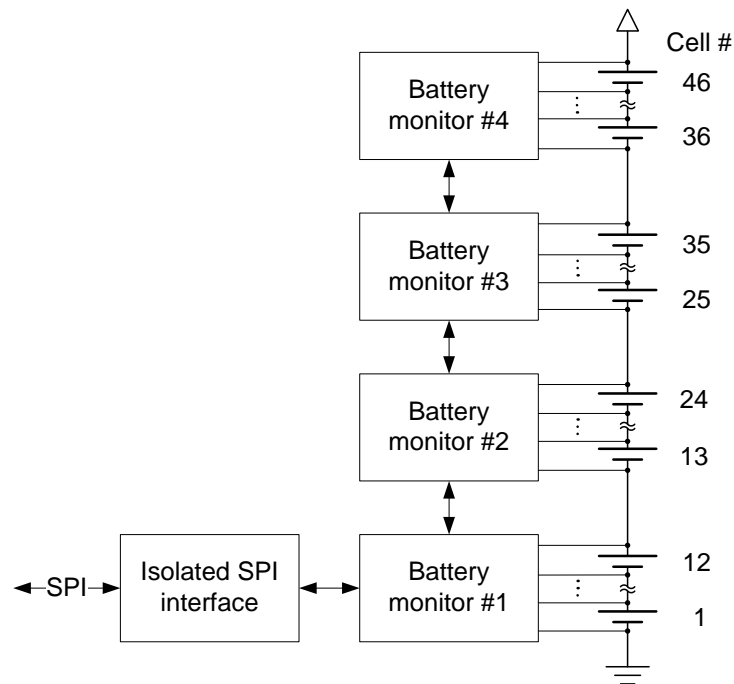


Figur 34: Overordnet skjema for spesialdel i brenselcellemodulen.

Måle cellespenninger

For å måle cellespenningene ble en løsning basert på en ferdig batterimonitorkrets valgt. Brenselcellestacken er elektrisk sett som et vanlig batteri med 46 celler koblet i serie. Kretsen som ble valgt er en LTC6802-1 fra Linear Technology. Den har inngang for 12 celler, hvor hver celle kan ha en spenning opp til 5V. Kretsen er basert på SPI-grensesnitt og flere kretser kan kobles i serie for å måle spenning på stacker med flere enn 12 celler. En seriekobling med fire kretser som er benyttet til brenselcellestacken i bilen er illustrert i Figur 35.

For å redusere støy på målingene er et enkelt lavpassfilter implementert for hver celle. 46 celler i serie krever 4 batterimonitorkretser hvor to måler på 12 celler og de to andre måler på 11 celler, en kan se av Figur 35 at krets 3 og 4 er koblet til 11 celler. Målekretsene tar strøm rett fra batteriet, dette gjør en separat strømforsyning overflødig og forenkler galvanisk isolering. Siden målekretsene tar strøm rett fra stacken og LTC6802-kretsene krever minst 4V forsyningsspenning, må spenningen over hver celle i stacken være minst 0.37V. 0.37V gir 4.07 V over 11 celler og 4.44 V over 12 celler. Spenningen er forventet å ligge over 0.5V ved normal drift. Fysisk kobling mellom målekretsene og brenselcellene er beskrevet i vedlegg F.



Figur 35: Fire batterimonitorkretser koblet i serie med isolert SPI-interface. Krets 1 og 2 betjener 12 celler hver, krets 3 og 4 er koblet til 11 celler hver.

Temperaturmåling

En av fordelene med batterimonitorkretsen som er valgt er at den har mulighet for å koble til to resistive temperatur-sensorer, NTC-motstander. Dette gjør at de fire kretsene tilsammen kan måle 8 forskjellige temperaturer i motorrommet. Temperaturmålingen foregår ved å koble en NTC-motstand fra 3.0V i serie med en vanlig motstand ned til jord og måle spenningen i midten av spenningsdelingen. Spenningen måles av samme 12 bit analog til digitalkonverter som måler cellespenninger.

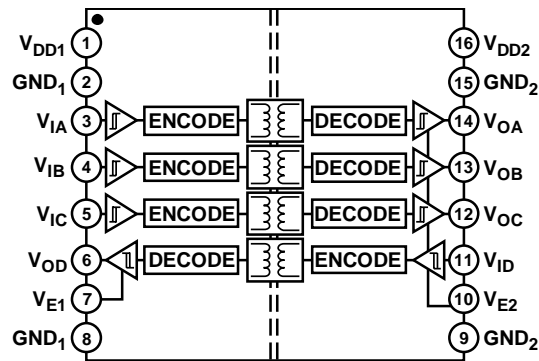
Galvanisk skille

Ettersom styresystemet er drevet av et batteri i bilen vil et galvanisk skille mellom brenselcelle og styresystem sannsynligvis ikke være nødvendig når stacken er montert i bilen. Men brenselcellemodulen er også planlagt benyttet på lab under test og utvikling av brenselcellesystemet. I labsammenheng er det naturlig å koble styringssystemet til en pc med foreksempel USB-tilkobling, samtidig som styresystemet er koblet til brenselcellen. Brenselcellen vil igjen være koblet til en programmerbar last. Brenselcellemodulen bør derfor isoleres galvanisk for å unngå jordsløyfe mellom PC'en og last-enheten.

Galvanisk skille er løst ved at målekretsene som er direkte koblet til stacken tar sin strøm rett fra stacken og kommuniserer med mikrokontrolleren over SPI som er isolert med en 3+1 kanals digital isolator, se Figur 36. Kretsen som ble valgt for denne jobben er en ADUM1401 fra Analog Devices [13]. Den isolerte siden av adum1401 drives av 5V spenningsregulatoren i den første batterimonitorkretsen. Denne kan maksimalt levere 4mA ved 5V, dette utelukket optiske isolatorer som krever mer strøm ved hastigheter opp mot 1Mb/s. Kretsen fra Analog Devices er basert på induktiv kobling og har ved 1Mb/s lavt nok strømforbruk til å drives direkte fra 5V regulatoren i batterimonitorkretsen. Dette gir en mye enklere kretsløsning enn optokoblingskretsen med egen isolert strømforsyning som er foreslått i databladet for batterimonitorkretsen [6].

Måle hydrogentrykk

Trykkmålerne som ble var i bilen i fjor er planlagt brukt igjen. Et interface for 4-20mA grensesnitt er derfor nødvendig. Strøm til sløyfen kan tas direkte fra 24V forsyningen. Ved å benytte analog til digitalkonverteren i mikrokontrolleren kan strømmen i sløyfa måles ved å måle spenningen over en 150Ω motstand. En 150Ω vil gi 3V ved 20mA og passer derfor bra for 3.3V referansen i mikrokontrolleren. Det er lagt opp til 3 strømsløyfemålinger på brenselcellekortet, to for trykkmålere og en ekstra i tilfelle det dukker opp ønsker om andre målinger.



Figur 36: En 3+1 kanals digital isolator krets fra Analog Devices. SPI-kommunikasjonen har klokke-, data- og chipselect-linje som går mot høyre i figuren, og en data-linje tilbake som går til venstre nederst i figuren. (Analog Devices)

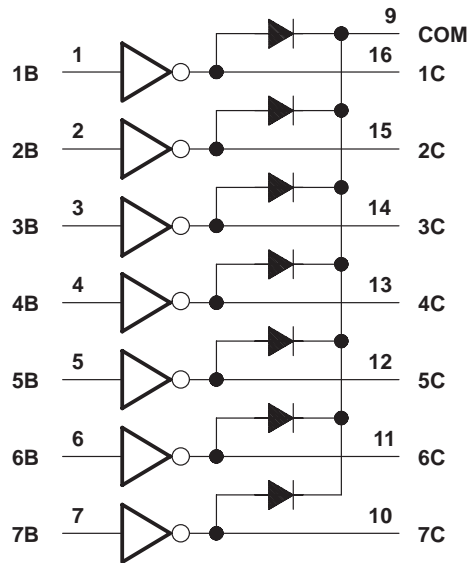
Styre ventiler og releer

Ventiler og releer er induktive laster som trekker mer strøm enn mikrokontrolleren kan levere. For å beskytte transistorbryteren må en flyback-diode monteres over lasten. Vanlige små NPN-transistorer kan ofte lede flere hundre milliampere, og passer derfor bra som bryter.

Det finnes ferdig små kretser som inneholder både transistor og flyback-beskyttelsesdioder innebygget. En slik krets er ULN2003AD fra Texas Instruments, gjengitt i Figur 37. De inverterte driverne er darlington-transistorkoblinger og fungerer som brytere til jord som er lukket når inngangen fra mikrokontrolleren er høy. Hver transistor-driver kan trekke opp til 500mA til jord. Ved å koble COM-terminalen til +24V vil transistorene bli beskyttet mot høy spenning fra den induktive lasten som oppstår i det transistorene slutter å lede strøm.

4.7 Motormodul hardware

Motormodulen som sitter i motorrommet skal ta seg av alle oppgaver utenom brenselcellestyring. Det vil si at når bilen går på batteri istedenfor hydrogen er det kun denne modulen igjen i motorrommet. Motormodulen skal sende kommandoer til motorstyringen fra Smart Motor, drive røde baklys, bremselys og blinklys og drive et lastrelé. Lysdioder for å vise batterispenning på 24V-batteriet er også lagt til. Et overordnet diagram for motormodulen er gjengitt i Figur 38.



Figur 37: Driverkretser i form av darlington-transistorkoblinger og beskyttelsesdioder bygget inn i en ULN2003 fra Texas Instruments. (Texas Instruments)

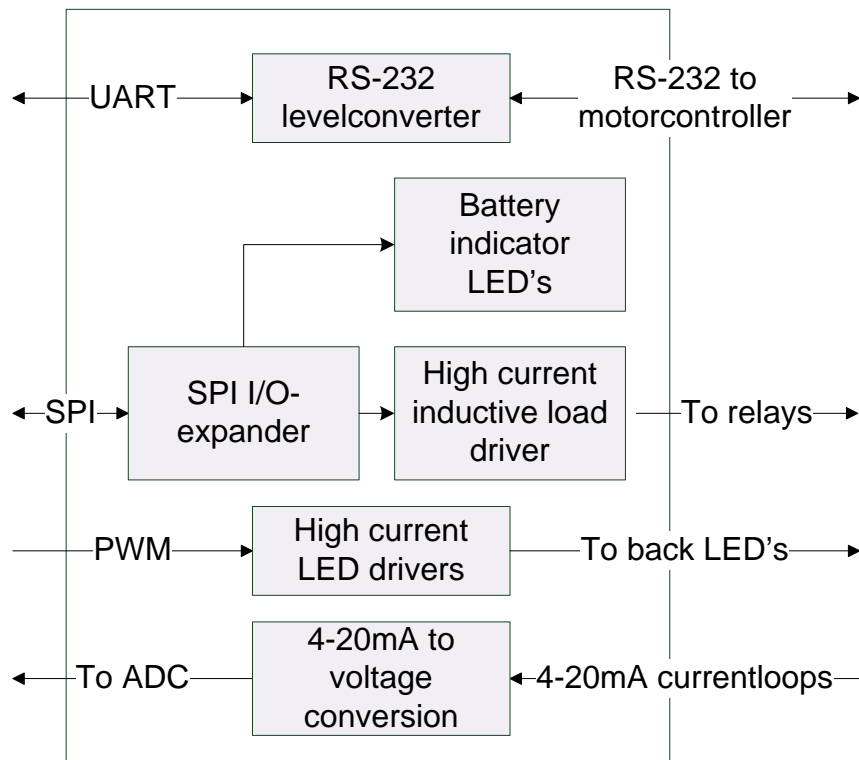
Drive lysdioder

Lysdiodene for baklys er samme type som for frontlys, og lysdiodedriverene i motormodulen er derfor de samme. For baklysene er det ekstra nyttig at lysstyrken kan reguleres, dermed kan samme lysdioder brukes som både bremselys og konstant lysende røde baklys. Fire lysdiodedriverer holder for både to blinklys, separat bremselys og baklys.

RS-232 kommunikasjon med motorstyring

Motorstyringen fra Smart Motor har et vanlig RS-232 interface med spenningsnivåer på $\pm 15V$, for at mikrokontrolleren skal kommunisere med motorstyringen må derfor spenningsnivået på UART'en endres. Det finnes egne RS-232 driverkretser for å konvertere spenningsnivået. Driverkretsen som er valgt er en MAX3232E fra Texas Instruments. Den støtter 3-5V forsyningsspenning og passer dermed bra for 3.3V inngangssignal fra mikrokontrolleren. Maksimal hastighet er 250kbit/s som er mer enn 115.2kbit/s som motorstyringen benytter.

Et problem med hastigheten på 115.2kbit/s er at dette ikke passer perfekt med mikrokontrolleren sin klokkefrekvens på 8 MHz. I følge databladet til AT90CAN-mikrokontrolleren vil det bli 3.5% avvik fra 115.2kbit/s, noe som kan være nok til å gi ustabil kommunikasjon [14]. Dette avviket kommer av at 8 MHz ikke kan divideres ned til nøyaktig riktig hastighet i mikrokontrolleren. Hvis dette viser seg å være et problem kan hastigheten senkes til



Figur 38: Overordnet skjema for spesialdel i motormodulen.

9600bit/s for eksempel, men da må firmware i motorstyringen også endres.

Siden motorstyringen er den samme som ble brukt året før, er den pakkebaserte kommunikasjonsprotokollen også den samme. Protokollen har sjekksum for å detektere feil i mottatte pakker. C-kode for denne protokollen ble laget i 2009, og samme kode er benyttet dette året, se [12].

Måle strøm

En strømmåleløsning som ikke må kobles i serie med motor og brenselcelle, men som er basert på en spole rundt ledningen er ønskelig. For å måle strømmen som går til motoren er det planlagt å bruke en strømtransducer for 4-20mA strømsløyfe. Strøm kan da måles på samme måte som hydrogentrykk måles av brenselcellemodulen. Strømtransducere som ble kjøpt inn viste seg dessverre å kun ha mulighet for å måle vekselstrøm og kunne derfor ikke brukes, se datablad [15]. Det ble dessverre ikke tid til å skaffe en strømmåler for likestrøm.

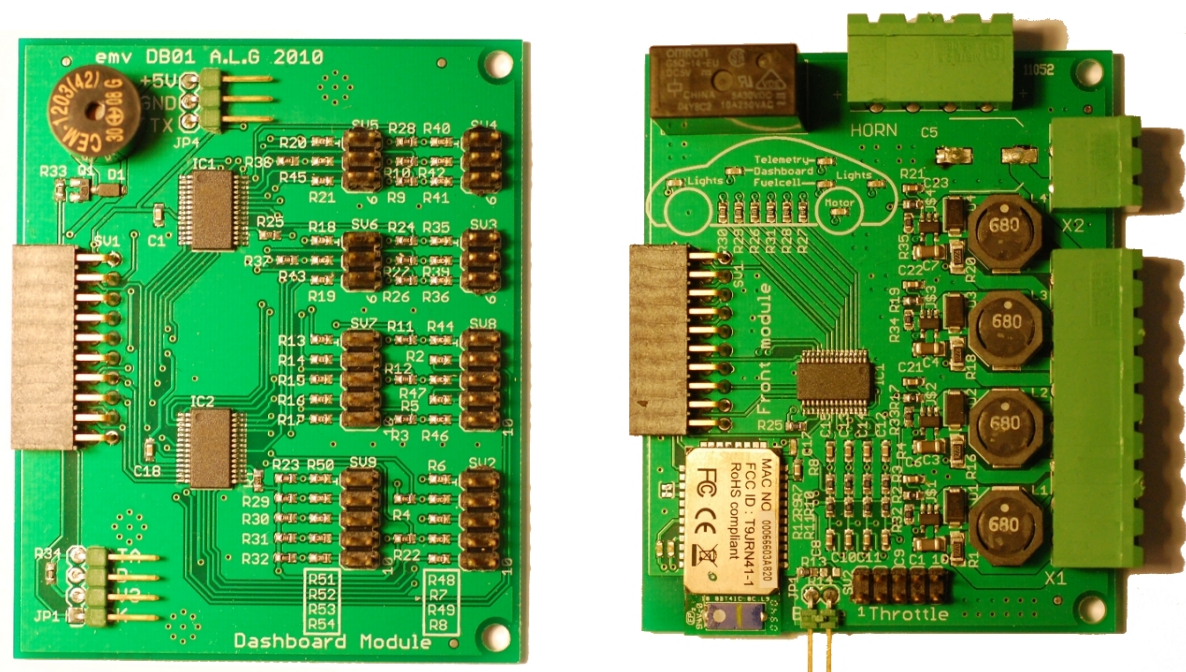
Drive last-releet

For å styre last-releet når bilen går på batteridrift er den samme darlington-transistorkretsen som ble benyttet i brenselcellemodulen benyttet. På grunn av mangel på vanlige io-linjer ble en io-expander benyttet i motormodulen. Io-expanderen er den samme som er benyttet i andre moduler i systemet.

