



Universitetet  
i Stavanger

**DET TEKNISK-NATURVITENSKAPELIGE FAKULTET**

## **MASTEROPPGAVE**

Studieprogram/spesialisering:

Robotteknologi og signalbehandling

Vårsemesteret, 2020

Åpen

Forfatter: Raymond Tjørhom

Fagansvarlig: Kristian Thorsen

Veileder: Kristian Thorsen

Tittel på masteroppgaven:

Estimering av ladestatus i en elektrisk racerbil med bruk av Kalmanfilter

Engelsk tittel:

Estimating State of Charge in an Electric Race Car Using Kalman Filter

Studiepoeng: 30

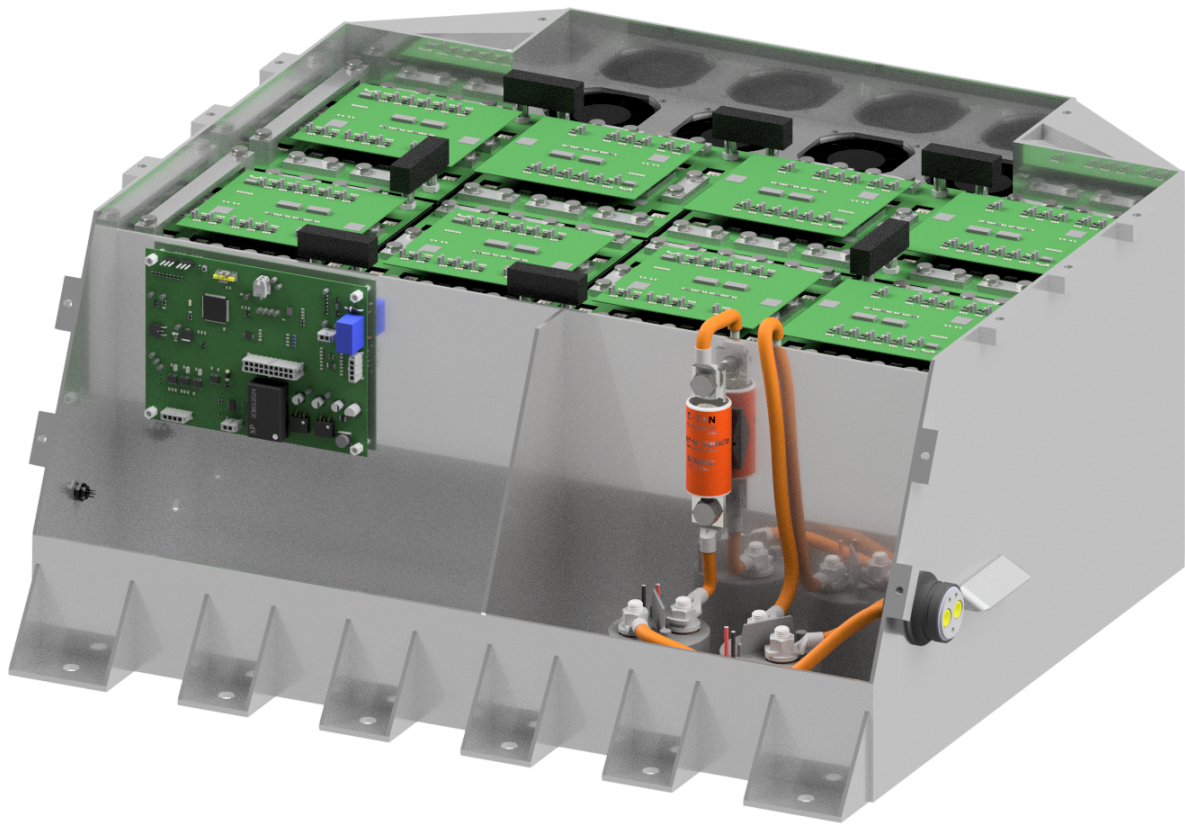
Emneord:

Batteri, Kalmanfilter, ION Racing,  
FSAE, Formula Student, SPKF, UKF

Sidetall: 44

+ vedlegg/annet: 60

Stavanger, 12.06.2020  
dato/år



# Estimering av ladestatus i en elektrisk racerbil med bruk av Kalmanfilter

Raymond Tjørhom

Vårsemester 2020

# Sammendrag

Denne oppgaven er skrevet i samarbeid med ION Racing, en studentorganisasjon ved Universitetet i Stavanger som hvert år bygger en elektrisk racerbil. Oppgaven dreier seg om ferdigstilling av et batteriovervåkningssystem samt utviklingen av en effektiv algoritme for estimering av ladestatus i racerbilens akkumulatørsystem.

Batteriovervåkningssystemet var et påbegynt prosjekt fra høsten 2019, og hadde en ferdig master-modul. Til denne oppgaven måtte derfor slave-modulene designes og konstrueres. Slave-modulene er et samlenavn for to kretskort stablet i høyden, hvor det nederste er direkte tilkoblet celleterminaler, mens det øverste foretar kommunikasjon med master-modulen. Det øverste kortet er enkelt utskiftbart med konnektorer og adresseringsbrytere. Slave-modulene er godt gjennomtenkte og bærer preg av et enkelt men elegant design.

Til å estimere ladestatus er det benyttet et unscented Kalmanfilter med Sigma-punkter, som gir bedre resultater enn et utvidet Kalmanfilter i særdeles ulineære prosesser. Estimeringen har gitt gode resultater under simulerte prosesser.

Hverken batteriovervåkningssystemet eller estimeringen av ladestatus ble ferdigstilt. Før koronautbruddet hadde begge hovedprosessene god fremgang men grunnet de ekstraordinære omstendighetene rundt covid-19 ble flere prosesser i oppgaven satt på vent på ubestemt tid, og enkelte deloppgaver ble det også umulig å fullføre innen levering av oppgaven. Det arbeidet som er fullført antyder et vellykket produkt.

# Forord

Denne rapporten markerer slutten på min tid som student ved UiS, og ved rapportens innlevering er det også slutten på min tid som et aktivt medlem av ION Racing. Etter å ha vært et medlem i to sesonger, midtveis i den tredje, var det naturlig å velge en masteroppgave i samarbeid med ION Racing. Siden akkumulatortsystemet var et system som både hadde relevante problemstillinger for min studieretning og komponenter som jeg trives å jobbe med var det et nærliggende fokusområde. Etter å ha gjennomført et prosjekt om batteriovervåkningssystemet høsten 2019, falt valg av masteroppgave på å utvikle en algoritme for estimering av ladestatus. Dessverre medførte utbruddet av covid-19 en full stans i produksjon av deler, og jeg fikk ikke anledning til å fullføre alle delprosessene jeg hadde begynt på.

## Takk

Jeg vil gjerne takke de følgende:

- Min veileder, Kristian Thorsen, for god hjelp og tilbakemelding.
- Tom Ryen & Patricia Retamal Bravo for at de kjempet for at jeg skulle få lov til å starte på masterprogrammet til tross for at jeg ikke oppfylte karakterkravet. Jeg håper jeg har vist meg verdig tilliten.
- Min familie for all deres støtte. Jeg har kanskje vært stresset i telefonsamtaler og fraværende på sammenkomster, men det har vært deilig å kunne ta turen hjem fra tid til annen og koble av.
- Alle sponsorer av ION Racing. Uten deres støtte hadde det simpelthen ikke vært mulig å ha et studentprosjekt av en sånn størrelse.
- Sist, men ikke minst, tidligere og nåværende medlemmer av ION Racing for alle minnene. Dette har vært mitt tredje strake år i ION Racing, og det har vært tre år med læring, glede og vennskap, men det har også vært frustrasjon, utmattelse og skuffelser. Det har vært litt av en opplevelse, og jeg ser tilbake på den med glede.

Raymond Tjørhom  
*Stavanger, Juni 2020*

# Innhold

<b>1</b>	<b>Introduksjon</b>	<b>1</b>
1.1	ION Racing .....	1
1.2	Motivasjon .....	4
1.3	Tidligere arbeid .....	5
1.4	Teori: Estimering av ladestatus .....	9
1.5	Utstyr.....	16
<b>2</b>	<b>Konstruksjon</b>	<b>17</b>
2.1	Oppbygning av slave-systemet .....	17
2.2	Kalmanfilter .....	23
<b>3</b>	<b>Resultater</b>	<b>27</b>
3.1	Kretskort i slave-modulen .....	27
3.2	Simuleringer i MATLAB .....	30
<b>4</b>	<b>Diskusjon og Konklusjon</b>	<b>41</b>
	<b>Bibliografi</b>	<b>43</b>
<b>A</b>	<b>Oppbygning av akkumulatorsystemet</b>	<b>I</b>
<b>B</b>	<b>Skjematikk: AMS-slave</b>	<b>IX</b>
<b>C</b>	<b>MATLAB-script</b>	<b>XX</b>
<b>D</b>	<b>Simuleringer i MATLAB</b>	<b>XXX</b>