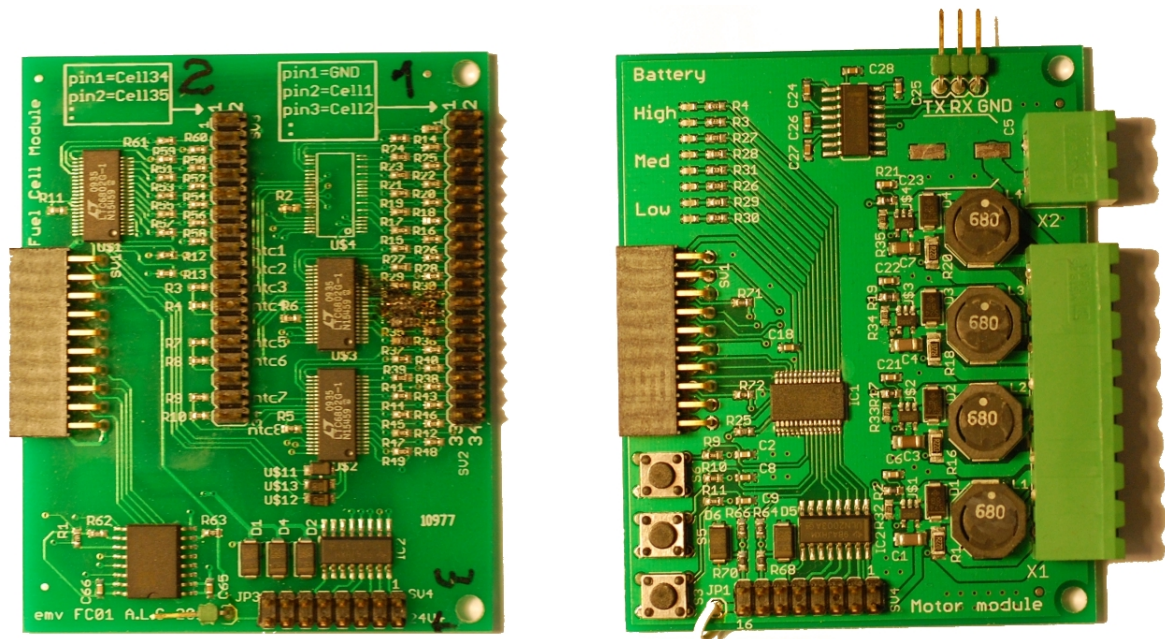
4.8 Implementasjon av hardware

Etter at kretsløsningen for hver spesialmodul var laget ble kretskortutlegg tegnet og klargjort for produksjon. Her er det viktig å være nøye med mål på boks, kontakter og lignende slik at alt passer sammen til slutt. Kretskortene ble profesjonelt produsert på grunn av tidligere dårlige erfaringer med hjemmeetsede kort og fuktighetsproblemer. I vedlegg E er det satt opp en sjekklister som bør gjennomgås før en sender utlegget til produksjon.

Etter at kretskortene kommer tilbake fra produksjon må komponenter loddes på. Etersom det nesten utelukkende er overflatemonterte komponenter, ble mesteparten av loddingen gjort ved hjelp av varmluft og loddepasta. Først blir loddepasta plassert med en sprøyte på alle plasser komponenter skal stå, deretter blir komponentene plassert forsiktig på pastaen med pinsett. Til slutt blir kortet varmet opp med varmluft til loddetinnet smelter. Bilder av ferdige kretskort er gitt i Figur 39 og 40.



Figur 39: Ferdige kretskort for dashboardmodulen til venstre og frontmodulen til høyre.



Figur 40: Ferdige kretskort for brenselcellemodulen til venstre og motormodulen til høyre.

4.9 Sikkerhetssystem

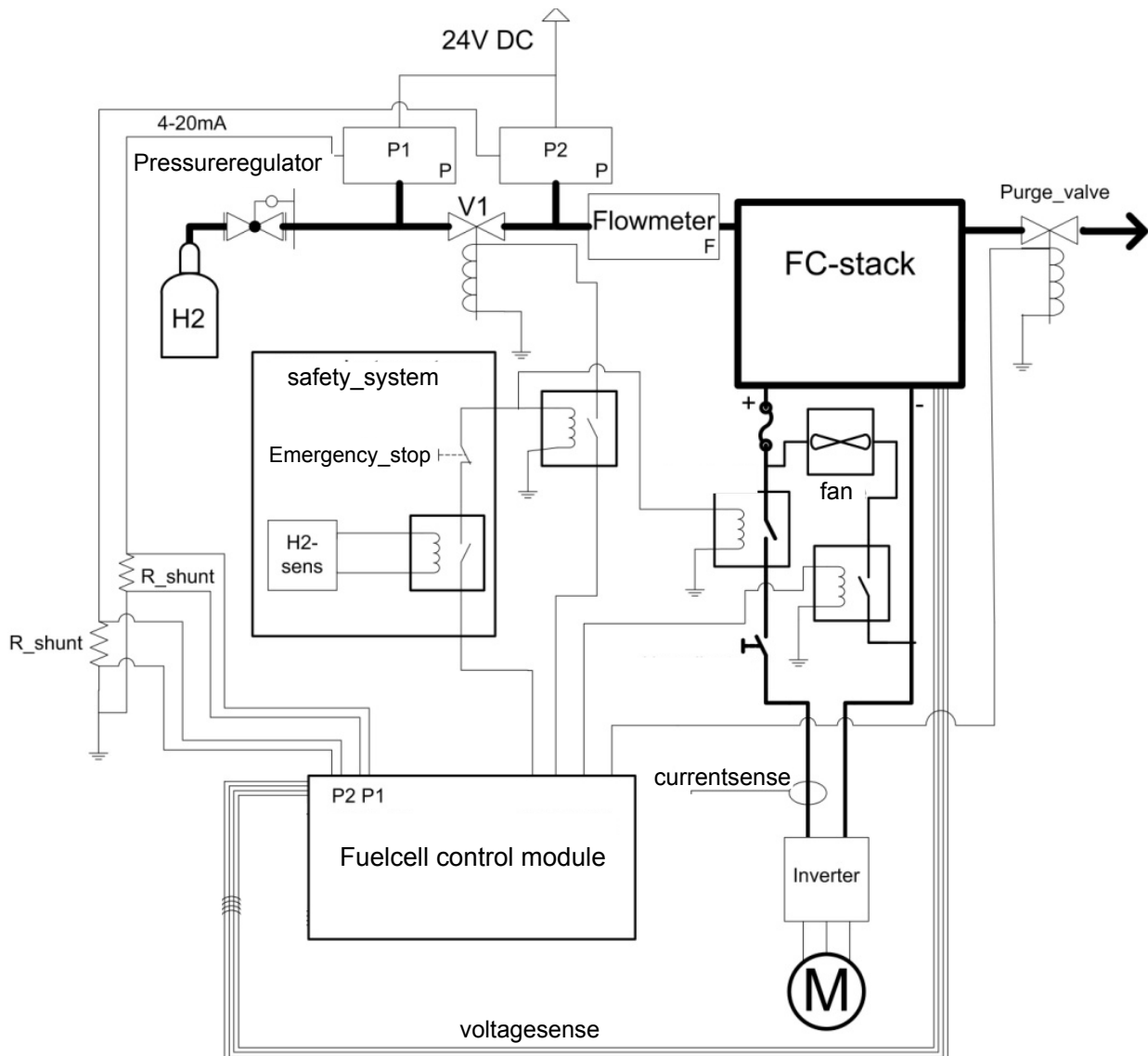
Sikkerhetssystemet som ble implementert i 2009 er gjengitt i Figur 41. Dette systemet fungerte bra under testing og oppfylder kravene i reglementet. Det er et relativt enkelt design og krever ingen endringer selv om brenselcellestyringen blir byttet ut.

Sikkerhetssystemet baserer seg på releer og ventiler som er i en “trygg” posisjon ved strømbrudd. Det betyr at hydrogenventilen er normalt lukket, og elektrisk hovedrelé er normalt åpent. Slike releer og ventiler er dessuten et krav ifølge reglene. Ved å la styresstrømmen til sikkerhetsventil og relé gå gjennom nødstoppbryteren og hydrogensensoren vil systemet bli slått av dersom en av bryterne kutter strømmen. Nødstoppbryteren på taket av bilen er hovedstrømbryter og kutter batteriforsyningen til hele systemet.

4.10 Viftestyring for brenselcellevifte

Alle hjelpesystemer til brenselcellestaken som ikke er direkte sikkerhetsrelatert må drives av strøm fra brenselcellestaken, dette er et krav i reglene. Brenselcellen i bilen har kun en 12V kjølevifte som havner i hjelpesystem-kategorien. For å drive viften ble en DC-DC-omformer-modul fra Traco benyttet, se Figur 42.

Inngangsspenningen er 18-75V og dekker dermed godt og vel området mellom 20V og 50V som staken opererer i, se datablad [16] for detaljer. For å eliminere tomgangsstrøm i DC-DC



Figur 41: Hydrogentilførsel og sikkerhetssystem. Hydrogentilførselen er gjengitt øverst med flowmeter, ventiler og trykksensorer. Sikkerhetsstrømsløyfen er i midten av skjemaet gjennom H₂ sensoren og nødstopppknappen, releene er normalt åpne, H₂ sensorreleet åpnes ved gassdeteksjon.

konverteren ble et relé benyttet for å kutte spenningsforsyningen fra brenselcellestaken til DC-DC omformeren. Dette releet kan også pulsbredde-moduleres for å styre hastigheten på vifta. Releet er spesielt valgt for å tåle mange svitsjesykler, opp til 10 millioner sykler [17].



Figur 42: Traco DC-DC omformer med 18-75V inngang og 12V utgang. Omformeren tar strøm fra brenselcellen og er spenningsforsyning for kjøleviften til brenselcellestacken.

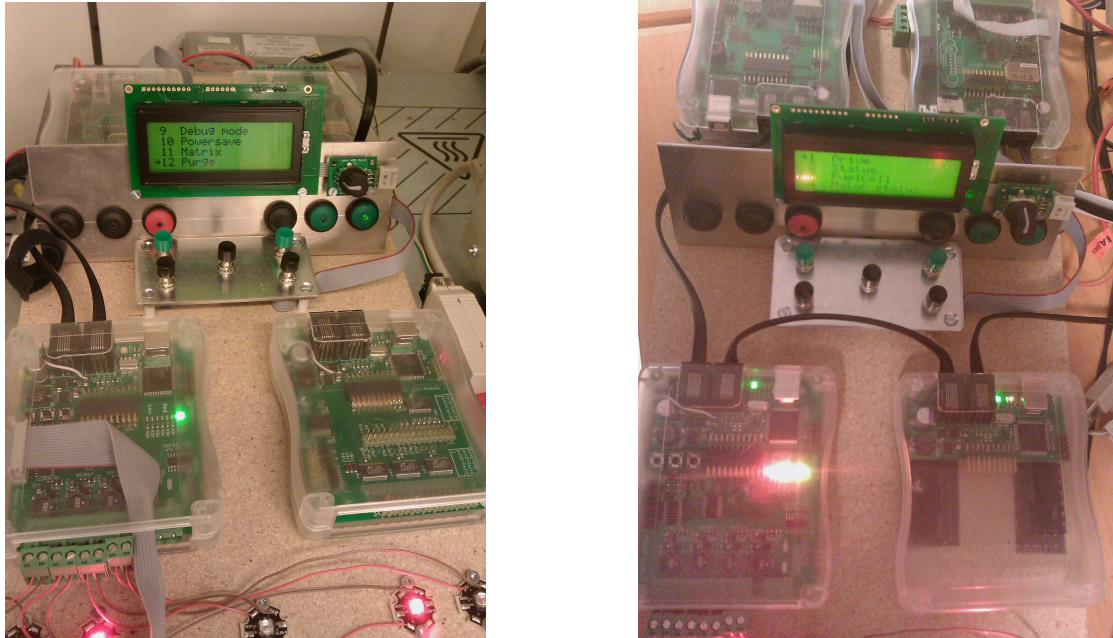
4.11 Testplan

For å verifisere at alle funksjonene på modulene fungerer som de skal bør alle små funksjoner først testes hver for seg. Ved å koble hver enkelt modul til en pc via USB-serieport-tilkoblingen kan testing av enkeltfunksjoner foretas før alle moduler er ferdig. Før systemet kobles sammen og kommunikasjonen skal foregå på CAN-bus er det viktig å være kjent med enkeltfunksjonene og være sikker på at enkeltmodulene fungerer som de skal. De viktigste delene som er planlagt testet alene på denne måten er listet opp i tabell 2.

- Lysdiodedriverne og PWM-modulering
- Skjerm, navigasjonshjulenkoder og knapper
- Gasspedal-enkoder
- Signalhornrelé
- Brenselcellemodul med målesystem
- CAN-bus kommunikasjon med PC
- RS-232 kommunikasjon med motorstyring

Tabell 2: Liste over delfunksjonalitet og moduler som kan testes hver for seg.

Hele instrumenteringssystemet med alle lysdioder, skjerm og knapper er planlagt montert på en treplate for å lette softwareutvikling for systemet og testing av hele systemet. Dette gjør at systemet enkelt kan plasseres på en kontorpult uten løse ledninger og løse komponenter. Bilder av dette oppsettet er vist i Figur 43.



Figur 43: Ferdige moduler montert på treplate for softwareutvikling. Skjerm og knapper er montert der de er tenkt plassert i bilen. Lysdioder for frontlys, blinklys og bremselys er også montert.

4.12 Sammenkobling av moduler

Etter sammenkobling av modulene er kommunikasjons-protokollen mellom modulene viktig. For å være sikker på at tidskritiske oppgaver blir utført i tide og for å være sikker på at ekstra funksjoner kan legges til uten problemer må protokollen på CAN-bussen være nøye planlagt. Etter at hver modul er testet og fungerer ved kommunikasjon over USB-serieport er det en relativt enkel oppgave å benytte CAN-bus istedenfor serieport for å sende kommandoer.

Meldingsfilter i AT90CAN-mikrokontroller

Systemet består av 5 moduler som alle skal sende og motta meldinger på CAN-bussen. De fleste av meldingene er ment å mottas av en bestemt modul, men samtidig er det ønskelig å ha mulighet for å sende en melding til flere moduler. AT90CAN-mikrokontrolleren sin CAN-driver har et innebygget meldingsfilter som kan filtrere på meldings-id i hardware. Dette filteret gjør at software kun trenger å ta hånd om meldinger som faktisk er nyttige for hver modul. Hvis alle modulene måtte lagre unna alle meldinger for så å forkaste flesteparten igjen, ville det tatt opp både mye minne og kjøretid på mikrokontrolleren.