# Figurer

1	ION Racings bil for 2019 .....	2
2	Blokkskjema av batteriovervåkningssystemet .....	6
3	Oversiktsskjema over komponenter i akkumulatortsystemet.....	7
4	Sammenstilling av akkumulatortsystemet i Autodesk Inventor .....	8
5	Cellemodell .....	10
6	Prinsippskisse av unscented-transformen.....	13
7	Alternativer for positiv låsing .....	19
8	Cellesegment-sammenstillinger fra Autodesk Inventor .....	20
9	Strømvei i akkumulatoren .....	20
10	Modell av slavekort fra Altium Designer .....	22
11	Flytskjema over spenningsestimering .....	24
12	Utladekurver for cellemoduler fra datablad .....	24
13	Utladekurver for celledspenning under forskjellig last .....	25
14	Skråperspektiv av slave-kort .....	27
15	Topplaget til slave-kort .....	28
16	Bunnlaget til slave-kort .....	28
17	Fugleperspektiv av layout-kort: indre versjon .....	29
18	Fugleperspektiv av layout-kort: ytre versjon .....	29
19	Fugleperspektiv av sammenstilling av slave-kort .....	30
20	Kurver fra simulering med konstant 1C last .....	33
21	Kurver fra simulering med rolig lastmønster .....	34
22	Kurver fra simulering med middels hardt lastmønster .....	36
23	Kurver fra simulering med hardt lastmønster .....	37
24	Kurver fra simulering med hardt lastmønster på 96 cellemoduler .....	39
25	Estimering av ladestatus med 96 celler vs. 1 celle .....	40

# Tabeller

1	Nøkkeldata for battericellene .....	9
2	Nøkkeldata for cellesegmentene .....	9
3	Nøkkeldata for akkumulatorssystemet .....	9
4	Alternativer for positiv låsing av celledmoduler .....	18

# Algoritmer

1	Unscented Kalmanfilter med Sigma-punkter .....	15
2	Unscented Kalmanfilter med Sigma-punkter for cellemodell .....	26



# Nomenklatur

- AIR Accumulator Isolation Relay  
*Mekanisk relé som elektrisk isolerer akkumulatoren fra resten av HV-systemet ved feil*
- AMS Accumulator Management System  
*System som overvåker akkumulator-parametre*
- CAD Computer Aided Design  
*Dataassistert konstruksjon/3D-modellert design*
- EKF Extended Kalman Filter  
*Kalmanfilter som estimerer prosesser via linearisering*
- EV Electric Vehicle  
*Elektrisk bil*
- HV High Voltage  
*Høyspenning, definert av regelverk i Formula SAE som spenning over 60 VDC*
- IC Integrated Circuit  
*Kollektiv terminologi for elektroniske komponenter*
- IMD Insulation Monitoring Device  
*Isolasjonsovervåker plassert mellom HV-systemet og jordpotensiale*
- LUT Look-Up Table  
*Oppslagstabell*
- PCB Printed Circuit Board  
*Kretskort*
- SMD Surface Mount Device  
*Overflatemontert komponent på kretskort.*
- SNR Signal-to-noise ratio  
*Størrelsesforhold mellom signal og støy*

- SOC State of Charge  
*Ladestatus i et akkumulatorsystem*
- SPI Serial Peripheral Interface  
*Kommunikasjonsprotokoll mellom master og slave*
- SPKF Sigma Point Kalman Filter  
*UKF med sigma-punkter*
- TS Tractive System  
*Den elektriske delen av drivlinjen. Er en del av HV-systemet*
- UKF Unscented Kalman Filter  
*Kalmanfilter som estimerer prosesser uten linearisering*
- VDC Volt, Direct Current  
*Forkortelse som angir spenningen som DC*

# Kapittel 1

## Introduksjon

Denne oppgaven fokuserer på ferdigstilling av et batteriovervåkningssystem og utvikling av en effektiv algoritme for estimering av ladestatus i en elektrisk racerbil. I løpet av det siste tiåret har andelen elbiler i Norge økt kraftig, og i inngangen til 2020 utgjorde elektriske biler 10 % av den norske bilstanden [10][11][12]. Med denne nye bilparken kommer det også nye begreper, og i 2013 kom *rekkeviddeangst* på andreplass i Språkrådets kåring av årets ord [17]. Rekkeviddeangst er frykten for å ikke rekke frem til destinasjonen din med den energien du har tilgjengelig. Begrepet kan brukes om alle typer biler, men det brukes primært om helelektriske biler. *State of Charge*, SOC, er et begrep brukt om gjenværende energi i et akkumulatorsystem av battericeller, og oppgis vanligvis til føreren som % av fulladet.

### 1.1 ION Racing

Oppgaven er i samarbeid med ION Racing, en studentorganisasjon ved Universitetet i Stavanger som konkurrerer i den største studentkonkurransen i verden, Formula SAE. Formula SAE, eller Formula Student som konkurransen kalles på folkemunne, er en ingeniørfaglig designkonkurransen som utfordrer lag av universitetsstudenter til å beregne, tegne og konstruere en liten formel-type bil (se Figur 1 for referanse) som de så konkurrerer med mot andre lag.



**Figur 1:** ION Racings bil for 2019. Bilde tatt under Formula Student Netherlands 2019

## Regelverk i Formula Student

Det er 15 forskjellige konkurranser på verdensbasis, og de har forskjellige regelverk som studentene må følge. Alle de europeiske konkurransene tar utgangspunkt i regelverket utarbeidet av Formula Student Germany [4], men de kan også velge å ha et sett med tilleggsregler. De viktigste reglene for denne rapporten er et utvalg fra *Technical*, *T* og *Electric Vehicle*, *EV*:

[4, T 10.2.2]

The following methods are accepted as positive locking mechanisms:

- Correctly installed safety wiring.
- Cotter pins.
- Nylon lock nuts (ISO 7040, ISO 10512, EN 1663 or equivalent) for low temperature locations (80 °C or less).
- Prevailing torque lock nuts (DIN 980, ISO 7042 or equivalent, and jet nuts or K-nuts).
- Locking plates.
- Tab washers.

Any locking mechanism based on pre-tensioning or an adhesive is not considered a positive locking mechanism.

[4, EV 5.3.2]

Each TS accumulator segment must not exceed a maximum static voltage of 120 VDC, a maximum energy of 6 MJ, see EV 5.1.2, and a maximum mass of 12 kg.

[4, EV 4.5.13]

All electrical connections, including bolts, nuts and other fasteners, in the high current path of the TS must be secured from unintentional loosening by the use of positive locking mechanisms that are suitable for high temperatures, see T 10.2.

[4, EV 4.5.15]

Soldered connections in the high current path are only allowed if all of the following are true:

- connections on PCBs
- the connected devices are not cells or wires
- the devices are additionally mechanically secured against loosening

[4, EV 5.8.3]

Cell temperature must be measured at the negative terminal of the respective cell and the sensor used must be in direct contact with the negative terminal or less than 10 mm along the high current path away from the terminal in direct contact with the respective busbar. It is acceptable to monitor multiple cells with one sensor if this requirement is met for all cells sensed by the sensor.

## Konkurransenes innhold

De forskjellige konkurransene i Formula Student består av de samme grenene:

### Statiske grener:

- Business

Lagene skal presentere bilen som en prototype av et forretningskonsept for en banebil, hvor dommere er "potensielle investorer".

- Design

Utvalgte medlemmer blir grillet av dommere om de forskjellige design-valgene de har gjort og hvordan dette støtter opp om lagets hovedkonsept.

- Kost

Lagene skal forsvare ressursbruken og vurdere hvordan de skal foreta masseproduksjon av deler i en innbilt forretningssituasjon.

**Dynamiske grener:**

- Akselerasjon

75 meter drag race. (Stresser drivverk)

- ”8-tall”, *Skid pad*

2 runder i høyre halvdel etterfulgt av to runder i venstre halvdel. (Tester hjuloppheng)

- Sprint

Én runde på kortbane. (Eksplisivt stress på hele bilen)

- Langløp

Mange runder på kortbanen fra Sprint-grenen i til sammen 22 km. (Tester hele bilen med vedvarende stress)

- Effektivitet

Antall kWh brukt på Langløp. (Vurderer hvor godt energien blir utnyttet)

Disse forskjellige grenene skal teste hvor godt lagene har vurdert de forskjellige valgene de har tatt underveis i prosessen. De statiske grenene avgjør hvordan laget har fungert som en helhetlig organisasjon, mens de dynamiske grenene tester hvor gode de forskjellige bildesignene er under krevende omstendigheter. Langløpet er den mest krevende grenen for akkumulatorsystemet, og det er dette som ligger til grunn for valg av kapasitet og kjøling. Under langløpet vil snittfarten være ca. 60 km/h, avhengig av drivtype og aerodynamiske innretninger.

## 1.2 Motivasjon

Høsten 2019 ble det utført et prosjekt som omhandlet design av master-modulen i et batteriovervåkningssystem (*Accumulator Management System*, AMS), samt opprettelsen av kommunikasjon med systemet [7]. Selv om dette prosjektet hadde god fremgang i implementeringen av AMS-masteren, var det fremdeles arbeid som gjenstod for å få et fullverdig overvåkningssystem. Blant annet måtte slave-modulene til AMS lages, og hele systemet måtte testes før det kunne vurderes som sikkert nok til å brukes i akkumulatoren. Det var ønskelig å fortsette på AMS, men det var også viktig å finne en masteroppgave som var mer relevant for studieretningen. Dette skulle ideelt sett være

en funksjon som ikke var kritisk for systemet. Siden det inngår en stor andel signalbehandling i testing av et akkumulatorsystem, ble estimering av ladestatus vurdert til å være ønskelig, men ikke være kritisk for systemet dersom det ikke skulle bli tid til å implementere.

### 1.3 Tidligere arbeid

ION Racings akkumulatorsystem er en del av bilens *High Voltage*(HV)-system. På bakgrunn av resultatlisten fra Formula Student Germany 2019<sup>1</sup> ble det vurdert til at ION Racing burde ha en kapasitet på minimum 21 Ah. Valg av celler falt på cellemoduler bestående av 10 Samsung INR18650-25R celler i parallell. Hver module har en kapasitet på 25 Ah og har en utladeevne på 15C.<sup>2</sup> 96 slike moduler er koblet i serie, noe som gir akkumulatorsystemet en 10P96S konfigurasjon. Ved å koble 96 cellemoduler i serie oppnår man en maksimal spenning på 403.2 VDC. Denne spenningen er valgt for å ikke overstige den maksimalt tillatte spenningen i motorkontrolleren, som er 440 VDC [18, side 15].

#### C-rate

Battericeller har en viss karakteristikk kalt utladeevne, eller C-rate. C-rate er et tall på hvor fort cellen kan utlade dens fulle kapasitet på en sikker måte, eller for å si det mer folkelig: hvor mange ampere man kan trekke fra batteriet. 1C vil si at cellen kan fullstendig utlades i løpet av én time [9].

$$C_{max} = \frac{I_{max}}{\text{Cellekapasitet}} \quad [\text{h}^{-1}] \quad (1.1)$$

Cellemodulene i akkumulatoren 25 Ah kapasitet, og har en C-rate på 15C, hvilket gir et maksimalt teoretisk strømtrekk på:

$$I_{max} = C_{max} \cdot \text{Cellekapasitet} = 15 \text{ h}^{-1} \cdot 25 \text{ Ah} = 375 \text{ A} \quad (1.2)$$

For å beskytte motoren er det installert en 125 A sikring i akkumulatorsystemet, så cellene vil ikke kunne levere mer enn tilsvarende

$$C = \frac{I_{max}}{\text{Cellekapasitet}} = \frac{125 \text{ A}}{25 \text{ Ah}} = 5 \text{ h}^{-1} \quad (1.3)$$

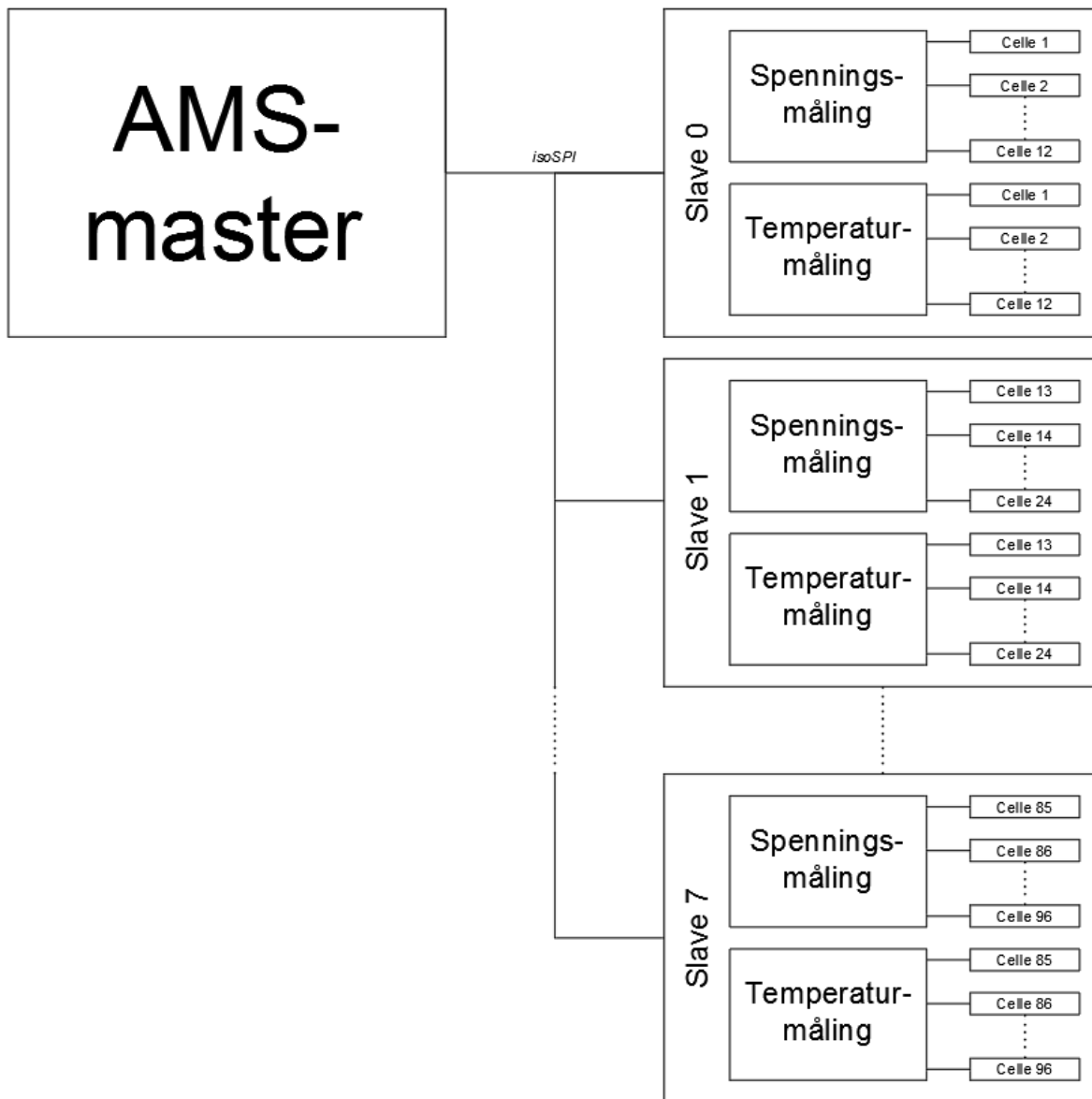
#### Overvåkning av akkumulatorsystemet

AMS-en har et master-/slave-system, som beskrevet i [7]. Masteren kommuniserer med slavene via isolert SPI, *Serial Peripheral Interface*, og får informasjon om spenning

<sup>1</sup><https://www.formulastudent.de/fsg/results/2019/> (Krever bruker)

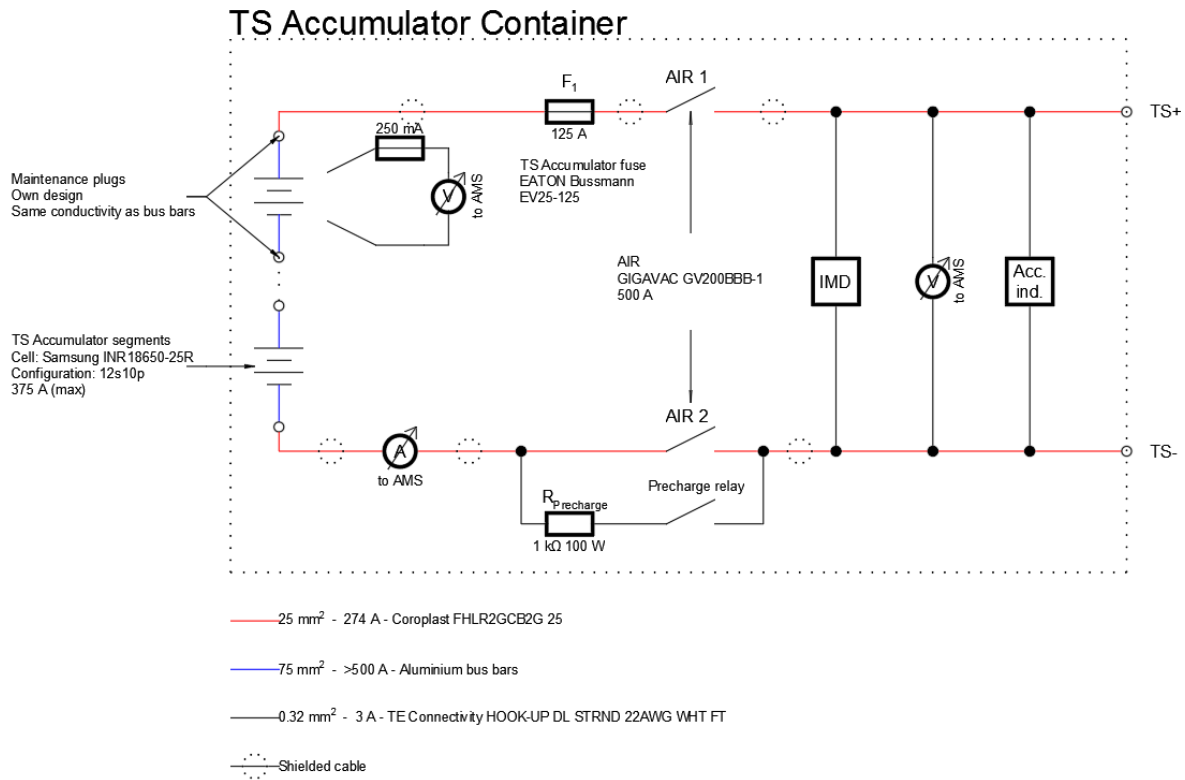
<sup>2</sup>Levert fra Energus Power Solutions. Produktnummer Li1x10p25RT

og temperatur. Strømmåling foretas av en strømsensor på utgangen av HV-systemet. Cellene er delt inn i 8 segmenter á 12 celledmoduler hver. Dette har en todelt grunn; for å overholde regel EV 5.3.2, samt at det er mer praktisk å feilsøke på et enkelt segment enn på hele systemet. For mer om utformingen av akkumulatorsystemet se Tillegg A.