En vanlig CAN-melding har en 11 bits meldings-id. Det gir totalt $2^{11} = 2048$ mulige forskjellige meldinger. Filteret i AT90CAN-mikrokontrolleren fungerer ved å bestemme hvilke av de 11 bitene i ID'en som må være enten 0 eller 1, og hvilke som kan være både 0 og 1 for å godta meldingen. Hvordan dette er løst ved hjelp av bit-masker er vist i Figur 44.

```

Full filtering: to accept only ID = 0x317
- ID MSK = 111 1111 1111  b
- ID TAG = 011 0001 0111  b

Partial filtering: to accept ID from 0x310 up to 0x317
- ID MSK = 111 1111 1000  b
- ID TAG = 011 0001 0xxx  b

No filtering: to accept all ID from 0x000 up to 0x7FF
- ID MSK = 000 0000 0000  b
- ID TAG = xxx  xxxx  xxxx  b

```

Figur 44: Bitvis filtrering av CAN-meldinger i hardware. En ener i ID-MSK betyr at det bit'et må være enten 0 eller 1. Hva hvert bit skal være bestemmes av TAG'en. En 0 i masken betyr at bit'ets verdi er likegyldig. (Atmel)

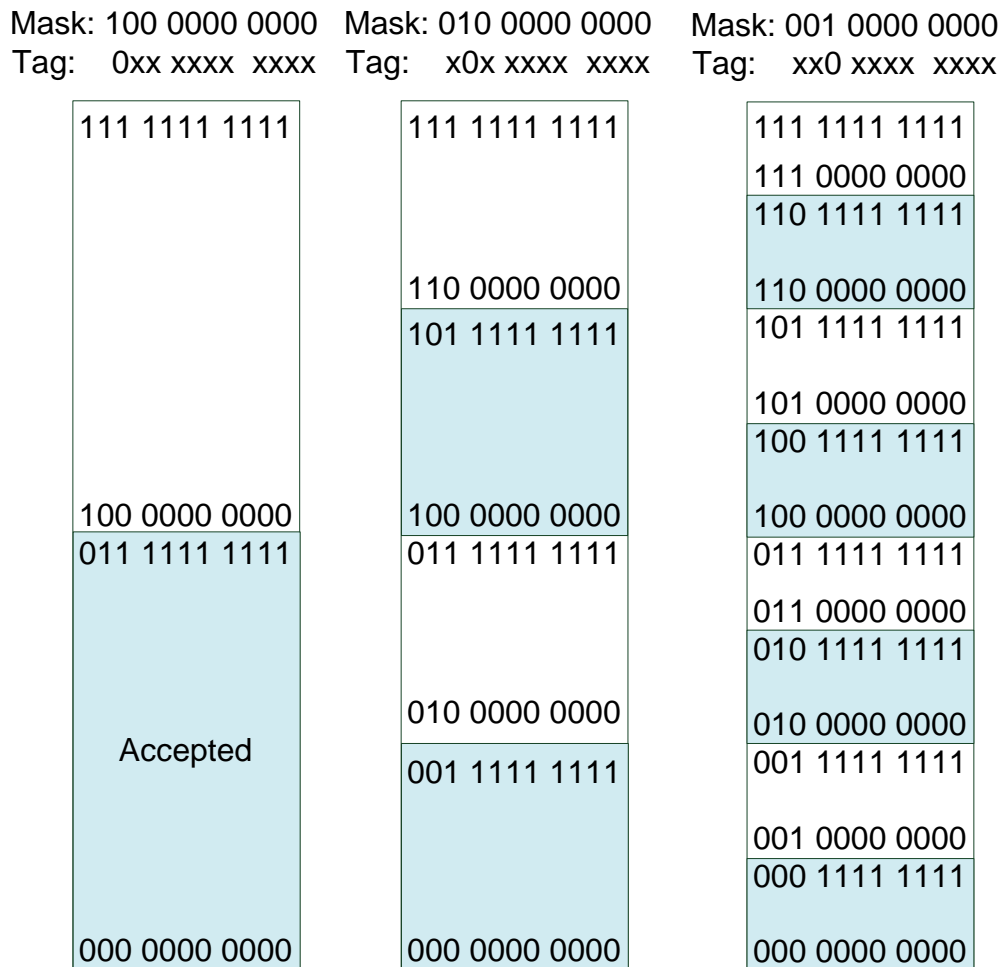
At filtreringen foregår på denne måten gir visse begrensninger i mulighetene for filtrering. Meldings-id'er som skal mottas av en modul kan være vilkårlig mange, men vil være organisert i periodiske grupper. Dette er illustrert i Figur 45.

Filtreringsstrategi

For å oppfylle ønsket om at enkeltmoduler skal ha hver sin del av meldingsrommet og samtidig kunne motta fellesmeldinger må en bestemt filtreringsstrategi velges. Metoden som er valgt utnytter prinsippet med periodiske grupper og gir mulighet til å sende meldinger både til enkeltmoduler og til vilkårlige grupper av moduler. Filtreringen er en utvidelse av filterene illustrert i Figur 45.

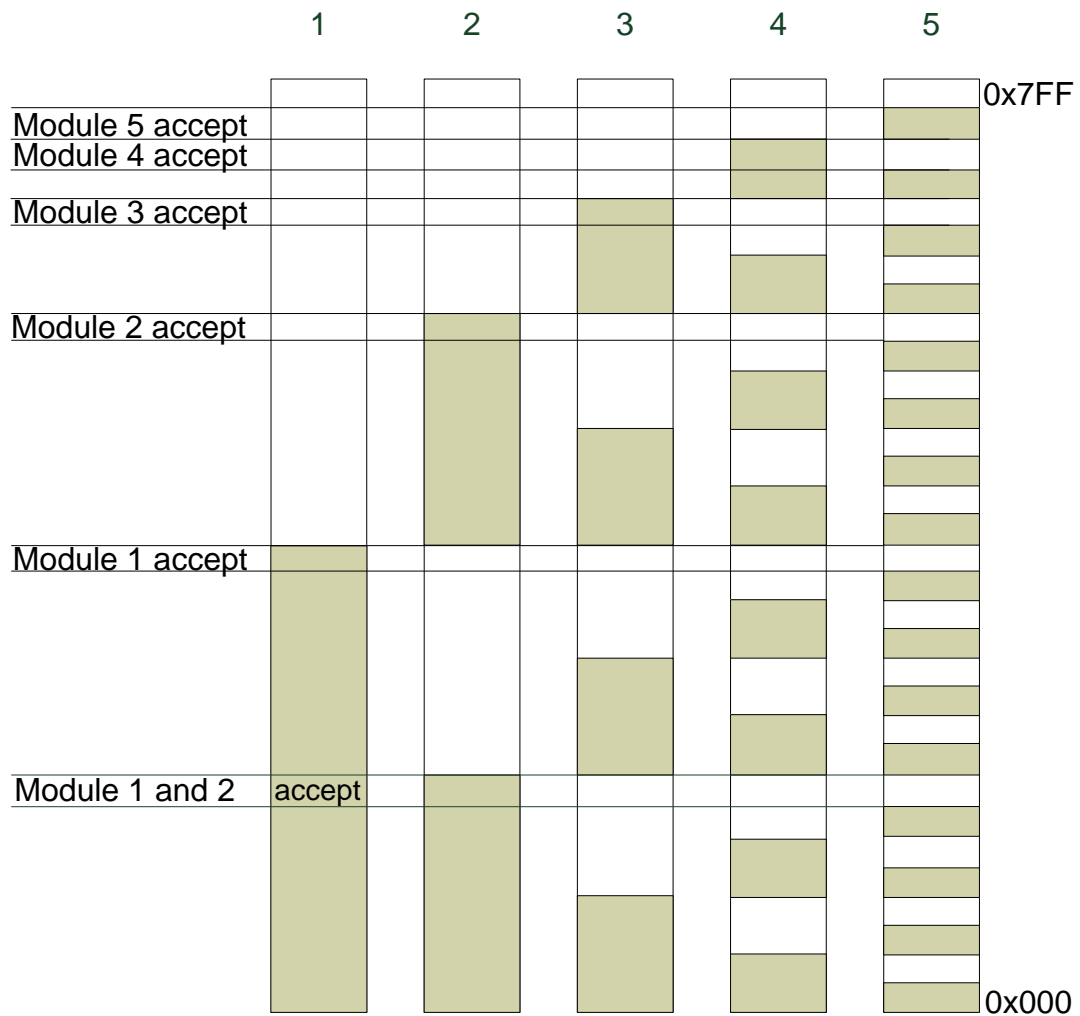
Modul 1 godtar bare meldinger med 0 som mest signifikante bit. Modul 2 godtar kun meldinger med 0 som nest-mest signifikante bit. Videre har modul 3 det tredje mest signifikante bit'et. Hvis en modul nå skal sende en melding til kun én annen modul velges en meldings-id helt øverst i mottakermodulen sitt aksept-område. Dermed vil bare denne modulen motta meldingen. Dersom flere enn en modul skal motta en melding må meldings-id velges lavere i akseptområdet. Dette er illustrert i Figur 46.

Atmel-mikrokontrolleren har mulighet for å definere flere filtere og "innbokser" i hardware, såkalte "Message Objects" [14]. Fordi det er et stort antall meldings-id'er tilgjengelig og det



Figur 45: Filtrering på enkeltbit gir periodiske grupper i mottatte meldinger. Farget område indikerer at meldingen aksepteres.

ikke var behov for mange forskjellige meldings-id'er til hver modul ble metoden beskrevet ovenfor benyttet istedet. Det gir også enklere og mer portabel kode ettersom software ikke er avhengig av spesialfunksjoner i mikrokontrolleren.



Figur 46: Farget område indikerer at meldinger aksepteres. Hver kolonne indikerer hvilke meldinger som godtas av hver modul. Modul 1 godtar alle de nederste meldings-id'ene. Modul 2 godtar id'er i to grupper, modul 3 i 4 grupper og så videre. Ved å sende en melding med id helt øverst i akseptområdet til mottaker vil kun denne modulen motta meldingen.

4.13 Bootloader

For å gjennomføre enkel oppdatering av firmware i modulene er det nødvendig med en bootloader. Konseptet med å overføre ny kode over CAN-bus ble testet og benyttet for én modul under prosjektoppgaven. Målet er å gjøre det enkelt å overføre ny kode til en vilkårlig modul i systemet.

Reset ved CAN-melding

For å resette en modul er det lagt inn en sjekk i mottaksinterrupert for CAN-meldinger som starter watchdog-timeren ved mottak av en bestemt CAN-melding. CAN-meldingene som er valgt for å resette modulene er listet opp i tabell 3.

For at bootloaderkoden skal vite om mikrokontrolleren ble resatt for å laste opp ny kode, eller om den ble resatt av en annen grunn, er eeprom-minnet (Electrically Erasable Programmable Read-Only Memory) benyttet for å huske hvorfor den ble resatt. Dersom det kommer en CAN-reset-melding skrives en bestemt byte til eeprom. For hver gang mikrokontrolleren våkner opp fra reset starter bootladeren med å sjekke eeprom, dersom byten i eeprom er satt til den bestemte verdien starter bootloaderkoden, ellers starter hovedapplikasjonen igjen. Dette er illustrert i Figur 47.

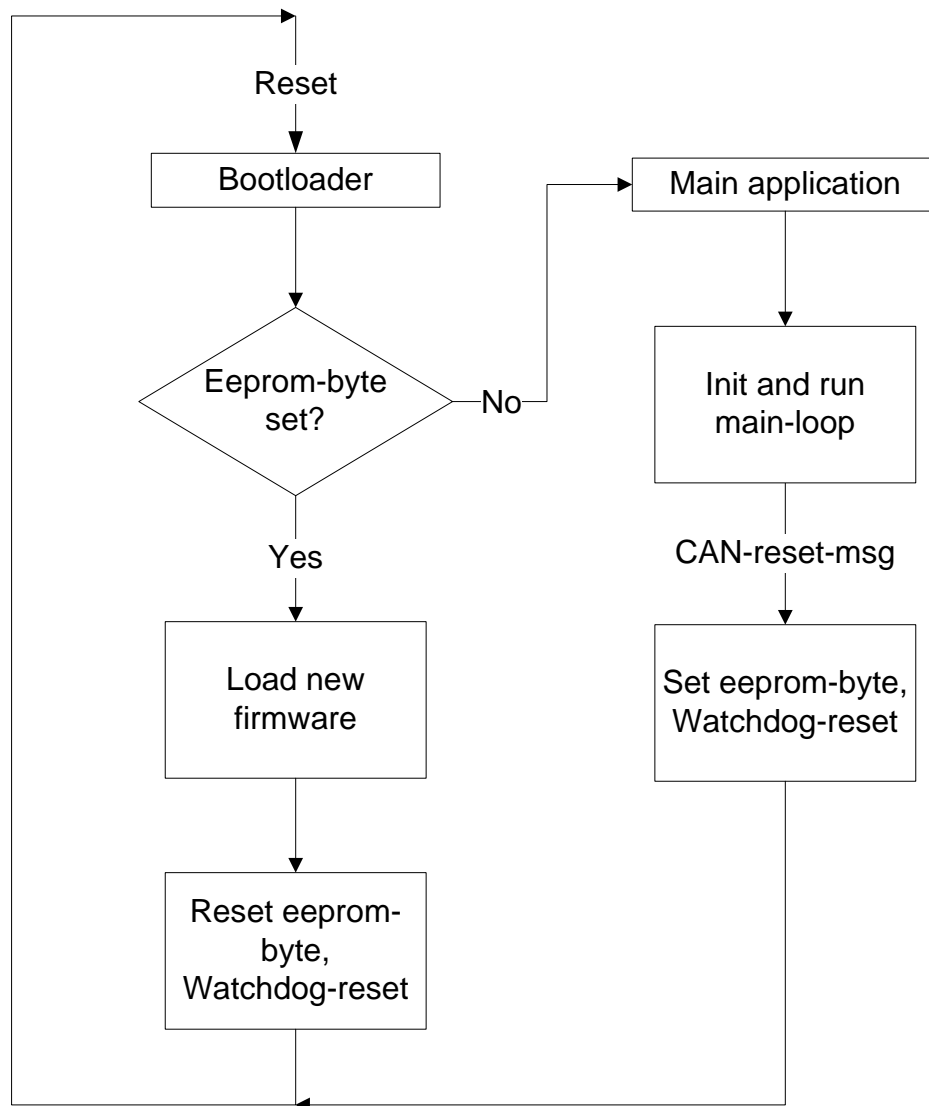
Modul	CAN-ID	Første byte i melding
Brenselcelle	0x00	0x01
Frontmodul	0x00	0x02
Motormodul	0x00	0x03
Dashbord	0x00	0x04
Telemetry	0x00	0x05

Tabell 3: CAN-meldinger for reset av moduler. Ved mottak av en CAN-resetmeldingen vil modulen skrive en bestemt verdi til eeprom og resette seg selv. Deretter vil bootladeren sjekke eeprom og vente på ny firmware-opplasting dersom den bestemte eeprom-byten er satt.

FLIP3 og batch mode

For å laste opp koden er bootloader-rammeverk og en tilhørende Windows-applikasjon fra Atmel kalt FLIP (FLexible In-system Programmer) benyttet. Detaljene rundt bootloader-applikasjonen er beskrevet i manualen for applikasjonen [18].

For å gjøre prosessen helt strømlinjeformet er kommandolinjeversjonen av programmet benyttet. Det gir mulighet for å kjøre en tekstfil med kommandoer. Dermed kan programmet utføre både reset av mikrokontroller, sletting, opplasting av ny kode og verifisering automatisk, for deretter å resette modulen igjen og starte applikasjonen på nytt. Alt ved å kun kjøre en *.bat fil i Windows.



Figur 47: Flytskjema som viser hvordan bootloaderen fungerer ved å sjekke en eeprom-byte ved oppstart. Eeprom-byten avgjør om hovedapplikasjonen skal starte, eller om bootladerkoden skal fortsette å kjøre.

4.14 Generelle softwareteknikker

Software for hver modul består i hovedsak av et sett med drivere for SPI-kommunikasjon, AD-konverter, PWM og lignende, interrupt-rutiner og en overordnet hovedløkke som tar seg av oppgavene til hver modul og håndterer CAN-meldinger. I tillegg kommer et litt mer avansert rammeverk for LCD-skjermdriver og menysystem. Her er det først representert en del praktiske teknikker benyttet i koden, deretter er flytskjema for hver modul presentert.

Interruptbasert input

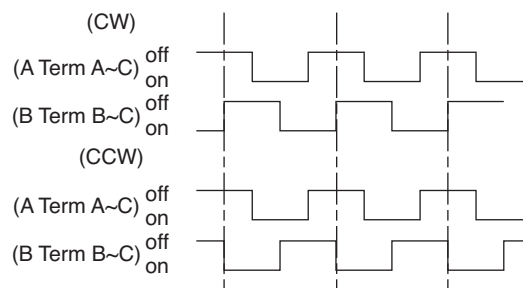
Interruptbasert input er benyttet for at mikrokontrolleren skal reagere på knappetrykk og andre sporadiske hendelser uten å bruke tid på å polle alle input-pinner. MCP23S17 io-expanderen gir et interrupt-signal til mikrokontrolleren dersom en eller flere pinner har endret tilstand siden siste interrupt. Mikrokontrolleren må i interrupt-rutinen spørre io-expanderen hvilke pinner som har endret tilstand, dette lagres unna og kjøring av den vanlige koden gjenopptas.

Hver io-expander gir mulighet for interrupt på alle sine 16 inngangspinner ved å kun benytte en eller to interrupt-pinner på mikrokontrolleren. For å ikke lage problemer i SPI-kommunikasjonen med io-expanderen må interruptet deaktiveres dersom SPI-kommunikasjon finner sted utenom interrupt-rutinen.

Tidskritiske meldinger som skal sendes på CAN-bussen kan sendes rett fra knappetrykk-interruptet. Dette vil være tilfelle for signalthorn og bremselysbryterene. En melding om å skru på hornet eller bremselys sendes i det knappen trykkes inn, og en melding for å skru av sendes i det knappen slippes. For bremselyset sitter trykkbryteren i bremsepedalen og CAN-meldingen går fra frontmodulen.

Kvadraturenkoder

Enkoderen for meny navigasjon gir et pulstog på utgangen for hvert trinn den roteres, dette er gjengitt i Figur 48. Faseforskjellen på pulstoget på pinne A og B bestemmer rotasjonsretning.

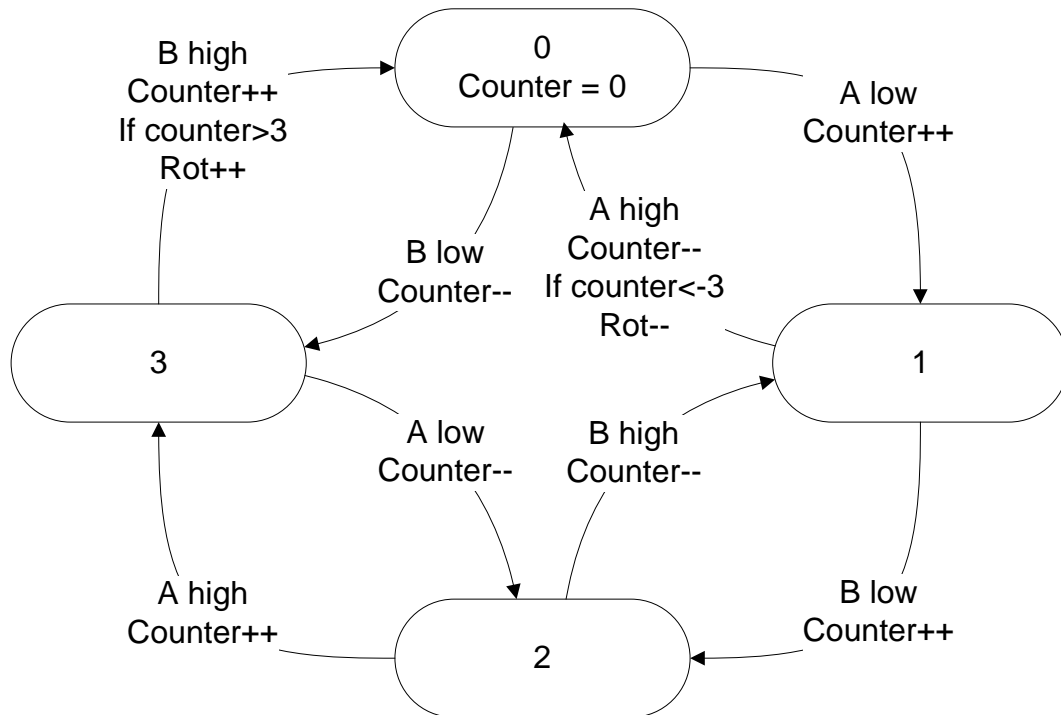


Figur 48: Enkoderen gir et pulstog på utgangen for hvert trinn den roteres. Faseforskjellen på pulstoget på pinne A og B bestemmer rotasjonsretning.

For å dekode pulstoget på pinne A og B til en gyldig rotasjon med eller mot klokka ble en enkel tilstandsmaskin benyttet i interruptrutinen. Hver gang A eller B endrer verdi hopper tilstandsmaskinen til en ny tilstand og oppdaterer en tellevariabel. Det er kun et godkjent rotasjonsklikk dersom tilstandsmaskinen er tilbake til utgangspunktet og har vært gjennom

alle tilstandene i rekkefølge.

Det er bare et fullstendig rotasjonsklikk og dermed et fullstendig pulstog som skal gi input til meny navigasjonen. Dette er for å sikre at ikke meny pekeren hopper flere skritt eller oppfører seg rart når enkoderen for eksempel roteres et halvt klikk og tilbake. Tilstandsmaskinen er gjengitt i Figur 49.



Figur 49: Tilstandsmaskin for kvadraturenkoder. Tilstand 0 er starttilstand, tellevariabelen endres for hver endring på A eller B. Det er kun en fullstendig tur rundt tilstandsmaskinen som endrer rotasjonsvariabelen som angir posisjonen til enkoderen.

Tabelloppslag

Det er hovedsakelig to oppgaver i styringssystemet som egner seg godt for tabelloppslag, den ene er overgangen fra målt spenning over en temperaturfølsom motstand til temperatur, og den andre er overgang fra absolutt-enkoder til gasspådrag. Absolutt-enkoderen baserer seg på en proprietær form for gray-kode som krever tabelloppslag for å få fornuftige verdier. Denne tabellen er oppgitt i databladet for absolutt-enkoderen, se [19].

For temperatursensorene er ligning (3) benyttet til å lage en tabell som regner om fra ADC-verdi til temperatur. T_0 er temperaturen ved 25°C og R_0 er motstanden ved samme temperatur. Beta-verdien er oppgitt i datablad for den spesifikke NTC-motstanden som benyttes.

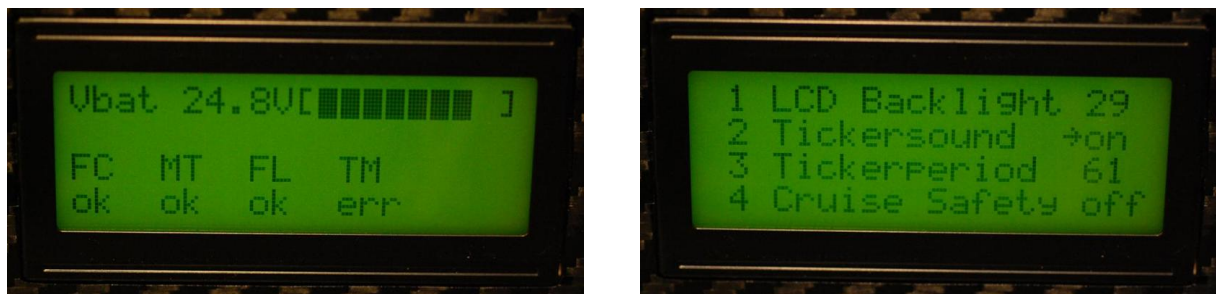
$$\frac{1}{T} = \frac{1}{T_0} + \frac{1}{B} \ln\left(\frac{R}{R_0}\right) \quad (3)$$

Verdiene fra analog til digital-konverteren er for store til å brukes for indeksering av en tabell. Derfor er tabelloppslaget gjort omvendt ved at indeks i tabellen er temperaturen og på hver plass i tabellen er den forventede ADC-verdien for denne temperaturen. Temperaturen finnes ved å traversere tabellen til gjeldene ADC-verdi er funnet og returnere indeksen. Dette er illustrert i tabell 4.

Indeks og temperatur i °C	Forventet ADC-verdi
0	1538
1	1520
2	1501
.	.
.	.
.	.
35	819
36	801
37	782

Tabell 4: Eksempel på oppslagstabell for temperaturomregning. Temperaturen finnes ved å traversere tabellen til gjeldene ADC-verdi er funnet og returnere indeksen.

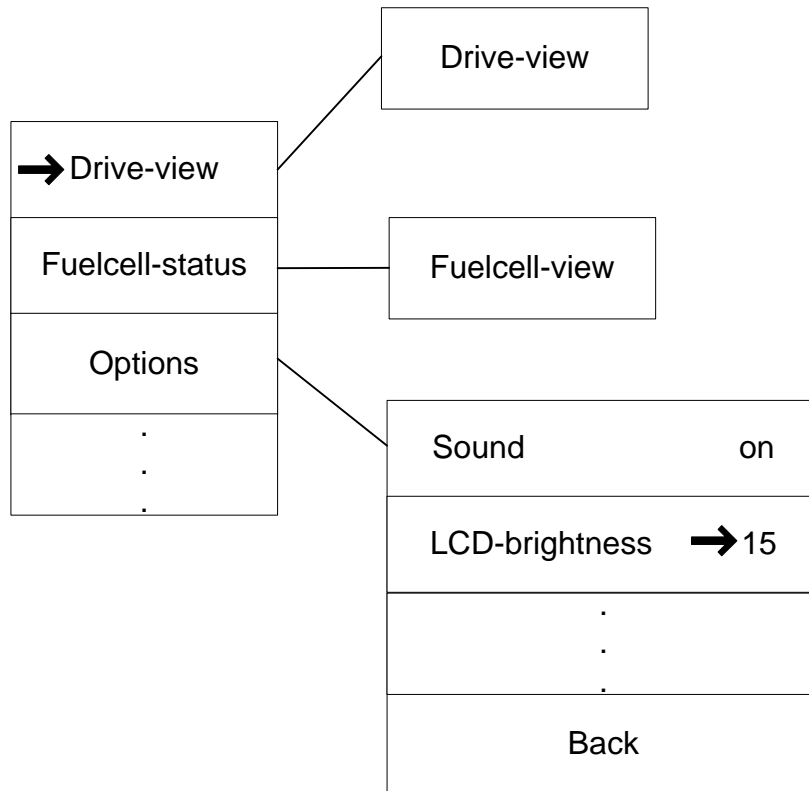
Menysystem for skjerm



Figur 50: Bildet til venstre er en statusvisning av batterispenning og status på de 4 modulene i systemet i tillegg til dashbordmodulen. Bildet til høyre viser en opsjonsmeny med blant annet baklys-innstilling.

Ettersom LCD-skjermen kun kan vise 4 tekstlinjer er menysystemet er lagt opp med 4 menyelementer på skjermen til enhver tid, og en pil på venstresiden som viser hvilken linje som velges dersom navigasjonshjulet trykkes inn. I tillegg til flere forskjellige statusvisninger er det to typer menyer i systemet, overmenyer og opsjonsmenyer.

Ved valg av et element i en overmeny kommer en enten til en ny meny eller til en statusvisning. Ved valg av et element i en opsjonsmeny hopper pila til høyre på skjermen der en verdi eller on/off kan velges ved å vri på hjulet, ett trykk til og pila hopper tilbake til venstresiden. Et trykk på hjulet benyttes også for å gå inn og ut av statusvisninger. Prinsippet for menysystemet er vist i Figur 51.



Figur 51: Prinsippskisse for menysystemet, pila til venstre viser at drive-statusvisningen er valgt. Visningen hopper til drive-view ved å trykke på navigasjonshjulet. Pila til høyre i opsjonsmenyen viser at lysstyrke er valgt, og ved å vri på hjulet endres lysstyrken opp eller ned.

For å gjøre menysystemet enkelt å utvide og konfigurere er all tekst og menyelementer lagt i tabeller. For å legge til eller fjerne menyelementer er det bare å endre tabellen. All tekst for menyen er lagret i flash-minnet i mikrokontrolleren for å spare minneplass, å benytte flashminnet til tekststrenger ble også benyttet i telemetrimodulen, se [1].

Tilstandsmaskinen for menyen er en switch-case som holder orden på hvilken meny som skal tegnes til enhver tid. Plassering av pila tas hånd om av egne draw-metoder for enten hovedmenyen eller om det er en opsjonsmeny. Ved å benytte funksjonspekere støtter menysystemet å ha vilkårlig mange opsjonsmenyer. Se Figur 50 for bilder av meny og statusvisning med batterispenning.

Callback-funksjoner

Mange funksjoner i styringssystemet baserer seg på at ting skal skje i et visst tidsrom. Det kan være alt fra blinklys som skal blinke til ventiler som skal åpnes og så lukkes etter en viss tid. For å håndtere slike oppgaver med nøyaktige tidskrav uten å legge inn venteløkker i koden ble det opprettet et eget timerinterrupt for callback-funksjoner.

Callback-funksjonene realiseres ved bruk av funksjonspekere som legges inn i en liste sammen med parametere som bestemmer hvor ofte de skal kalles. Deretter vil timerinterruptet gå gjennom listen, dekrementere alle funksjonene sin teller og kjøre de funksjonene som står for tur. For å aktivere blinklys kan callback-funksjonen benyttes slik:

```
void IndicatorLights(void)
{
    if(LightOn == 1)
    {
        LightOn = 0;
    }
    else
    {
        LightOn = 1;
    }
}

int main(void)
{
    initTimer0();
    Timer0RegisterCallback(IndicatorLights, INDICATOR_PERIOD);
}
```

CAN-meldinger

Meldingene som går mellom modulene i systemet benytter meldings-id både for å adressere til rett mottaker og for å definere meldingens innhold. Mange av CAN-meldingene benyttes kun for å gi beskjed om at noe må slås på eller av. Første byte i meldingen er da satt til 0 for av eller 1 for på.

Dersom en verdi sendes benyttes bare det nødvendige antallet bytes til dette. Dette gjør at de fleste meldingene blir kortere enn 8 byte, og belastningen på CAN-bussen reduseres. Et viktig prinsipp ved kommunikasjonen på CAN-bussen er at den skal være tilstandsløs, det vil si at betydningen av en melding ikke skal være forskjellig fra gang til gang. Dersom en

melding ikke kommer fram vil det derfor ikke få følgefeil senere. Vedlegg B lister opp alle CAN-meldinger i systemet.

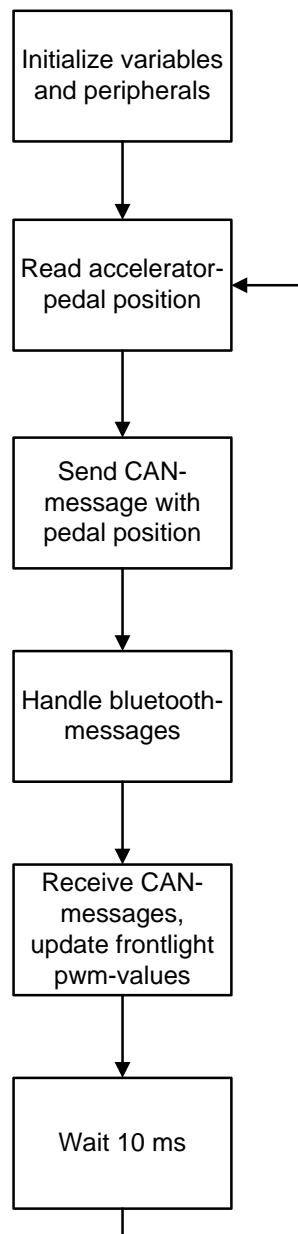
4.15 Software flytskjema

Software for alle modulene er i bygget opp på samme prinsipielle måte. Ved oppstart initialiseres variabler og alle nødvendige hardware-funksjoner som timere og interrupt i mikrokontrolleren initialiseres. Deretter er det én hovedløkke som går gjennom og håndterer alle mottatte CAN-meldinger, oppdaterer sensorverdier, knappetrykk og lignende og sender CAN-meldinger til andre moduler.

Ettersom alle modulene har forskjellig klokke og er helt usynkronisert kan CAN-meldinger komme når som helst. Det er derfor viktig at CAN-meldinger blir lagret unna i ram i det de mottas. Dette er håndtert i CAN-interruptrutinen. Dersom dette interruptet skrus av er det risiko for at CAN-meldinger blir forkastet uten å lagres i ram. En løsning på dette er å ikke skru av alle interrupt, men velge spesielt de interruptene som er nødvendig å skru av. Dette er benyttet i blant annet SPI-kommunikasjonen med io-expanderene.