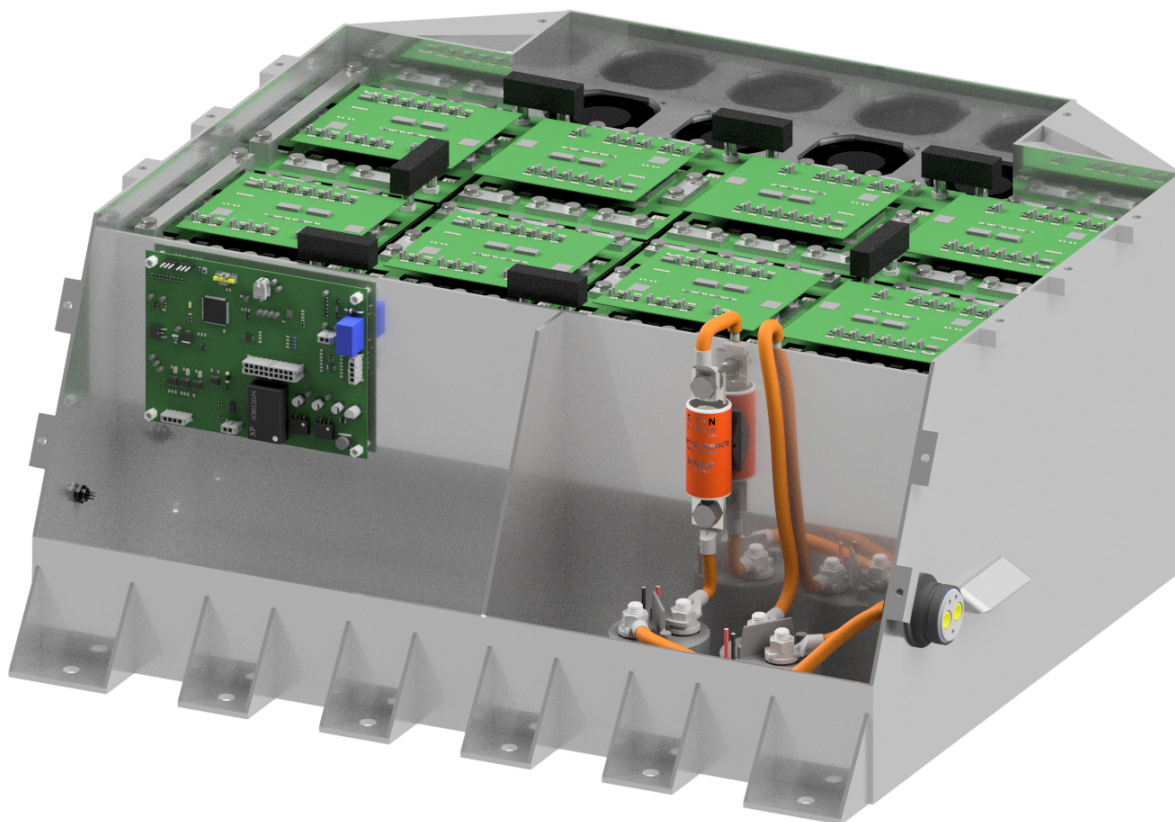


**Figur 2:** Blokkskjema av batteriovervåkingssystemet. Hver slave foretar spenning- og temperaturmåling. Master-modulen kommuniserer med slavene over isoSPI. Mastermodulen foretar beregninger og eventuell frakobling av HV-systemet dersom akkumulatorparametre er utenfor akseptable nivåer.





**Figur 3:** Oversiktsskjema over komponenter i akkumulatortsystemet. Figuren er hentet fra dokumentasjon påkrevd av konkurransearrangører. De forskjellige forkortelsene på komponenter i figuren er; TS - *tractive system*, AMS - *accumulator management system*, AIR - *accumulator isolation relay*, IMD - *insulation monitoring device*, Acc. ind. - *Accumulator indicator*.



**Figur 4:** *Computer aided design*, CAD-sammenstilling av akkumulatorsystemet med cellesegmenter, koblinger mellom segmentene og AMS-systemet. AMS-masteren er kretskortet i venstre frontkammer. Høyre frontkammer er ment for HV-komponenter som bl.a. hovedsikring og strømsensor. De svarte pluggene mellom cellesegmentene, som kalles *maintenance plugs*, er de elektriske koblingene som er tilkoblet under drift. Disse pluggene vil bli fastlåst av lokket, og fungerer både som elektrisk ledende, i tillegg til å bidra til å holde segmentene på plass ved en eventuell velt. Figur hentet fra Autodesk Inventor.

## Nøkkeldata for akkumulatorsystemet

Celledata

$U_{\text{Maks,Celle}}$	$U_{\text{Nom,Celle}}$	$U_{\text{Min,Celle}}$	Kapasitet <sub>Celle</sub>	Energi <sub>Celle</sub>	Masse <sub>Celle</sub>
4.2 VDC	3.6 VDC	2.5 VDC	2500 mAh	10.5 kJ	45.0 g

**Tabell 1:** Nøkkeldata for battericellene [14].

Segmentdata

$U_{\text{Maks,Segment}}$	$U_{\text{Nom,Segment}}$	$U_{\text{Min,Segment}}$	Kapasitet <sub>Segment</sub>	Energi <sub>Segment</sub>	Masse <sub>Segment</sub>
50.4 VDC	43.2 VDC	30.0 VDC	25 Ah	1.26 MJ	7.40 kg

**Tabell 2:** Nøkkeldata for cellesegmentene. Vekten er basert på en CAD-sammenstilling av cellesegmentene med elektronikk og mekanisk sammenstilling.

Akkumulatordata

$U_{\text{Maks,Total}}$	$U_{\text{Nom,Total}}$	$U_{\text{Min,Total}}$	Kapasitet <sub>Total</sub>	Energi <sub>Total</sub>	Masse <sub>Total</sub>
403.2 VDC	345.6 VDC	240 VDC	25 Ah	10.08 MJ	59.2 kg

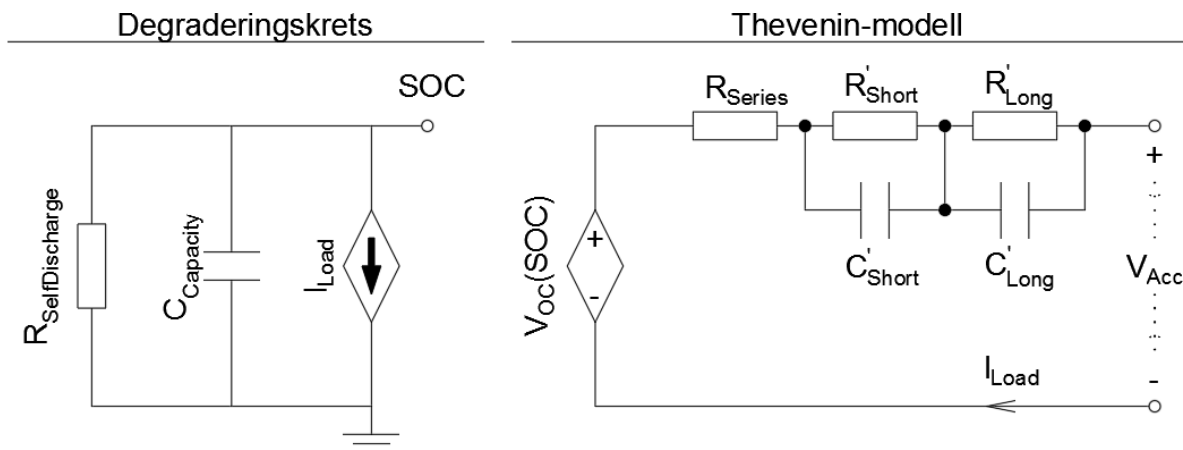
**Tabell 3:** Nøkkeldata for akkumulatorsystemet. Vekten er basert på en komplett CAD sammenstilling av akkumulatoren.

## 1.4 Teori: Estimering av ladestatus

Den mest nærliggende måten å gå frem på når man skal estimere SOC er å tenke på akkumulatorsystemet som en tank med elektrisk ladning som oppfyller  $\sum q_{\text{lagret}} = \sum q_{\text{inn}} - \sum q_{\text{ut}}$ , og dermed at mengden ladning lagret i akkumulatorsystemet kan måles direkte. Denne metoden går ut på å telle antall Coloumb under lading og drift. Dessverre tar ikke antakelsen hensyn til korrigerende faktorer som degraderingen i battericeller etter som de blir eldre, effektivitetsforskjellen mellom lading og drift, eller dynamiske påvirkninger i varierende ytre forhold. Den faktiske prosessen er rett og slett for ulineær til å benytte en så enkel tilnærming.

### Modell av akkumulatorsystemet

Det er nær umulig å fastsette en perfekt modell av akkumulatorsystemet, det er for mange variabler involvert. For å forenkle modellen er det vanlig å sette opp en modell for én celledmodul og anta at utgangspunktet er likt for alle 96 celler. I [2] foreslår forfatterne å dele celledmodellen i to deler: En degraderingskrets for celledspenning basert på SOC og livstidsdegradering, og en utvidet Thevenin-modell. Celledmodellen vises i Figur 5.



**Figur 5:** Cellemodell. Figuren er fra [2, Fig. 2], men med endret benevnelse for enkelte komponenter og signaler.

Degraderingskretsen i Figur 5 er modellert som en kondensator som blir utladet gjennom  $R_{SelfDischarge}$  og en kontrollert strømkilde avhengig av laststrømmen gjennom cellen. Signalet  $SOC$  er ladestatus i cellen fra 0 til 1. Kondensatoren representerer cellekapasiteten, og  $R_{SelfDischarge}$  angir selv-utladningen som foregår uavhengig av laststrømmen. Den utvidede Thevenin-modellen inkluderer en kontrollert spenningskilde avhengig av  $SOC$ , en seriemotstand og to parallelle RC-ledd i serie, som representerer den transiente oppførselen til cellen. Grunnet forskjellige temperaturer lokalt i akkumulatortsystemet kan man ikke anta uniform celleutladning, da seriemotstanden og transient-leddene har dynamiske verdier ut fra celletemperatur [15]. Komponentene/signalene i Figur 5 er:

### Degraderingskrets

$SOC$  - Ladestatus.

$R_{SelfDischarge}$  - Modellert resistans for cellens selv-utladning uavhengig av last.

$C_{Capacity}$  - Modellert kondensator for ladestatus i cellen.

$I_{Load}$  - Strømkilde som representerer utladning, avhengig av laststrømmen.

### Thevenin-modell

$V_{OC}(SOC)$  - Den ubelastede cellespenningen som en funksjon av  $SOC$ .

$V_{Acc}$  - Den målbare spenningen fra cellemodulen.

$I_{Load}$  - Laststrøm.

$R_{Series}$  - Resistans som representerer transientuavhengig internmotstand.

$R'_{Short}$  &  $C'_{Short}$  - RC-ledd som beskriver de korte transientene i systemet.

$R'_{Long}$  &  $C'_{Long}$  - RC-ledd som beskriver de lange transientene i systemet.

## Tilstandsrommodell

I Thevenin-modellen er både  $V_{OC}(SOC)$  og de temperaturavhengige  $R_{Series}$  og transientleddene ulineære, så cellemodellen er en ulineær tidskontinuerlig prosess. Måling av celledspenning og -temperatur skjer gjennom sampling, noe som fører til at estimering av ladestatus må beregnes via en tidsdiskret tilstandsrommodell. En generell tidsdiskret prosess kan uttrykkes på følgende måte [16, seksjon 2.5]:

$$\mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{u}(k), \mathbf{v}(k)) \quad (1.4)$$

$$\mathbf{y}(k) = g(\mathbf{x}(k), \mathbf{u}(k), \mathbf{n}(k)) \quad (1.5)$$

Basert på cellemodellen er det noen antakelser:

- Tilstandsvektoren  $\mathbf{x}(k)$  består av SOC samt spenningene over RC-leddene i Thevenin-modellen,  $V'_{Short}(k)$  og  $V'_{Long}(k)$ .
- Det finnes bare én last:  $I_{Load}$ .
- $\mathbf{v}(k)$  er prosesstøy-vektor med middelerdi  $\hat{\mathbf{v}}(k)$  og autokovariansmatrise  $\mathbf{Q}$ .
- $\mathbf{n}(k)$  er målestøy-vektor med middelerdi  $\hat{\mathbf{n}}(k)$  og autokovariansmatrise  $\mathbf{R}$ .
- Tilstandsbidraget er en ulineær prosess i målevektoren, og erstattes av funksjonen  $\mathbf{H}(\mathbf{x}(k))$ .

Dette fører til endringer fra (1.4) & (1.5):

$$\mathbf{x}(k+1) = f(\mathbf{x}(k), u(k), \mathbf{v}(k)) \quad (1.6)$$

$$\mathbf{y}(k) = g(\mathbf{H}(\mathbf{x}(k)), u(k), \mathbf{n}(k)) \quad (1.7)$$

Oppdaterer med matriser:

$$\mathbf{x}(k+1) = \mathbf{\Phi}(k)\mathbf{x}(k) + \mathbf{\Gamma}(k)u(k) + \mathbf{v}(k) \quad (1.8)$$

$$\mathbf{y}(k) = \mathbf{H}(\mathbf{x}(k)) + E(k)u(k) + \mathbf{n}(k) \quad (1.9)$$

Matrisene i tilstandsrommodellen beregnes av Ohms lov, og er gitt av [15], med komponentnavn fra Figur 5.  $\mathbf{\Phi}$  &  $\mathbf{\Gamma}$  inneholder uttrykk som er avhengige av transientene i systemet, og effekten fra transientene er avhengig av tidssteget  $T(k) = t(k) - t(k-1)$ .

I (1.8):

$$\mathbf{x}(k) = \begin{bmatrix} SOC(k) \\ V'_{Short}(k) \\ V'_{Long}(k) \end{bmatrix} \quad (1.10)$$

$$u(k) = I_{Load} \quad (1.11)$$

$$\mathbf{v}(k) \sim N(\hat{\mathbf{v}}(k), \mathbf{Q}) \quad (1.12)$$

$$\Phi(k) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 - \frac{T(k)}{R'_{Short}C'_{Short}} & 0 \\ 0 & 0 & 1 - \frac{T(k)}{R'_{Long}C'_{Long}} \end{bmatrix} \quad (1.13)$$

$$\Gamma(k) = \begin{bmatrix} -1 \\ \frac{1}{C'_{Short}} \\ \frac{1}{C'_{Long}} \end{bmatrix} T(k) \quad (1.14)$$

I (1.9):

$$\mathbf{y}(k) = V_{Acc}(k) \quad (1.15)$$

$$\mathbf{n}(k) \sim N(\hat{\mathbf{n}}(k), \mathbf{R}) \quad (1.16)$$

$$\mathbf{H}(\mathbf{x}(k)) = V_{OC}(SOC(k)) - \mathbf{D}\mathbf{x}(k) \quad (1.17)$$

$$\mathbf{D} = [0 \quad -1 \quad -1] \quad (1.18)$$

$$E = -R_{series} \quad (1.19)$$

## Kalmanfilter

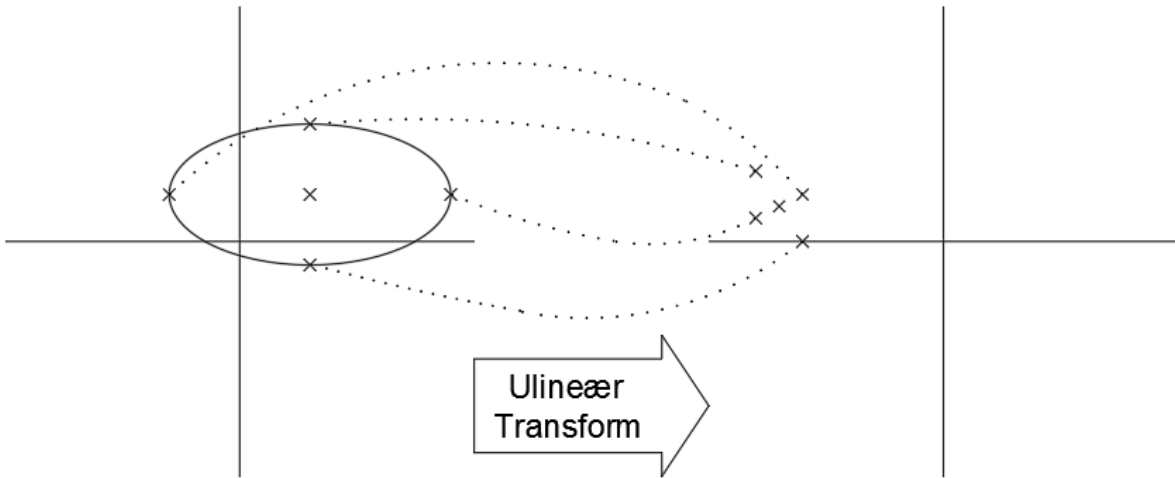
For å estimere ulineære prosesser kan man bruke et utvidet Kalmanfilter, EKF, som estimerer ulineære prosesser ved å linearisere dem med bruk av Taylorrekke før det blir kjørt gjennom et lineært Kalmanfilter. Dette filteret fungerer tilfredsstillende for enkelte prosesser, men estimeringen blir dårligere jo mer ulineær prosessen er [19, seksjon 2.4 - 2.5]. Dette har mesteparten av sitt utspring i at det bare er førsteordens-leddet i Taylorrekken som blir brukt i lineariseringen. EKF er også komputasjonsmessig tungt, med bruk av Jacobi- eller Hessematriser for beregning av kovariansene i systemet. I [15] og [19] foreslår forfatterne å heller bruke et *Unscented Kalman Filter*<sup>3</sup>, UKF, med sigma-punkter.

## Sigma Point Kalman Filter, SPKF

Et Unscented Kalmanfilter med sigma-punkter benytter en skalert unscented-transform med deterministiske valgte punkter. UKF ble først presentert av Julier & Uhlmann i

<sup>3</sup>Forfatteren av [19] kommenterer at han ikke har fått et godt svar på hvorfor filteret kalles *unscented*, så derfor er det heller ikke foreslått en norsk oversettelse

1997 [6]. Ideen er at det er enklere å estimere en Gaussisk sannsynlighetstetthet basert på et visst antall punkter enn å korrekt estimere en ulineær modell/transferfunksjon.



**Figur 6:** Prinsippskisse av unscented-transformen. Figur tilsvarende som [6, Figure 2].

I Figur 6 vises det hvordan fem punkter  $\mathcal{X}_{a;i-1}$  går gjennom den ulineære prosessen og gir nye punkter  $\mathcal{X}_{a;i|i-1}$ . I motsetning til bruk av Monte Carlo-metoden er ikke punktene tilfeldige, men angitt av en spesifikk algoritme. Initielle prosessverdier beregnes ved hjelp av (1.22) - (1.24), og skaleringen av vektningen i SPKF kommer fra (1.25) - (1.29).  $L$  er dimensjonen til  $\hat{\mathbf{x}}_i$ , mens  $\alpha$ ,  $\beta$  &  $\kappa$  er valgte verdier, og er typisk  $0 < \alpha < 1$ ,  $\beta \geq 0$  og  $\kappa \geq 0$  [19, side 56]. Den generelle algoritmen for SPKF som beskrevet i [19, side 60-61] er gjengitt i Algoritme 1, med små adaptasjoner.