Frontmodul software

Frontmodulen sitter foran i bilen og håndterer blant annet gasspedal, frontlys og signalhorn. I hovedløkken er de viktigste oppgavene å lese gasspedalposisjonen, sende den til dashbordet og håndtere mottatte CAN-meldinger. Av meldinger som mottas er det bare lysstyrkeverdier for frontlys som mottas i hovedløkken. CAN-melding for signalhornet håndteres i selve CAN-interruptet. Brems pedalbryteren håndteres i io-expanderinterruptet.



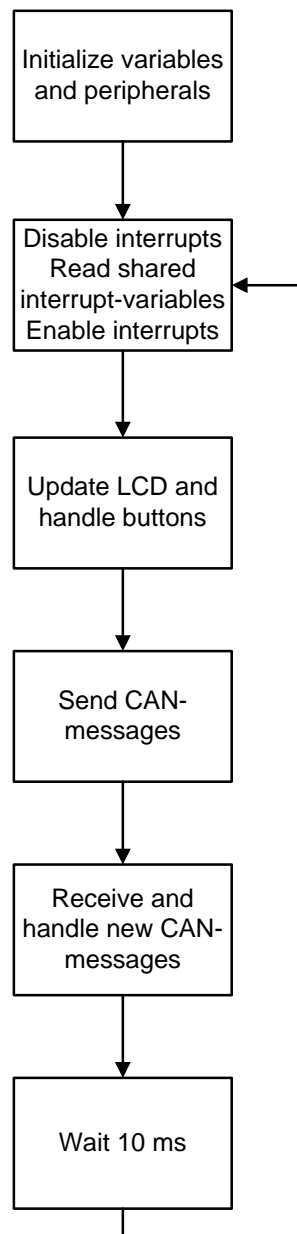
Figur 52: Software for frontmodulen håndterer blant annet avlesing av gasspedalvinkel og styring av frontlys.

Dashbordmodul software

Dashbordmodulen sitter i dashbordet og håndterer menyen på lcd-skjermen, knappetrykk og navigasjonshjulenkoderen. Dashbordmodulen sender også meldinger til de fleste andre modulene i systemet. Knappetrykk og rotasjonsendringer registreres i interruptrutinen for io-expanderen og lagres i delte variabler. De delte variablene blir sjekket og håndtert i

hovedløkken. Hovedløkken kjører ofte nok til at det ikke er noen merkbar forsinkelse fra knappen er trykket til noe skjer. For å unngå at interrupt-rutinen aksesserer 16-bit variablene samtidig som de leses i hovedløkken er lesingen beskyttet ved å skru av interruptet.

Cruise-control for bilen tas også hånd om av dashbordmodulen. Å kalle det cruise-control er litt overdrevet da motorstyringen fra Smart Motor tar hånd om å holde konstant fart. Det eneste som må sendes fra dashbordet er en CAN-melding til motormodulen når hastighetsreferansen endres. Hastighetsreferansen kan endres med knapper på rattet eller med navigasjonshjulet.



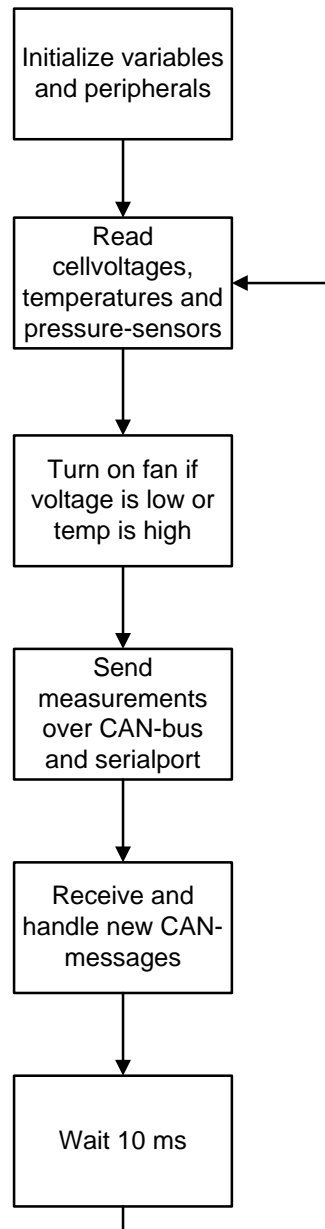
Figur 53: Software for dashbordmodulen håndterer meny på lcd-skjermen, knappetrykk og navigasjonshjulenkoder.

Brenselcellemodul software

Brenselcellemodulen måler alle celledspenninger, temperaturer og trykk og sender måledata over CAN-bus til dashbordmodulen og telemetrimodulen. Ved mottak av oppstartsmelding fra dashbordet åpnes hydrogentilførsel og stacken purges en gang. Dersom totalspenningen på stacken faller for lavt, eller temperaturen blir for høy startes viften. Det er også mulighet

for å sende CAN-melding for manuell purge og viftestart.

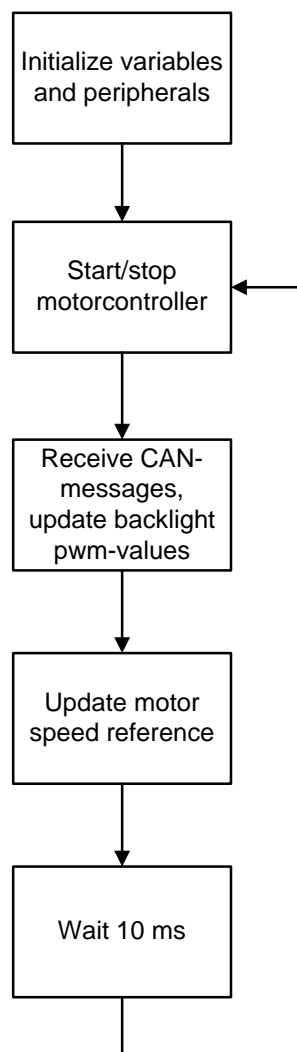
Dersom cellespenninger faller kritisk lavt vil stacken skrus av ved å stenge hydrogentilførselen og åpne last-releet. Software-pulsbreddemodulering er benyttet for å regulere hastigheten på vifta. Dette er løst ved å registrere en callback-funksjon som kjøres hvert 10 millisekund og håndterer pulsbreddemodulering med periodetid på omtrent ett sekund. Eksempel på bruk av callback-funksjoner er gitt i seksjon 4.14.



Figur 54: Brenselcellemodulen måler blant annet spenning, temperatur og trykk i brensel-cellestacken og sender dette til telemetri- og dashboardmodulen.

Motormodul software

Motormodulens hovedoppgaver er å sende motorkommandoer fra dashbordmodulen videre til motorstyringen over RS232 og håndtere baklys. Baklysene fungerer akkurat som frontlysene med pulsbreddemodulert lysstyrke og oppdatering av denne dersom det kommer CAN-melding med nye verdier. I tillegg tar motormodulen imot bremsesignal fra frontmodulen dersom bremsepedalen trykkes inn. Ved bremsing blir lysstyrken i de røde lysdiodene økt fra en lav baklysverdi til en høyere bremselysverdi. I tillegg holder software i motormodulen kontroll på om motorkontrolleren er av eller på, da det ikke bør sendes oppstartskommando hvis motorstyringen allerede er på.



Figur 55: Motormodulens hovedoppgaver er å sende motorkommandoer fra dashbordmodulen videre til motorstyringen over RS232 og håndtere baklys.

4.16 Androidmobil og blåtann

Ettersom frontmodulen fikk blåtann-funksjonalitet og undertegnede har en mobiltelefon med Google sitt mobil-operativsystem Android, ble dette benyttet til å lage en demoapplikasjon for bilen. Eksempelkode for et blåtann-chatprogram fra google ble modifisert til å kunne koble seg til en SPP-blåtannenhet (Serial Port Profile). Protokollen for kommunikasjon er enkel, bokstaver benyttes til å kommunisere. 'a' betyr for eksempel å slå på nærløst og 'b' slå av. Demo-applikasjonen har mulighet til å skru av og på lys, og aktivere hornet.

Modifikasjonen av chat-programmet bestod i å legge til flere knapper i det grafiske grensesnittet og sende bokstaver over blåtann ved knappetrykk på mobilen. Eksempel på registrering av knappe-lytter i java er gitt i kodeeksempel nedenfor. Det er kun enveis kommunikasjon som er implementert. Toveis kommunikasjon er enkelt å legge til da chat-programmet støtter det i utgangspunktet. For å få android til å godta tilkobling med en SPP-enhet måtte UUID-identifikatoren (Universally Unique Identifier) i blåtann-chatprogrammet endres, se [20]. UUID og tilkoblingskode er også gjengitt nedenfor.

```
private static final UUID MY_UUID = UUID.fromString("
    00001101-0000-1000-8000-00805F9B34FB");
.
.
BluetoothSocket tmp = null;
try {
    tmp = device.createRfcommSocketToServiceRecord(MY_UUID);
} catch (IOException e) {
    Log.e(TAG, "createRfcomm_failed", e);
}
.
.
Horn = (Button) findViewById(R.id.hornButton);
Horn.setOnClickListener(new OnClickListener()
{
    public void onClick(View v)
    {
        String message = "i";
        sendMessage(message);
    }
});
```

5 Resultater

Etterhvert som kretskort ble produsert og moduler begynte å bli ferdig ble alle viktige funksjoner testet fortløpende.

5.1 Universal-kretskort

Universalmodulen ble testet med 5-30V inngangsspenning og fungerte bra i hele området. CAN-bus-kommunikasjon med pc'en fungerte også bra, 6 meter ledningslengde og kommunikasjonshastighet på 500kbit/s ble benyttet. USB-serieport funksjonaliteten fungerte bra og JTAG, klokkekrySTALL, status-lysdiode og spenningsmåling med ADC'en fungerte bra, akkurat som på telemetri-modulen. Etter at noen spesialmoduler var utviklet viste det seg at det hadde vært praktisk med en egen 24V-linje lagt opp til spesialmodul-kortene. Dette ble løst med en ekstra ledning som ble loddet på mellom de to kretskortene.

5.2 Frontmodul

Frontmodulen består av et universal-kretskort som ble funnet å fungere bra, og et spesial-kretskort med mange individuelle funksjoner. Softwaredrivere for funksjonene ble utviklet, og funksjonene ble testet en etter en.

Lys

Lysdiodedriverne ble testet med PWM-frekvens på ca. 122Hz, og oppløsning på 256 nivåer. Da dette fungerte veldig bra, og det ikke var mulig å se at lyset blinket ble dette oppsettet beholdt videre uten mer testing. Effekten av å redusere lysstyrken litt ned fra å være kontinuerlig på hadde stor innvirkning på strømforbruket. Tre lysdioder ble også koblet i serie på én driverkrets, dette fungerte uten problemer. Driverkretsene skal tåle å få en kortslutning over utgangen, dette ble testet ved å kortslutte med et ampere-meter. Kretsløsningen tok ikke skade og kortslutningsstrømmen tilsvarte strømmen som gikk med lysdiode koblet til.

Gasspedalenkoder

Absolutt-enkoderen til gasspedalen er koblet til io-expanderen, etter at tabelloppslag-konvertering av vinkelen var implementert fungerte absolutt-enkoderen veldig bra. Vinkelen

som leses av er stabil og repeterbar. Det ble ikke funnet noen uregelmessigheter i overgangen mellom to verdier.

Signalhorn

Et signalhorn i serie med en motstand koblet til 24V viste seg å ikke fungere. To like horn laget for 12V i serie fungerte derimot veldig bra. Signalhorn-releet fungerte bra da det ble testet uten horn koblet til. Da releet ble testet med horn og en 4Ω motstand i serie med hornet på 24V virket det som om releet hang seg opp. Et separat signalhorn-relé av typen som blir benyttet i vanlige biler ble benyttet i stedet. Løsningen med to horn i serie, og et vanlig bil-relé fungerte bra. Å kun benytte releet montert på kretskortet ble ikke testet med to horn i serie.

5.3 Dashbordmodul

Dashbordmodulen håndterer både skjerm, knappetrykk, navigasjonshjul, lydalarm og statuslysdioder i knappene. Det største usikkerhetsmomentet var det enkoder-baserte navigasjonshjulet, det er viktig at det føles responsivt når en navigerer i menyene.

Knapper

Knappene er koblet til en io-expander via et RC-debounce-filter. Interruptbasert input fra io-expanderen fungerte veldig bra. Det var ikke mulig å fremprovosere feil oppførsel ved å trykke fort, eller trykke knappene halvveis inn eller lignende. Verdiene til RC-filteret ble ikke testet nærmere da knappene fungerte bra og uten problemer.

Skjerm

Skjermen ble først testet ved å koble den rett til et terminalprogram på pc'en. Bokstaver og tall kom opp på skjermen akkurat som forventet. Skjermen var veldig godt synlig både i sollys og i mørket med bakgrunnsbelysningen på. Hastigheten ble satt opp til maks hastighet på 38400 bit/s for å kunne sende kommandoer og tegn raskere. Hvis en sender en "clear all"-kommando til skjermen for så å sende samme teksten på nytt fører det til blinkende tekst på skjermen. Løsningen er å kun overskrive det som skal endres av tekst uten å slette noe først.

Navigasjonshjul

Etter at interruptbasert input fungerte bra for knappene, ble tilstandsmaskinen for navigasjonshjulet testet. Kvadraturenkoderen fungerte over all forventning, og ved normale rotasjonshastigheter mistet den ingen klikk. Dersom en vrir fort fram og tilbake hendte det at et og annet klikk ble “oversett”. Men dette har ingen praktisk betydning. Å vri halvveis inn i et klikk, for så å vri tilbake igjen ga ingen respons slik som forventet. Det har ikke blitt oppdaget noen problemer med navigasjons-enkoderen senere.

Lydalarm

Den lille pipe-høytteren, som er spesielt laget for pipetoner rundt 2 kHz, ble testet på mange ulike frekvenser og 2 kHz var absolutt høyest. Det fungerte bra å lage klikke-lyder også, da med korte 100-200Hz signaler. Volumet på klikkelydene var litt for lavt til å være nyttig under kjøring med bilen. Kjørestøy, hjelm og handsfree-headset gjør at sjåføren ikke hører like godt. Pipe-signaler derimot var veldig godt hørbart.

5.4 Brenselcellemodul

Uten en fungerende brenselcellemodul kan ikke brenselcellestaken bli testet ordentlig på lab, det var derfor avgjørende å få den ferdig på et tidlig tidspunkt. Brenselcellemodulen ble benyttet i arbeidet med brenselcellen på Sintef sin lab gjennom hele våren. Dette gjorde at denne modulen ble mye brukt før konkurransen og ble dermed grundig testet.

Spenningsmålinger

Ved første test av batterimonitorkretsene ble de koblet til en enkel spenningsdeler som bestod av 4 stk. 100Ω motstander. Samlet forsyningsspennning var 30V og hver krets fikk dermed 7.5V. Inngangene for individuelle cellespenninger ble ikke koblet til. Kretsene svarte som forventet ved å prøve litt forskjellige SPI-kommandoer. Selv om det ikke var koblet til noen cellespenninger varierte spenningsmålingene mellom 0V og 1V. En interessant observasjon var at spenningsmålingene summerte seg til forsyningsspenningen for hver krets. Årsaken til dette ligger inne i kretsene og dette ble ikke videre undersøkt.

Etter at kretsene fungerte ble brenselcellestaken koblet til. Før det er spenning på stacken får heller ikke målekretsene noen forsyningsspennning og mikrokontrolleren får ikke kommunisert med kretsene. Straks hydrogen ble tilført stacken spratt total spenningen opp til 42V og hver av cellene leste rundt 0.91V. Verdiene så veldig bra ut, og summen av alle måleverdiene

var bare 0.1V fra det multimeteret viste. Resultatet var overraskende bra, og prosentvis feil var 0.23% som er under maksimalfeilen på 0.25% oppgitt i databladet for kretsen.