- Punktene kalkuleres fra en utvidet tilstandsvektor,  $\hat{\mathbf{x}}_{a;k}$ , og en utvidet kovariansmatrise,  $\mathbf{P}_{\mathbf{x}_{a;k}}$ . Den utvidete tilstandsvektoren og kovariansmatrisen har struktur som vist i (1.20) og (1.21). Det beregnes  $2L + 1$  punkter.
- Som det kommer av notasjonen  $\mathcal{X}_{a;k-1}$  blir de opprinnelige Sigma-punktene beregnet av forrige iterasjon, mens de nye punktene  $\mathcal{X}_{a;k|k-1}$  er et *estimat* av nåverdi. De opprinnelige punktene blir beregnet ut fra forrige tilstandsestimat, med punkter i periferi rundt midtestimatet, som i den ventre halvdel av Figur 6. Avstanden fra midtestimatet til punktene i periferien er standardavviket i estimatet, gitt av den skalerte verdien  $\gamma\sqrt{\mathbf{P}_{\mathbf{x}_{a;k}}}$ .
- $\mathcal{X}_{a;k|k-1}$  brukes til å estimere tilstandsvektor  $\hat{\mathbf{x}}_k$  basert på vektning  $\mathbf{w}_m$  av de forskjellige Sigma-punktene.
- $\mathcal{X}_{a;k|k-1}$  blir så brukt til å beregne Sigma-punktene  $\mathcal{Y}_{k|k-1}$  gjennom funksjonen  $g(\mathbf{H}(\mathcal{X}_{a;k-1}), \mathbf{u}(k), \mathcal{X}_{a;k-1})$
- $\mathcal{Y}_{k|k-1}$  brukes til å estimere målevektor  $\hat{\mathbf{y}}_k$  basert på vektning  $\mathbf{w}_m$  av de forskjellige Sigma-punktene.

- Ved å benytte kovariansmatrisene til tilstand- og målevektor finner man Kalmanforsterkningen, som i (1.30). Denne brukes til å oppdatere  $\hat{\mathbf{x}}_k$  og  $\mathbf{P}_{\mathbf{x}_k}$ , og prosessen er klar for et nytt tidssteg.

Den utvidede tilstandsvektoren og den utvidede kovariansmatrisen har følgende struktur:

$$\hat{\mathbf{x}}_{a;k} = [\hat{\mathbf{x}}_k^T \quad \hat{\mathbf{v}}_k^T \quad \hat{\mathbf{n}}_k^T]^T \quad (1.20)$$

$$\mathbf{P}_{\mathbf{x}_{a;k}} = \begin{bmatrix} \mathbf{P}_{\mathbf{x}_k} & 0 & 0 \\ 0 & \mathbf{Q} & 0 \\ 0 & 0 & \mathbf{R} \end{bmatrix} \quad (1.21)$$

Forventningsverdien til den utvidede tilstanden, og de tilhørende kovariansmatrisene:

$$\hat{\mathbf{x}}_0 = \mathbb{E}[\mathbf{x}_0], \quad \mathbf{P}_{\mathbf{x}_0} = \mathbb{E}[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T] \quad (1.22)$$

$$\hat{\mathbf{v}}_0 = \mathbb{E}[\mathbf{v}_0], \quad \mathbf{Q} = \mathbb{E}[(\mathbf{v}_0 - \hat{\mathbf{v}}_0)(\mathbf{v}_0 - \hat{\mathbf{v}}_0)^T] \quad (1.23)$$

$$\hat{\mathbf{n}}_0 = \mathbb{E}[\mathbf{n}_0], \quad \mathbf{R} = \mathbb{E}[(\mathbf{n}_0 - \hat{\mathbf{n}}_0)(\mathbf{n}_0 - \hat{\mathbf{n}}_0)^T] \quad (1.24)$$

Skaleringsparametrene i SPKF:

$$\lambda = \alpha^2(L + \kappa) - L \quad (1.25)$$

$$\gamma = \sqrt{L + \lambda} \quad (1.26)$$

$$w_{m;0} = \frac{\lambda}{L + \lambda} \quad (1.27)$$

$$w_{c;0} = \frac{\lambda}{L + \lambda} + (1 - \alpha^2 + \beta) \quad (1.28)$$

$$w_{m;k} = w_{c;k} = \frac{1}{2(L + \lambda)}, \quad k = 1, \dots, 2L \quad (1.29)$$

Kalmanforsterkningen beregnes ut fra:

$$\mathbf{K}_k = \mathbf{P}_{\mathbf{x}_k \mathbf{y}_k} \mathbf{P}_{\mathbf{y}_k}^{-1} \quad (1.30)$$



**Algoritme 1:** Unscented Kalmanfilter med Sigma-punkter

---

Initialiserer basert på (1.22)-(1.29):

$$\hat{\mathbf{x}}_0, \hat{\mathbf{v}}_0, \hat{\mathbf{n}}_0, \mathbf{P}_{\mathbf{x}_0}, \mathbf{Q}, \mathbf{R}, \lambda, \gamma, \mathbf{w}_m, \mathbf{w}_c$$

for k from 1 to  $\infty$  do

Oppdaterer utvidede matriser:

$$\hat{\mathbf{x}}_{a;k-1} = \begin{bmatrix} \hat{\mathbf{x}}_{k-1} \\ \hat{\mathbf{v}}_{k-1} \\ \hat{\mathbf{n}}_{k-1} \end{bmatrix}, \quad \mathbf{P}_{\mathbf{x}_{a;k-1}} = \begin{bmatrix} \mathbf{P}_{\mathbf{x}_{k-1}} & 0 & 0 \\ 0 & \mathbf{Q} & 0 \\ 0 & 0 & \mathbf{R} \end{bmatrix}$$

Kalkulasjon av Sigma-punkter:

$$\mathcal{X}_{a;k-1} = \left[ \hat{\mathbf{x}}_{a;k-1} \quad \hat{\mathbf{x}}_{a;k-1} + \gamma \sqrt{\mathbf{P}_{\mathbf{x}_{a;k-1}}} \quad \hat{\mathbf{x}}_{k-1} - \gamma \sqrt{\mathbf{P}_{\mathbf{x}_{a;k-1}}} \right]$$

Oppdatering av tilstandsestimat:

$$\mathcal{X}_{x;k|k-1} = f(\mathcal{X}_{x;k-1}, \mathbf{u}_k, \mathcal{X}_{v;k-1})$$

$$\hat{\mathbf{x}}_k = \sum_{i=0}^{2L} w_{m;i} \mathcal{X}_{\mathbf{x};k|k-1}$$

$$\mathbf{P}_{\hat{\mathbf{x}}_k} = \sum_{i=0}^{2L} w_{c;i} (\mathcal{X}_{\mathbf{x};k|k-1} - \hat{\mathbf{x}}_k) (\mathcal{X}_{\mathbf{x};k|k-1} - \hat{\mathbf{x}}_k)^T + \mathbf{Q}$$

Oppdatering av måleestimat:

$$\mathcal{Y}_{k|k-1} = g(\mathbf{H}(\mathcal{X}_{x;k-1}), \mathbf{u}(k), \mathcal{X}_{n;k-1})$$

$$\hat{\mathbf{y}}_k = \sum_{i=0}^{2L} w_{m;i} \mathcal{Y}_{i,k|k-1}$$

$$\mathbf{P}_{\mathbf{y}_k} = \sum_{i=0}^{2L} w_{c;i} (\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k) (\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k)^T + \mathbf{R}$$

$$\mathbf{P}_{\mathbf{x}_k \mathbf{y}_k} = \sum_{i=0}^{2L} w_{c;i} (\mathcal{X}_{\mathbf{x};k|k-1} - \hat{\mathbf{x}}_k) (\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k)^T$$

$$\mathbf{K}_k = \mathbf{P}_{\mathbf{x}_k \mathbf{y}_k} \mathbf{P}_{\mathbf{y}_k}^{-1}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k + \mathbf{K}_k (\mathbf{y}_k - \hat{\mathbf{y}}_k)$$

$$\mathbf{P}_{\mathbf{x}_k} = \mathbf{P}_{\hat{\mathbf{x}}_k} - \mathbf{K}_k \mathbf{P}_{\mathbf{y}_k} \mathbf{K}_k^T$$

end

---

## 1.5 Utstyr

De forskjellige programvarene benyttet er:

- Altium Designer 19  
Altium brukes til design av kretskort.  
Programvaren er sponset.
- Autodesk Inventor Professional 2020  
Inventor er det valgte programmet for CAD av deler.
- Autodesk Vault Professional 2020  
Vault er benyttet av ION Racing som en felles database for modeller.  
Programvaren er sponset.
- Mathworks MATLAB R2018a  
MATLAB er benyttet til estimeringen av ladestatus med Kalmanfilter.

Både MATLAB og Inventor er programmer som er tilgjengelig for UiS-studenter.

# Kapittel 2

## Konstruksjon

Dette kapitlet omhandler design og konstruksjon av kretskortene i slave-delen av batteriovervåningssystemet og tilpassingen av algoritmen for unscented kalmanfilter med sigma-punkter til bruk i akkumulatorsystemet.

### 2.1 Oppbygning av slave-systemet

Akkumulatorsystemet består av 96 seriekoblede cellemoduler á 10 parallellkoblede battericeller. Disse overvåkes av batteriovervåningssystemet som skal koble ut HV-systemet til bilen ved overtredelse av gitte parametre. Dette gjøres i form av et master/slave-system, hvor datainnhenting blir foretatt av slaven, mens masteren foretar databehandling, viftestyring, kommunikasjon med de øvrige komponentene i bilen og sikkerhetsfunksjoner. AMS må fungere under både utlading og opplading. Selv om regelverket tillater regenerering av energi under bremsing er denne funksjonen ikke implementert pr. 2020.

AMS mastermodulen kommuniserer med slavene over isolert SPI. Som nevnt i [7] var dette et valg tatt på bakgrunn av tilgang til eksisterende slavemoduler som brukte isoSPI. De gamle slavemodulene var bygd opp av et multikort-system, hvor selve battericellene var festet i et *Printed Circuit Board*, PCB, og kommunikasjonen var på et overliggende kort. Det nederte kortet benyttet skinner i aluminium som var festet i kretskortet som strømvei. Spenningsmåling ble gjort direkte fra celletilkoblingen, og temperaturmåling ble gjort med termistorer på den negative celletilkoblingen. Disse slavene var fungerende, men hadde enkelte problemer:

- Temperaturmålingene var svært støybelagte, og det ble antatt at det var en dårlig løsning på avlesningen.
- Kretskortene var unødvendig kompakte, og det gikk på bekostning av montasje.
- De gamle slave-modulene benyttet måledata fra celler via to konnektorer per PCB. Under kjøring ble det så store vibrasjoner at noen av konnektorene ble ødelagt.

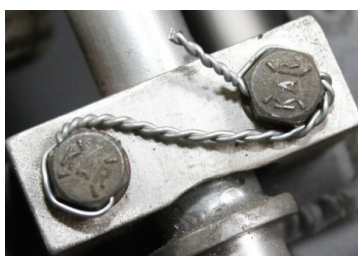
I utgangspunktet ble det vurdert at hver AMS slave skulle bestå av et flerlagskort hvor topplaget ble brukt til kommunikasjon og de andre lagene utgjorde strømveier, men det var uforenlig med enten strømføringsevne, avstandskrav eller fysisk utforming. Det ble derfor valgt å gå for en lik løsning som tidligere, med to kort i høyden og aluminiumsskinner som ledere i HV-kretsen. Det øverste kortet, slave-kortet, vil da foreta all kommunikasjon med masteren mens det nederste, layout-kortet, står for tilkoblingen til celleterminalene samt at det mekanisk samler cellemodulene i ønsket mønster.

## Utforming av cellesegmentene

De eksisterende slavemodulene var ment for avlesning av poseceller, som har et annet design og en svært forskjellig montering enn de nye cellemodulene brukt i 2020. Hovedargumentet for å gå for en annen celletype enn poseceller var at det var vanskelig å montere i henhold til regelverket. Det er nemlig vanskelig å oppnå positiv låsing av poseceller uten å sveise dem fast, hvilket er svært brannfarlig og i grenseland på hva man kan anse som forsvarlig praksis. Der poseceller har to celletabber som må sveises e.l. har cellemodulene gjengede M5 terminaler. Dette gir flere valg for montering med positiv låsing. Figur 7 viser de forskjellige låsemetodene, og Tabell 4 viser hvordan de forskjellige aktuelle låsemetodene ble vurdert. Skiver med låsetabber ble ansett som det beste alternativet, til tross for at det var det dyreste alternativet, da disse har laveste risiko for kortslutning under montering. En annen fordel med cellemodulene er at de kommer med temperatursensor på hver celle fra leverandør, og dermed trengs ingen ekstra tiltak for å være i henhold til regelverket, EV 5.8.3.

Alternativer for positiv låsing av cellemoduler			
	Låsetråd	Låseplate	Skiver med låsetabber
Tilgjengelighet	Enkelt å få tak i.	Enkelt å få tak i.	Må bestilles fra utlandet.
Gjenbruk	Kan ikke gjenbrukes.	Kan gjenbrukes.	Kan gjenbrukes.
Montering	Vanskelig å montere. Høy fare for kortslutning.	Enkelt å montere. Kan ved uforsiktighet forårsake kortslutning.	Enkelt å montere. <b>Liten fare for kortslutning.</b>
Verifisering av korrekt installasjon	Enkelt å verifisere med visuell inspeksjon.	Hver skrue må verifiseres ved visuell inspeksjon.	Hver skrue må verifiseres ved visuell inspeksjon.
Pris	Ca. 50 kr.	Ca. 100 kr.	Ca. 250 kr.

**Tabell 4:** Tabellen viser vurderingen av de tre aktuelle metodene for positiv låsing av cellemoduler.



(a) Låsetråd.

Figur hentet fra <https://camauto.com/image/cache/catalog/other-racing-parts/lockwire-800x800.jpg>



(b) Låseplate.

Figur hentet fra [https://nordic.kramp.com/pics/watermarked/2/2992/2992-691179\\_h.jpg](https://nordic.kramp.com/pics/watermarked/2/2992/2992-691179_h.jpg).



(c) Skive med låsetabber.

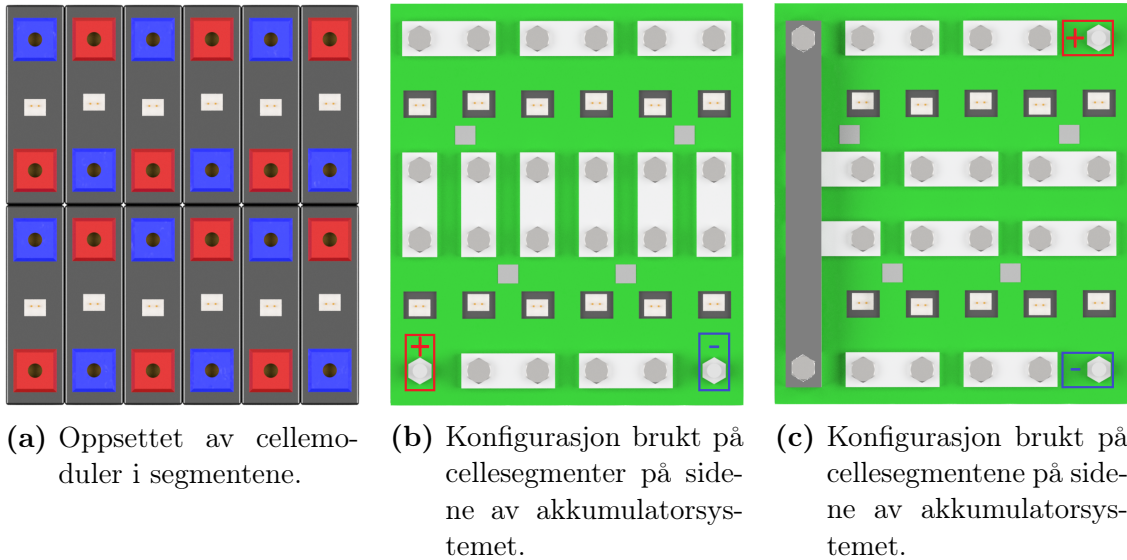
Figur hentet fra [https://images-na.ssl-images-amazon.com/images/I/31k35ApL-EL.\\_SR600%2C315\\_PIWhiteStrip%2CBottomLeft%2C0%2C35\\_SCLZZZZZZZ\\_.jpg](https://images-na.ssl-images-amazon.com/images/I/31k35ApL-EL._SR600%2C315_PIWhiteStrip%2CBottomLeft%2C0%2C35_SCLZZZZZZZ_.jpg)

**Figur 7:** Alternativer for positiv låsing av cellemoduler. Låsetråd tres gjennom et forborret hull i skruehodene, og tvinnes. Med låseplate bøyes enden av platen opp for å holde skruehodet på plass. På skive med låsetabber bøyes den ene tabben rundt godset til det skruen er festet i, og den andre tabben bøyes opp på skruehodet.