Etter mer testing ble det funnet et par problemer med målingene. Ved lastendringer på stacken oppførte målingene på celle 45 og 46 seg rart. Celle 45 gikk ned til nesten 0V og celle 46 gikk opp til 2V, summen av de to var fortsatt riktig. Med multimeteret ble hver cellespenning målt til å være 0.9V. Årsaken til dette avviket ble ikke funnet, løsningen ble å se bort fra de to cellespenningsmålingene.

Temperaturmålinger

Temperaturmålingene fungerte bra og var nøyaktige innenfor to grader ved romtemperatur. Måten tabell-oppslaget ble utført på gav bare heltallsverdier for temperaturen. Det ble ikke testet hvor nøyaktig temperaturmålingene er ved andre temperaturer.

Strømmålinger

Strømmåleren som ble kjøpt var dessverre kun laget for vekselspanning. På grunn av dette kunne den ikke benyttes til måling av DC-strømmen fra brenselcellestacken. På vekselstrøm fungerte måleren veldig bra, 4-20mA strømsløyfeløsningen klarte å drive måleren og gi nøyaktige målinger. En egnet 4-20mA strømmåler for DC-strøm ble ikke skaffet i tide til konkurransen og er derfor ikke testet.

Trykkmålinger

Trykkmålerene er de samme som ble brukt året før. 4-20mA strømsløyfen fungerte utmerket for trykkmålerene. Deteksjon av åpen sløyfe fungerte også bra ved at spenningsmålingen falt til 0V. Å kortslutte strømsløyfen ble ikke testet.

Ventil og reléstyring

Ventil og reléstyring med darlington-transistor-driverkretsen fungerte veldig bra. Strømstyrker opp til 300mA ble testet, driverkretsen ble ikke merkbart varmere.

5.5 Motormodul

Motormodulen har samme lysdiode-drivere som frontmodulen. 4-20mA strømsløyfe er de samme som på brenselcellemodulen. Den eneste nye kretsløsningen som måtte testes var RS-232-transcieveren.

RS-232 kommunikasjon

MAX3232E-kretsen ble først testet ved å kortslutte Rx- og Tx-pinnene. Alt som ble sendt på Tx-pinnen ble også mottatt riktig på Rx-pinnen. Deretter ble COM-porten på en vanlig pc koblet til, både sending og mottak fungerte bra til og fra pc'en. Alle testene ble kjørt på 115.2 kbit/s, det samme som motorstyringen er innstilt på. Da motorstyringen ble koblet til var det ingen motor tilstede, men en startkommando ble sendt og lysdioder på motorstyringen indikerte at meldingen ble riktig mottatt.

Motorstyringen fra Smart Motor har også mulighet for å sende status-meldinger tilbake til mikrokontrolleren. Dette ble testet, men mottatte data stemte ikke med de forventede data som var oppgitt i databladet. Antatt feilkilde var hastigheten på 115.2kbit/s som gav litt for stor feil i overføringshastighet, og dermed førte til bit-feil på mottakersiden. Det ble ikke tid til å omprogrammere motorstyringen til en lavere hastighet for å kjøre nye tester.

5.6 Sikkerhetssystem

Sikkerhetssystemet med strømsløyfe til hovedrelé og hydrogenventil fungerte akkurat som det skulle. Ved å trykke inn nødstop, eller ved hydrogen-deteksjon ble strømsløyfen brutt, ventilen stengte igjen og hovedreleet gikk til åpen posisjon. Ved strømbrydd eller trykk på tak-knappen ble forsyningsspenningen til systemet fjernet og det samme skjedde, ventil og relé gikk til standardposisjonene.

5.7 Sammenkoblet system

For det sammenkoblede systemet var det viktig å teste at CAN-meldinger kom fram og at responstiden var rask nok på vesentlige funksjoner som signalthorn, gass, bremselys og lignende.

CAN-bus

CAN-bussen fungerte fra første stund og uten problemer, det ble ikke oppdaget noen problemer med CAN-bussen. USB-CAN-adapteret som ble benyttet sammen med pc'en sier ifra ved kollisjoner og feiltilstander på bussen, dette ble ikke observert ved noe tilfelle. Ettersom CAN-meldinger angående signalhorn og bremselys ble sendt direkte fra det respektive knappetrykk-interruptet var det ikke mulig å manuelt detektere forsinkelse i overføringen, responstiden må i såfall måles med oscilloskop. Responsten føltes veldig kontant og det var ikke mulig å trykke for fort slik at trykket ikke ble registrert.

Prinsippet med meldingsfiltrering i hardware ble også testet ved å sende veldig mange CAN-meldinger fra PC'en, opptil en melding per millisekund. Dette førte til at de modulene som mottok meldingen i software begynte å miste viktige meldinger fordi meldingsbufferet ble fylt opp av søppelmeldinger. Moduler som filtrerte bort meldingene ble ikke påvirket.

CAN-bus bootloader

Opplasting av ny firmware over CAN-bussen ble benyttet gjennom hele utviklingsperioden for software. Bootloaderen og tilhørende pc-applikasjon er derfor nøye testet gjennom bruk.

6 Diskusjon

Stort sett fungerer instrumenteringssystemet veldig bra, CAN-bus-kommunikasjonen og mikrokontroller-kortene har fungert problemfritt under hele utviklingsprosessen og under konkurransen i Tyskland. De største problemområdene har vært batterimonitorkretsene på brenselcellemodulen og problemer under konkurransen knyttet til motorstyringen fra Smart Motor.

6.1 Brenselcellemodul

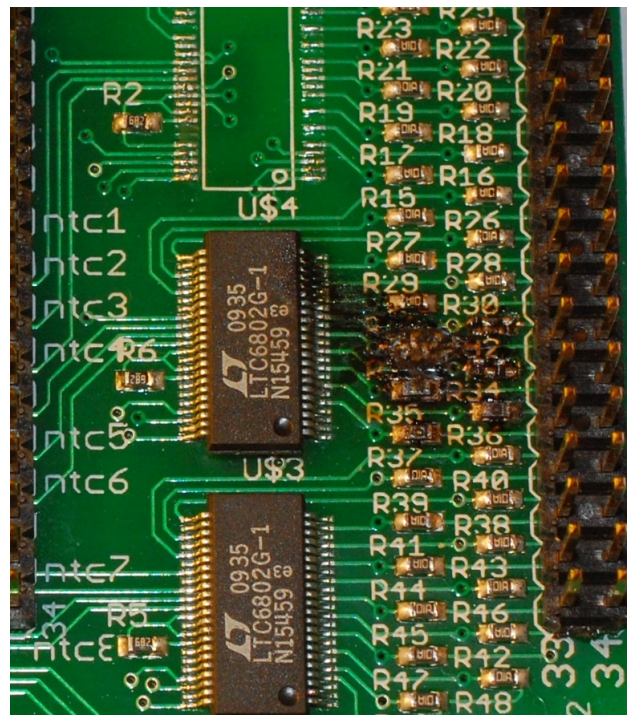
En del av de leverte batterimonitorkretsene viste seg å være ødelagt etter at de ble montert på kretskortet, tre kretser måtte byttes ut fordi de målte helt feil på flere av cellene. Problemet med celle 45 og 46 er sannsynligvis noe lignende. Om kretsene var ødelagt ved levering, eller om det er statisk elektrisitet eller for høy varme under lodding som forårsaket problemene er vanskelig å fastslå. Da kretskortet med fungerende batterimonitorkretser først fungerte og ble montert i boks fungerte det veldig bra.

Ved et tilfelle, under en testkjøring av brenselcellen på lab, brant noen av 100Ω motstandene som står i serie med cellespenningene opp. Motstandene mellom cellene og de respektive inngangene på batterimonitorkretsen var helt forkullet, og kretskortet under var også helt svart, se Figur 56. Dette tyder på kraftig varmeutvikling. En hendelse som kan ha forårsaket en slik varmeutvikling er en kortslutning av to celler gjennom motstandene. Etter å ha sett på loggen over cellespenningene så det ut som en av nabocellene har fått en negativ cellespenning. Dette kan ha ødelagt målekretsen og dermed laget en kortslutning. Negative cellespenninger i brenselcellestappen er noe som egentlig aldri skal forekomme da det tyder på en ødelagt celle. Negative cellespenninger var dermed ikke tatt høyde for i utvikling av kretsen. Å beskytte bedre mot slike hendelser er en forbedring som bør gjøres hvis brenselcellekortet skal benyttes videre.

Monitorkretsene har innebygde beskyttelsesdioder for overspenning og transienter, det kan være disse som har blitt skadet og dermed ført til feil måleverdier for noen av kretsene. Etter søk på nettet ble flere brukererfaringer funnet, se [21]. Problemet med celle 45 og 46, at den ene gikk ned og den andre gikk opp var et av problemene som ble nevnt på nettet og ødelagte dioder var foreslått årsak. Løsningen blir igjen å beskytte med eksterne dioder som kan håndtere mer strøm og dermed beskytte bedre mot transienter og overspenninger.

6.2 Konkurransen

Da vi dro nedover til Tyskland var bilen ennå langt fra ferdig. Deler av styringssystemet ble montert i løpet en halv ukes arbeid i Oslo og all montering i motor-rommet ble foretatt i



Figur 56: Forkullet motstand på brenselcellekortet. En kortslutning av to celler gjennom batterimonitorkretsen er trolig årsaken til de brente motstandene.

Tyskland. Heldigvis var det meste av styringssystemet grundig testet på forhånd. Planen var egentlig å bli ferdig med bilen noen uker før avreise for å få tid til å teste en ferdig bygget bil, se Figur 1 for progresjonsplanen. En hel måneds leveringsforsinkelse på karbonfiber og andre byggematerialer gjorde at de siste ukene før avreise ble brukt til bygging.

Selv om vi dro til Tyskland uten en ferdig bil ble den ferdigbygget der nede og den kom seg gjennom teknisk kontroll. Etter første løp stoppet den etter 400 meter. Feilen som lå i firmware i motorstyringen fra SmartMotor ble funnet og rettet opp. Problemet var at motorstyringen kuttet ut på grunn av en spenningsdipp i DC-spenning fra brenselcellen. Spenningsgrensen ble fjernet, og ny firmware lastet opp. I andre og siste løpsmulighet kom bilen nesten 1 km, men stoppet igjen.

Etter feilsøking ved ankomst i Trondheim ble det funnet at motorstyringsfirmwaren ikke taklet regenerativ bremsing. Ved kjøring i nedoverbakke, eller ved senking av hastighetsreferansen til motoren vil motorstyringen prøve å sende strøm tilbake til brenselcellen. En diode beskytter mot revers-strøm til stacken. Dette fører igjen til at spenningen i motorstyringen stiger over 60V og den kutter dermed ut. Løsningen er å skru av regenerativ bremsing, men dette var en litt mer komplisert oppgave i følge Smart Motor, og ble derfor ikke testet. Fjorårets team unngikk hele problemstillingen fordi de hadde et enveislager i hjulnavet, også kalt frihjulsnav.

Problemene som oppstod i Tyskland kunne vært unngått dersom det hadde vært tid til å teste bilen i Norge før avreise. Problemet med motorstyringen burde helt klart vært tenkt på på forhånd, men den nye motoren til bilen ble ikke ferdig før deler av teamet allerede hadde reist til Tyskland. Omprogrammering av firmware for den nye motoren skjedde bare noen dager før konkurransen. Under tidspress og spesielle omstendigheter er det lett å glemme små, men viktige detaljer.

Instrumeneringssystemet som denne oppgaven har tatt for seg fungerte stortsett bra i Tyskland. Mulighet for å omprogrammere moduler over CAN-bus viste seg å bli veldig nyttig. Da bilen ikke fungerte ble det under feilsøkingprosessen foretatt endringer i koden som så måtte lastes opp til moduler. Flere av modulene ble montert på veldig utilgjengelige plasser i bilen og opplasting over CAN-bus var derfor veldig praktisk. Telemetri-modulen som fungerte fint før avreise fra Norge, fungerte dessverre ikke i Tyskland, sannsynligvis på grunn av problemer med mobiltelefonabonnementet og GPRS-tilkobling. Dette var det ikke tid til å feilsøke under oppholdet i Tyskland.

7 Konklusjon

I løpet av arbeidet med denne masteroppgaven har det blitt utviklet og implementert et komplett instrumenteringssystem for en brenselcelledrevet, miljøvennlig bil som deltar i Shell Eco Marathon. Det er lagt stor vekt på modularitet og systemet er basert på CAN-bus som er en velprøvd standard kommunikasjonsbuss utviklet for bilindustrien. Både software og hardware har blitt testet og har fungert tilfredstillende i konkurranse-sammenheng.

En overvåknings- og styringsmodul for en brenselcellestack med 46 celler er utviklet og tatt i bruk. Modulen er benyttet på lab i flere måneder for å overvåke celledensiteter og temperaturer på brenselcellestacken. Problemer knyttet til elektronikken rundt spenningsmålingene er identifisert, og forbedringer forslått. Modulen kan enkelt utvikles videre for å benyttes som en kompakt batteriovervåkningsmodul i en elektrisk bil. Den er også USB-basert og kan brukes som overvåknings- og styringsmodul for lab-bruk knyttet til en batteri- eller brenselcellestack.

Et komplett interface for sjåføren er også utviklet. Knapper og indikatorlysdiodes, cruisecontrolknapper på rattet, lcd-skjerm og navigeringshjul for et enkelt navigerbart menysystem, er alt styrt fra en elektronikkmodul som er koblet til bilens kommunikasjonsbuss. Menysystemet på skjermen er enkelt utvidbart i software og intuitivt i bruk. En digital vinkelsensor for gasspedal og integrert trykkbryter i bremsepedal er også implementert.

Moduler for å drive lysdioder til kjørellys, indikatorlys og bremselys er laget. Lysstyrken kan reguleres trinnløst for adaptiv lysstyrke og strømsparing etter forholdene. Tilsammen er det mulighet for over 100W levert effekt til lysdioder for hver av modulene. Den høye effekten gjør at drivermodulene kan benyttes til alle former for belysning både innendørs og utendørs, ikke bare til lysdioder i bilen.

Utvikling av software for å ta imot data fra telemetri-modulen ble satt ut til EiT-gruppen som også laget nytt festesystem i transport-hengeren til eco-bilen. Et java-basert system ble laget og testet av EiT-gruppen i Norge, dessverre fungerte ikke GPRS-oppkobling i Tyskland, og det ble derfor ikke tatt i bruk under konkurransen. Systemet for resten av bilen fungerer helt uavhengig av telemetri-modulen, og denne kan derfor kobles inn og ut av bilen uten å påvirke resten av systemet.

Eco-marathon-prosjektet har bydd på en rekke utfordringer knyttet til både design av instrumenteringssystemet, tverrfaglig samarbeid og prosjektplanlegging med en fastsatt tidsfrist. Den aller viktigste erfaringen som er gjort er at ting som oftest tar lenger tid enn planlagt, enten det dreier seg om utvikling, leveringstid til komponenter, feilsøking eller administrative oppgaver. Dette har ført til mye helgejobbing, og døgnskift mot slutten av prosjektet. Det har vært samme tidspresset mot slutten hvert eneste år i forbindelse med Eco-Marathon-prosjektet.

Referanser

- [1] A. L. Guldahl, *Fjernovervåkningsystem for Shell Eco-marathon kjøretøy*, 2009.
- [2] C. Baldwin and K. Clark, *Design rules: The power of modularity* (The MIT Press, 2000).
- [3] J. Catsoulis, *Designing embedded hardware* (O'Reilly Media, Inc., 2005).
- [4] A. Dicks and J. Larminie, *Fuel cell systems explained* (John Wiley & Sons, 2000).
- [5] J. Williams and M. Thoren, (2007), <http://cds.linear.com/docs/Application%20Note/an112f.pdf>.
- [6] Linear Technology, *LTC6802-1 Multicell Battery Stack Monitor*, 2009, <http://cds.linear.com/docs/Datasheet/68021fa.pdf>.
- [7] Murata, *4-20mA Current Loop Primer*, 2009, <http://www.datel.com/data/meters/dms-an20.pdf>.
- [8] M. Day, Analog Applications Journal , 17 (2004).
- [9] A. Gellein, Framdriftssystem til eco marathon kjøretøy, Master's thesis, NTNU, 2009.
- [10] Seoul Semiconductor, *X42182 Z-power LED technical datasheet*, 2009, <http://www.acriche.com/>.
- [11] Microchip, Mcp23s17 io-expander, 2005.
- [12] J. M. H. Bakken, Styresystem for fremdrift av shell-ecomarathon-kjøretøy, Master's thesis, NTNU, 2009.
- [13] A. Devices, Adum1401 digital isolator, 2008.
- [14] Atmel, *AT90CAN Datasheet*, 2008, <http://www.atmel.com/>.
- [15] Automation Components Inc, *ACI A/SCTA series*, 2007, <http://www.workaci.com/content/ascta-50>.
- [16] Traco Power, *DC/DC Converters TEN 12WI Series, 12 Watt*, 2005, <http://www.tracopower.com/fileadmin/medien/dokumente/pdf/datasheets/ten12wi.pdf>.
- [17] IMO PRECISION CONTROLS, *SRF-1C-SL-24VDC - RELAY, SPCO, 24VDC*, 2005, <http://www.farnell.com/datasheets/8019.pdf>.
- [18] Atmel, *Flip3 Bootloader Datasheet*, 2009, <http://www.atmel.com/>.

- [19] Bourns, *Absolute Contacting Encoder (ACE)*, 2008, <http://www.bourns.com/pdfs/ace.pdf>.
- [20] Bluetooth rfcomm / sdp connection to a rs232 adapter in android, 2010, <http://stackoverflow.com/questions/1953888/bluetooth-rfcomm-sdp-connection-to-a-rs232-adapter-in-android>.
- [21] endless-sphere.com, 2009, <http://endless-sphere.com/forums/viewtopic.php?f=14&t=9493#p146557>.

A Innhold på CD

Den vedlagte cd'en inneholder aktuelle artikler, datablader, skjemategninger, kretskortdesign og bilder av komponenter.

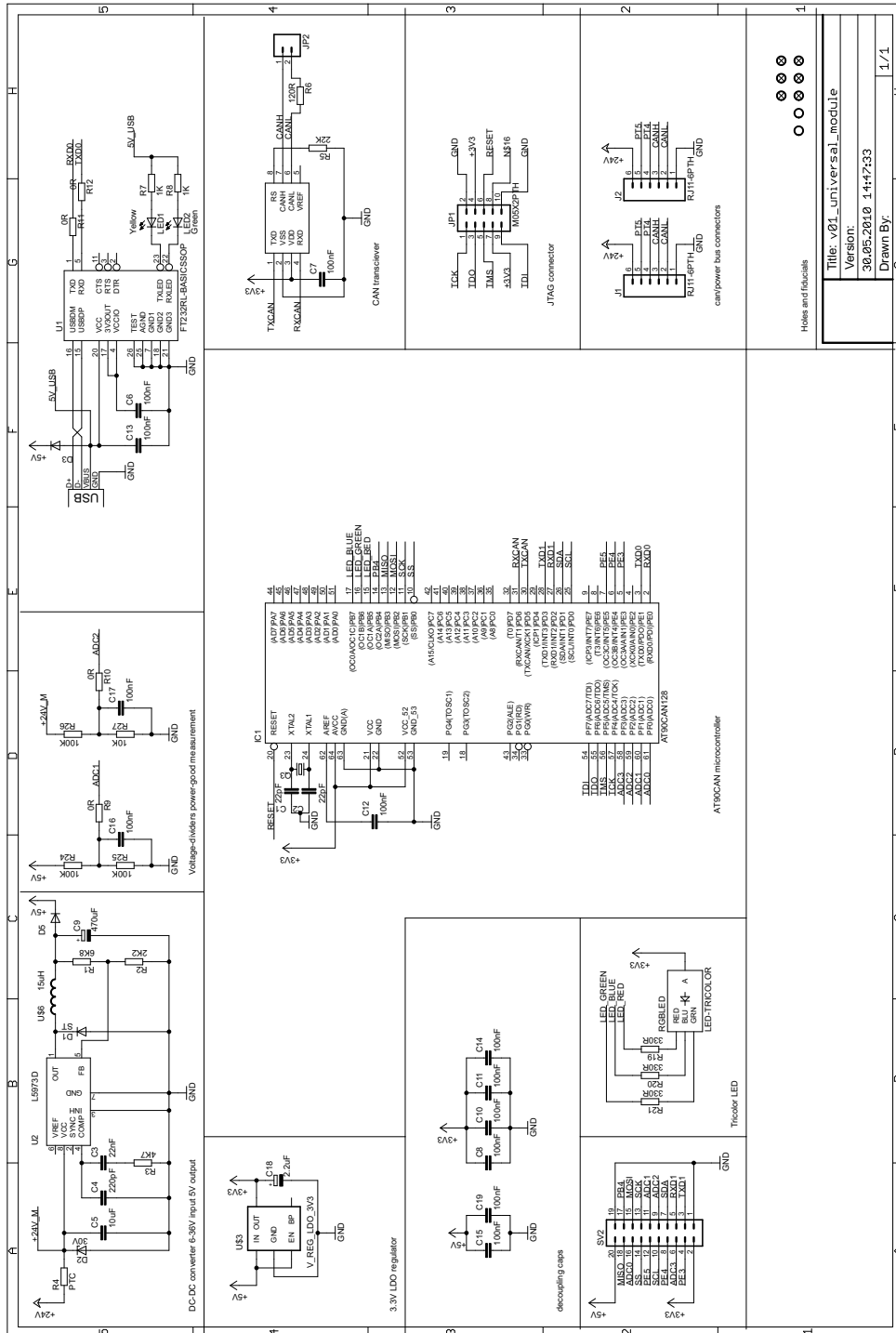
- Artikler
- Bilder
- Datablader
 - atmel_mcu
 - batterymonitor
 - bluetooth
 - dc/dc
 - display
 - encoder
 - highcurrendriver
 - io-expander
 - SPI-isolator
 - lysdioder
 - peccom
 - rs232
 - sensors
- Hardware
 - Eagle-utlegg, gerberfiler og bill of materials
- PDF'er med kretsskjema og kretskortutlegg
- Rapport
- Software
 - Android
 - Atmel
 - Bootloader flip-batchfiler

B CAN-meldinger

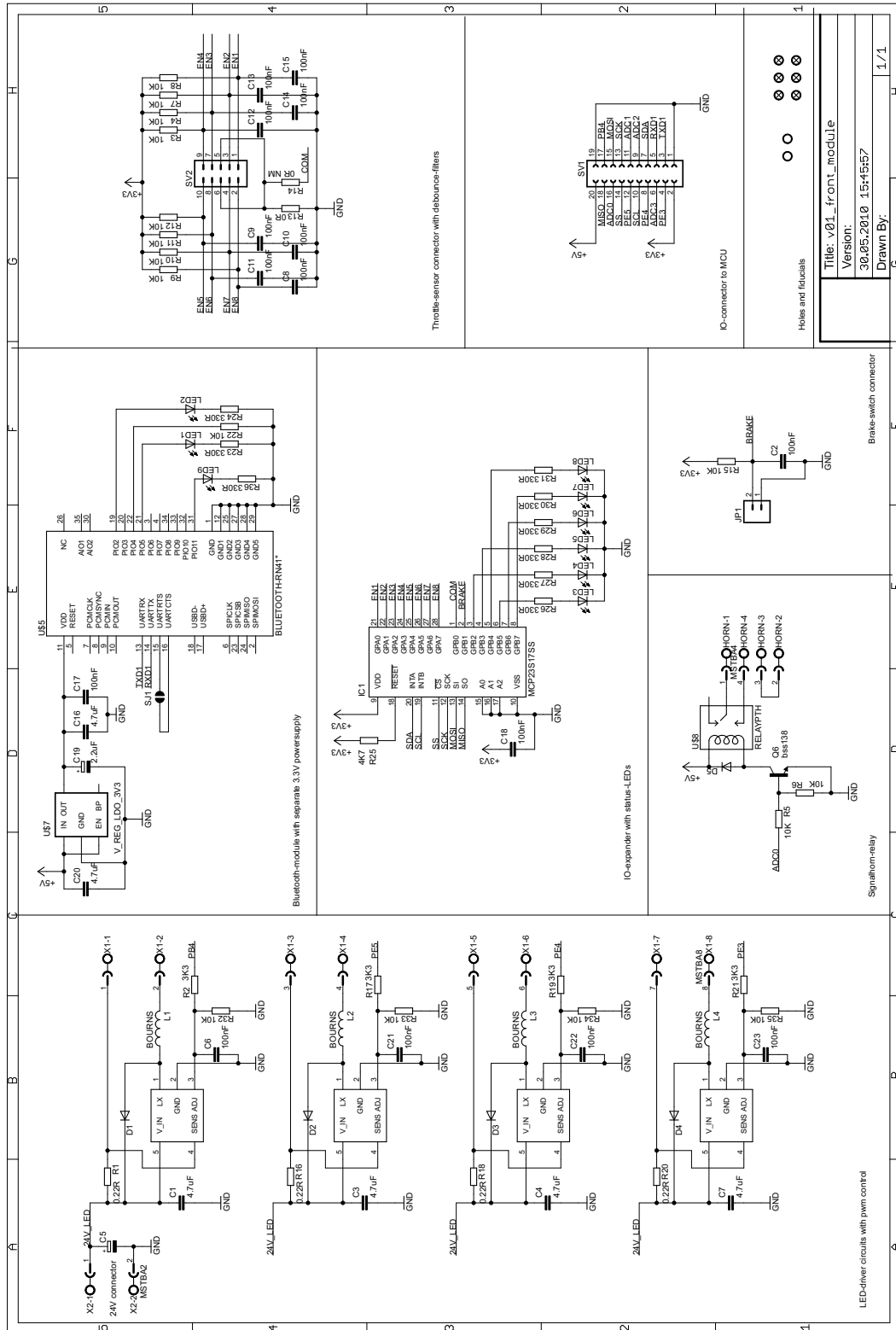
Msg-ID	# byte	Sender	Mottaker	Beskrivelse
0x3C0	1	Dashboard	Fuelcell	1 = purge
0x3C1	1	Dashboard	Fuelcell	1 = skru på brenselcelle med startpurge
0x3C2	1	Dashboard	Fuelcell	Viftelastighet 0-255, 0 = off
0x5C0	1	Dashboard	Frontmodule	Horn 1 = on, 0 = off
0x5C1	4	Dashboard	Frontmodule	Lysstyrke 0-255, nærløys, fjernøys, V blink, H-blink
0x6C0	1	Frontmodule	Motormodule	1 = brems på, 0 = brems av
0x6C1	4	Frontmodule	Motormodule	Lysstyrke 0-255, bakøys, bremsøys, V blink, H-blink
0x6C2	1	Dashboard	Motormodule	1 = motor på, 0 = motor av
0x6C3	1	Dashboard	Motormodule	Hastighetsreferanse 0-255
0x6C4	1	Dashboard	Motormodule	1 = lastrelé på, 0 = lastrelé av
0x6D0	2	Dashboard	Motormodule	16-bit integer, batterispennning
0x700-0x70B	8	Fuelcell	Telemetry & Dashboard	12-bit unsigned integer, 46 brenselcellespenninger
0x70C-0x70E	8	Fuelcell	Telemetry & Dashboard	12-bit unsigned integer, 8 temperaturer
0x70F	8	Fuelcell	Telemetry & Dashboard	16-bit unsigned integer, strømsløyfe målinger
0x710	5	Fuelcell	Telemetry & Dashboard	Høystrøm-utgangstilstander
0x742	1	Frontmodule	Dashboard	Pedalposisjon 0-255
0x745	5	Frontmodule	Dashboard	0 av, 1 på: nær-, fjernøys, nødbløsk, showmode, horn
0x750	8	Fuelcell	Dashboard	Status-msg til dashbord, 0 = OK
0x754	8	Frontmodule	Dashboard	Status-msg til dashbord, 0 = OK
0x758	8	Motormodule	Dashboard	Status-msg til dashbord, 0 = OK
0x762	8	Telemetry	Dashboard	Status-msg til dashbord, 0 = OK
0x763	8	Telemetry	Dashboard	Tid, dato[ff,ff,timer,minutter,sekunder,dag,måned,år]
0x764	8	Telemetry	Dashboard	Latitude, Longitude
0x782	1	PC	Telemetry	1 = Formater SD-kort
0x7F0	1	PC	Fuelcell	Initier bootloader etter bootloader-reset. NNB = 0x01
0x7F0	1	PC	Frontmodule	Initier bootloader etter bootloader-reset. NNB = 0x02
0x7F0	1	PC	Motormodule	Initier bootloader etter bootloader-reset. NNB = 0x03
0x7F0	1	PC	Dashboard	Initier bootloader etter bootloader-reset. NNB = 0x04
0x7F0	1	PC	Telemetry	Initier bootloader etter bootloader-reset. NNB = 0x05

C Skjemategninger

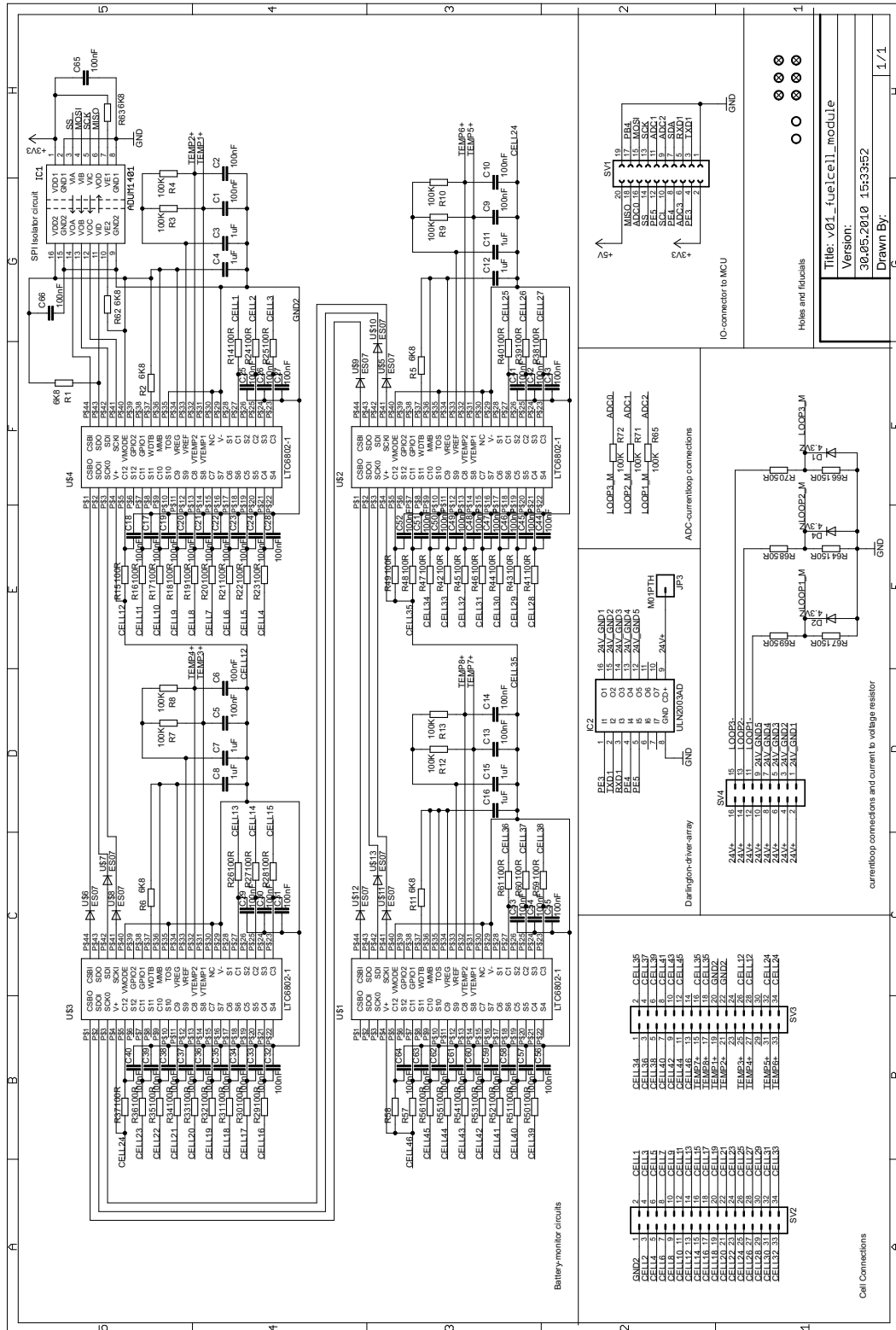
C.1 Universal-kretskort



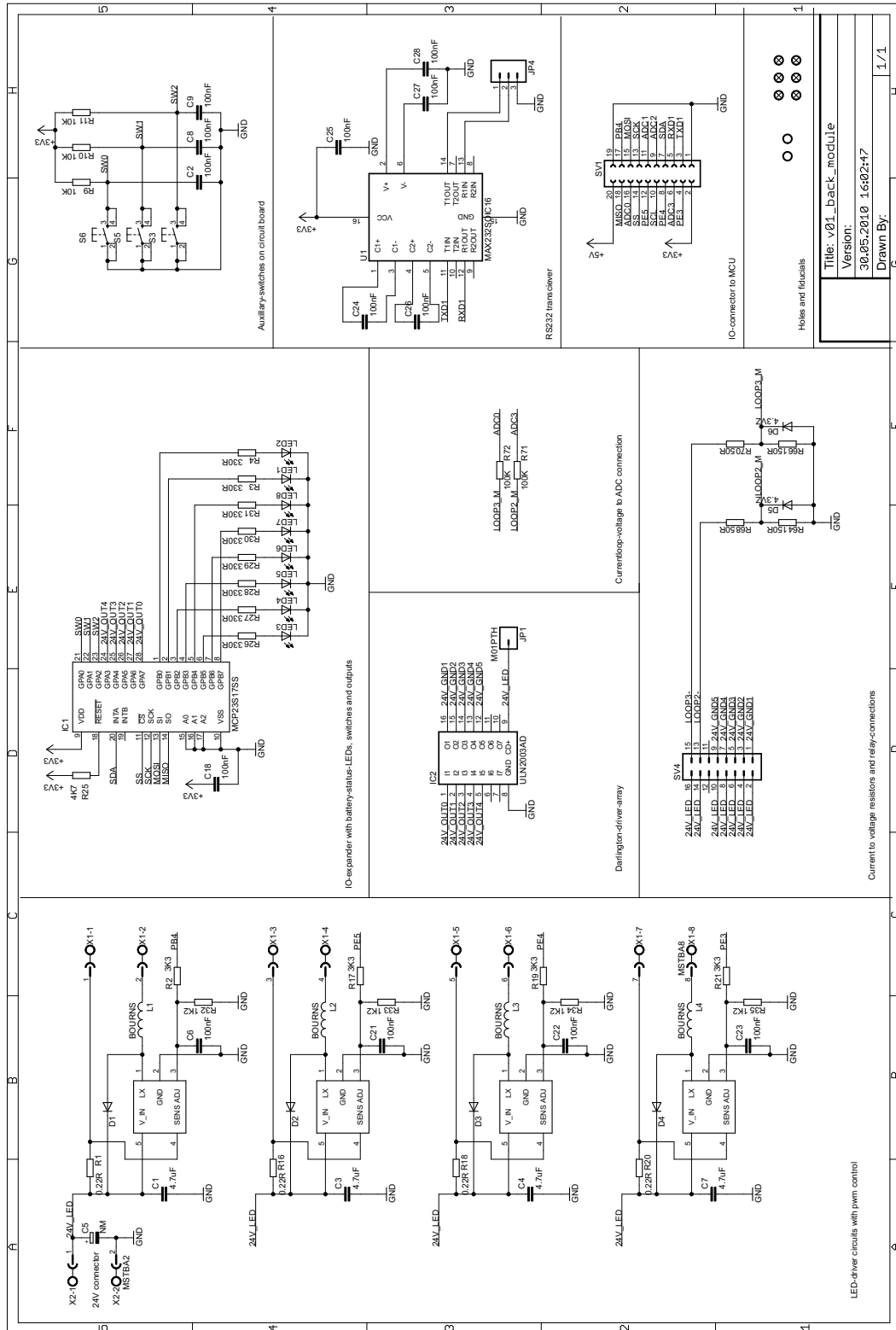
C.2 Frontmodul



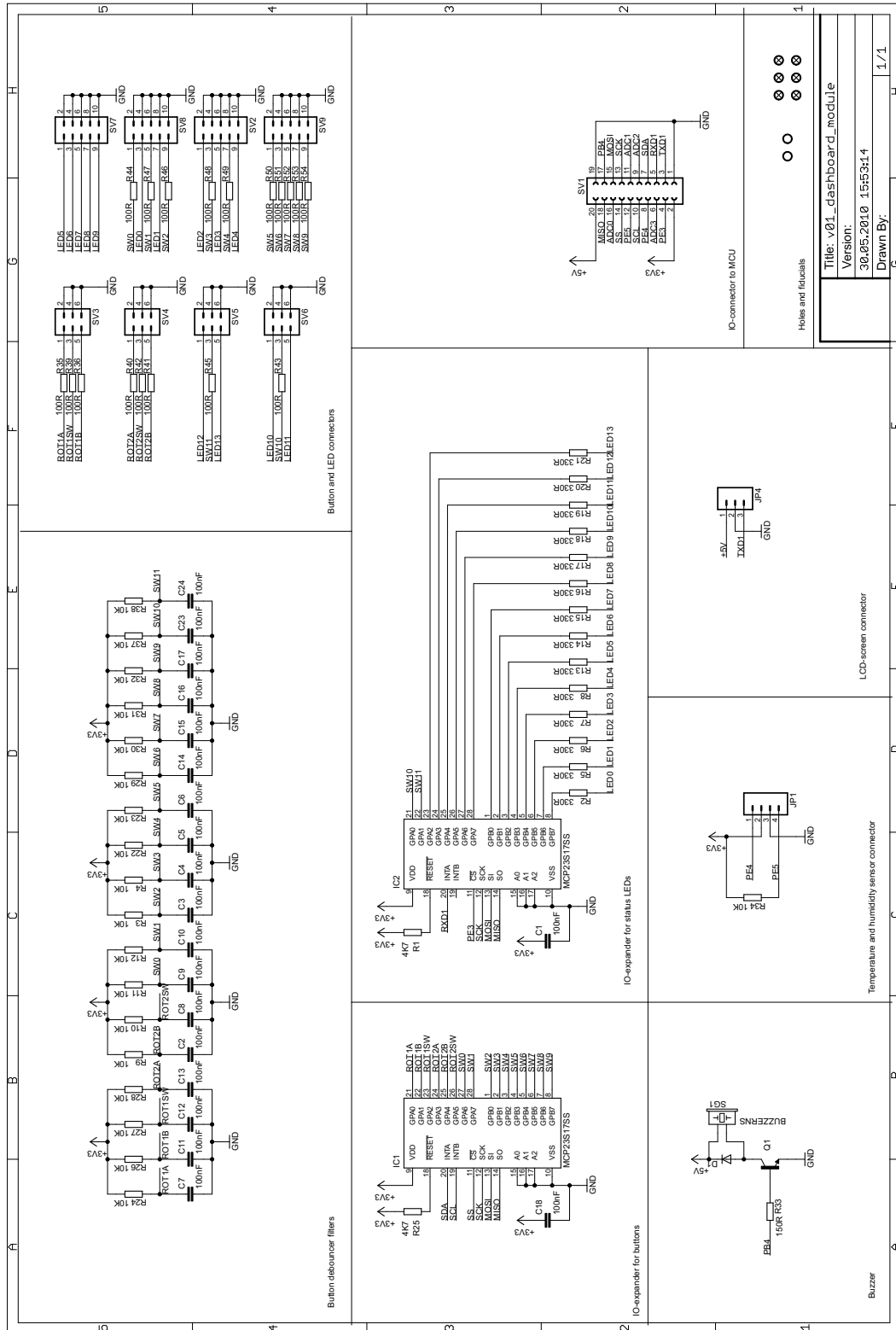
C.3 Brenselcellemodul



C.4 Motormodul



C.5 Dashbordmodul



D Bootloader

Framgangsmåten for å innstallere CAN-programvare og laste opp ny firmware til en modul er gitt under:

- Innstaller driver for PEAK CAN-USB-adapter.
- Innstaller siste versjon av Flip3, finnes på Atmel sin hjemmeside.
- Sjekk at referansen til hex-fil er riktig i den aktuelle batchisp_commandfila.
- Med systemet koblet til CAN-adapteren, kjør batch-filen med samme navn som modulen som skal programmeres.

```

c:\ ATMEL FLIP Command Line Interpreter
Device selection..... PASS
Hardware selection..... PASS
Opening port..... PASS
Timeout set to 1sec..... PASS
Opening CAN Node..... PASS* Cannot write to port.
Closing CAN Node..... PASS 0x02
Waiting 2sec..... PASS
Opening CAN Node..... FAIL The CAN node is closed.
Opening CAN Node..... PASS 0x02 0x7f
Reading Bootloader version..... PASS 1.0.4
Erasing..... PASS
Blank checking..... PASS 0x00000 0x1dfff
Parsing HEX file..... PASS ..\v01_front_module\default\Fron
t_module.hex
Programming memory..... PASS 0x00000 0x02b8b
Verifying memory..... PASS 0x00000 0x02b8b
Starting Application..... PASS NORESET 0
Closing CAN Node..... PASS 0x02

Summary: Total 17 Passed 16 Failed 1
C:\My Dropbox\Ecomarathon\software\bootloader>pause
Press any key to continue . . . _

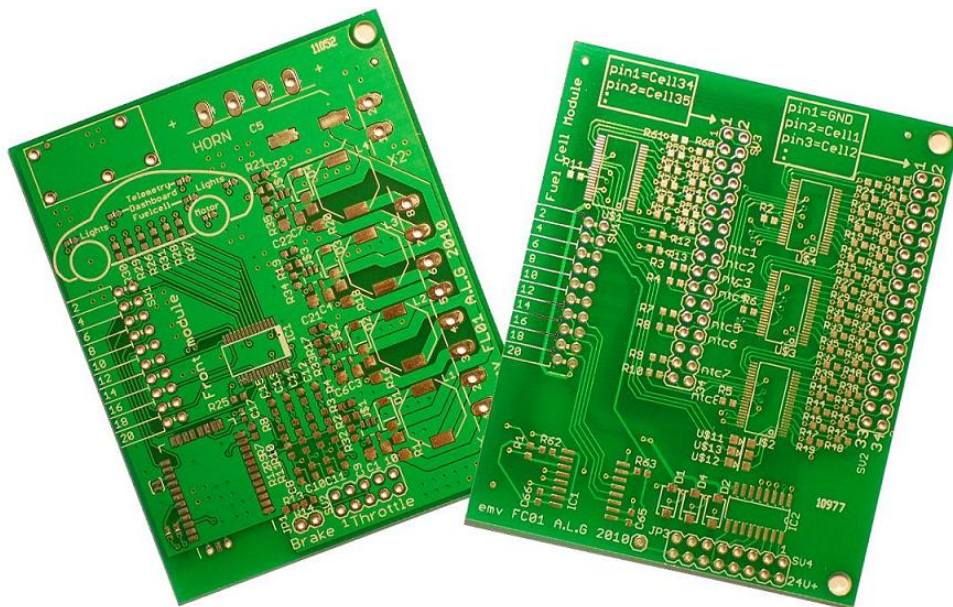
```

Figur 57: Skjermdump av en vellykket kjøring av bootloader-batchprogrammet med en retry på “opening CAN node”.

E Bestilling av kretskort

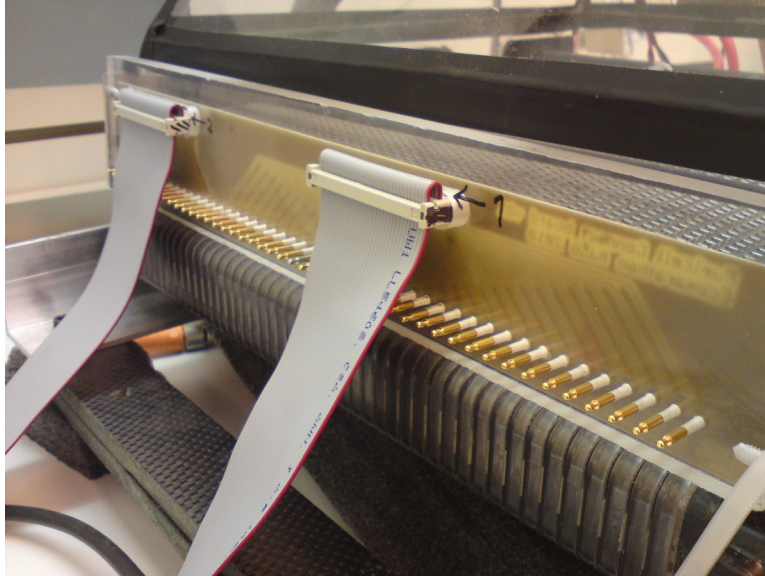
Her følger en sjekklister og oppskrift for bestilling av profesjonelt produserte kretskort, Cadsoft eagle er brukt for skjemattegning og utlegg:

- Sjekk at alle linjer er koblet riktig, Rx/Tx, klokke/data etc.
- Sjekk at jordplan er kontinuerlig, legg til flere via'er om nødvendig.
- Sjekk at alle komponentene er tilgjengelig med riktig footprint.
- Sjekk at all tekst er vektorgrafikk og flytt på komponentnavn så det ser bra ut.
- Sjekk at plassering av store komponenter ikke kolliderer med f.eks. JTAG-adapter.
- Kjør design rule check med riktig parametere avhengig av hvor det bestilles kort fra.
- Print utlegg i skala 1:1 og sjekk enda en gang om komponenter kolliderer og om footprint stemmer.
- Følg denne guiden (<http://hackaday.com/2009/01/15/how-to-prepare-your-eagle-designs-for-manufacture/>) og lag gerberfiler (produksjonsfiler for hvert lag i kortet).
- Sjekk gerberfilene i viewplot-programmet.
- Zip filene og send til produksjon. (batchpcb.com ble brukt for denne oppgaven)

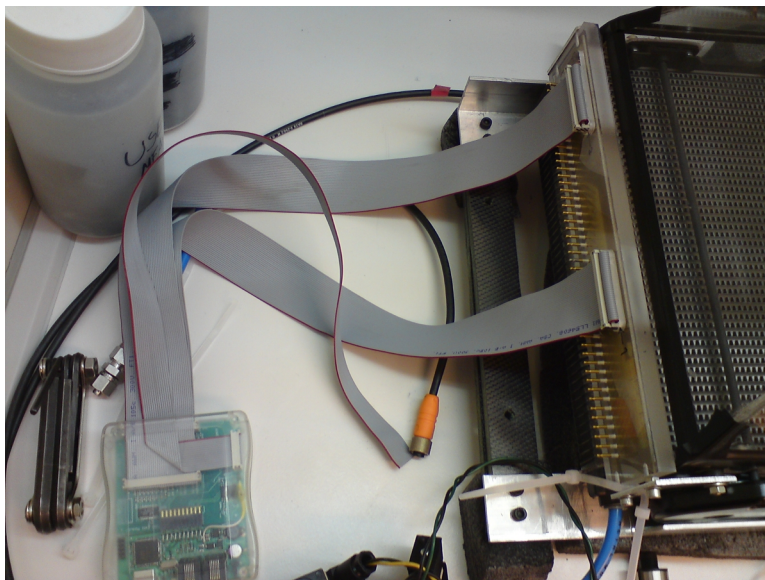


Figur 58: Kretskort rett fra produksjon.

F Brenselcelletilkobling



Figur 59: For å få elektrisk kontakt mellom hver celle i stacken og flatkablene som går til styringskretskortet er det benyttet en rad med fjærbelastede nåler. Nålene er laget for produksjonstest av kretskort og er derfor laget for å få god elektrisk kontakt med mange materialer.



Figur 60: Bildet viser hvordan tilkoblingen mellom brenselcellestacken og kretskortet er utført. Fra raden med fjærbelastede nåler går det to flatkabler med 40 ledere i hver til styringsmodulen for brenselcellestacken.