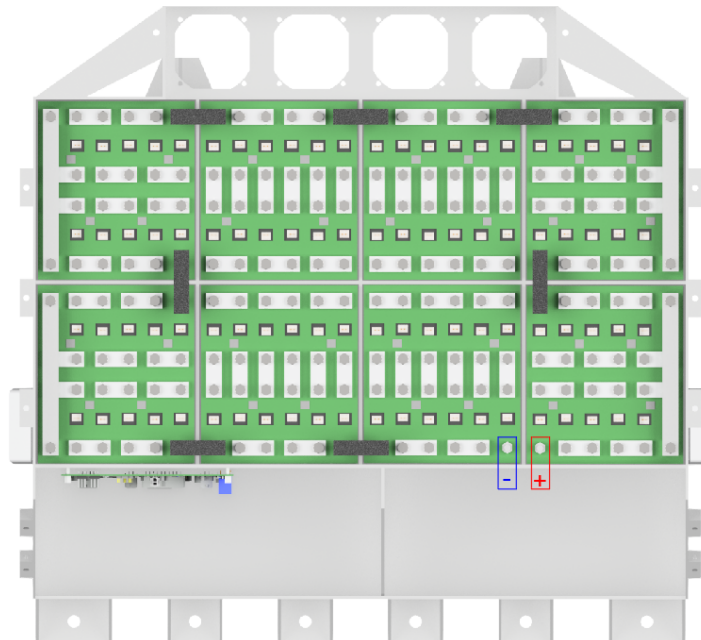
### Layout-kort

Cellesegmentene består av 12 cellemoduler i serie. Segmentene seriekobles så med hverandre for å få den ønskede spenningen i akkumulatorsystemet. For å få ønsket strømvei gjennom akkumulatoren har segmentene to forskjellige koblingsmønstre. Ved å gjøre det på denne måten er det fysisk umulig å feilkoble internt og man unngår HV-kabler som krysser over cellesegmenter, og dermed er sårbare for mekaniske belastninger. Oppbygningene av cellesegmentene vises i Figur 8a, og strømveien i akkumulatorsystemet vises i Figur 9. Tillegg A inneholder en beskrivelse av oppbygningen til hele akkumulatorsystemet, mens Tillegg B inneholder skjematikken til slave-modulene. I Figur 8b og 8c vises det ene av kretskortene som inngår i AMS-slavesystemet. Dette kortet kalles layout-kortet og er et to-lags PCB. Kortet er designet for å kunne brukes på begge segmentoppbygninger, hvor topplaget er i bruk i Figur 8b og bunnlaget er i bruk i Figur 8c. Kortet har utskjæringer til konnektorer for temperaturavlesning på cellemodulene. Ideelt sett skulle disse være tilkoblet en *Surface Mount Device*, SMD-konnektor på layout-kortet, men da disse bygde 2 mm for høyt til at kortet kom i kontakt med terminalene på cellemodulene ble det inngått et kompromiss i form av utskjæringer til konnektorer med ledninger som går til det øverste kretskortet i systemet, slave-kortet. Layout-kortet har fire 6-pin konnektorer til det andre kortet. Disse konnektorene har samme plassering på begge sider av layout-kortet og er plassert asymmetrisk både i x-

og y-akse for å unngå feilmontering.



**Figur 8:** CAD-sammenstillinger av de to forskjellige konfigurasjonene av cellesegmenter. I Figur 8c er den ene busbaren markert i en annen farge, fordi den krysser over de andre. Figurer hentet fra Autodesk Inventor.



**Figur 9:** Fugleperspektiv av CAD-sammenstilling av akkumulatørsystemet, uten AMS-slaver, for å vise strømveien i akkumulatørsystemet. Mest negativ og mest positiv tilkobling er markert. Figur hentet fra Autodesk Inventor.

## Slave-kort

Det øverste PCB i slavesystemet, selve slavekortet, er vist i Figur 10. Slave-kortet er designet for å være enkelt utskiftbart for alle segmenter, hvor man bare adresserer kortet, kobler til temperatursensorer og plasserer kortet over layout-kortet via konnektorer. Kortet er et firelagskort, og er bygd opp av flere funksjoner/komponenter:

- I midten er det to *Integrated Circuit*, IC-er, én for spenningsmåling og én for temperaturmåling.

IC-ene er LTC6804-2, batteriovervåkningschiper fra Linear Technology. Disse IC-ene bruker 290  $\mu$ s på å lese spenningen for hele cellesegmentet. I tillegg krever IC-ene ikke ekstern strømforsyning, men benytter en Wakeup-funksjon når AMS-master gir beskjed om å aktiveres [8]. Temperaturmåling foretas også som en spenningsmåling, i tråd med cellemodulenes datablad [3]. Det er kun data fra temperatur- og spennings-IC som sendes til AMS-master. B

- Kortet adresseres ved hjelp av to *Dual In-line Package*-brytere, SW1 og SW2.

Begge bryterne stilles inn likt, men adresseringen for temperatur-IC har negativ logikk. Segmentene nummereres 0-7, så segment nr. 2 vil ha adressene 0010 og 1101.

- De tolv markerte områdene er måledata/-behandling for hver celle.

Alle de tolv områdene er like. De består av en overlasterikring, en MOSFET og diverse motstander. Signalene ut fra måleområdet går direkte til sine respektive IC-er. Måleområdene er konstruert i samsvar med datablad for cellemodulene [3].

- J1 og J2 er konnektorer for isoSPI.

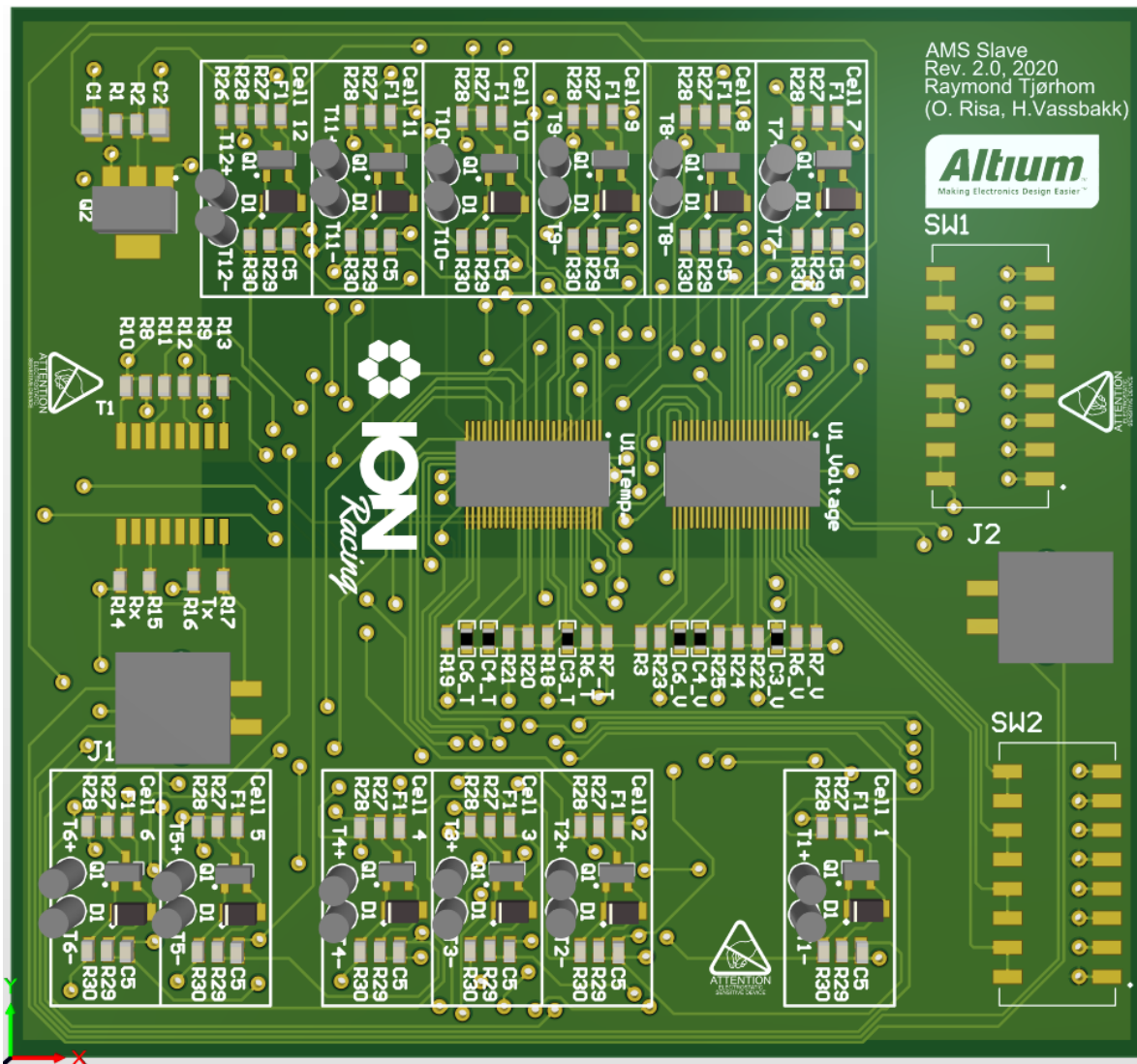
Planen med å ha to konnektorer for isoSPI, er at de går inn og ut av kortet, og gjør feilsøking mer fleksibelt.

- T1, plassert på venstre side av kortet er isolasjonstransformator.

Denne konverterer isoSPI  $\leftrightarrow$  SPI. Ved å koble den regulerte forsyningsspenning "Vreg" til inngang "ISOMD" på IC-ene blir de satt opp til å kommunisere over SPI.

- Konnektorene på bunnet av kretskortet er av type Molex 79109-1002.

Disse ble valgt på bakgrunn av at konnektorene fra layout-kortet burde være SMD, og at 79109-1002-konnektorene har gode testresultater mht. til- og frakobling. Under tester fra Molex har disse blitt målt til en minimum frakoblingstyngde til 7.31 lbs (3.32 kg) [5, Sequence V]



**Figur 10:** CAD-modell av slavekortet. Konnektorer til layout-kortet er de eneste komponentene som er plassert på bunnlaget. Figur hentet fra Altium Designer.

Kretskortene som inngår i slave-modulene er designet i Altium Designer 19. Full skjematikk finnes i Tillegg B.



## 2.2 Kalmanfilter

For å estimere ladestatus i racerbilen skal det benyttes et unscented Kalmanfilter med Sigma-punkter. Algoritme 1 er den generelle algoritmen for dette systemet. Når cellemodellen fra Figur 5 skal implementeres er det med enkelte endringer i forhold til Algoritme 1. Algoritme 2 side 26 angir implementeringen av cellemodellen.

- Det antas at prosess- og målestøyvektorene,  $\mathbf{v}$  &  $\mathbf{n}$ , er hvit støy, og at dermed:

$$\hat{\mathbf{v}}_k = \mathbb{E}[\mathbf{v}_k] = 0 \quad (2.1)$$

$$\hat{\mathbf{n}}_k = \mathbb{E}[\mathbf{n}_k] = 0 \quad (2.2)$$

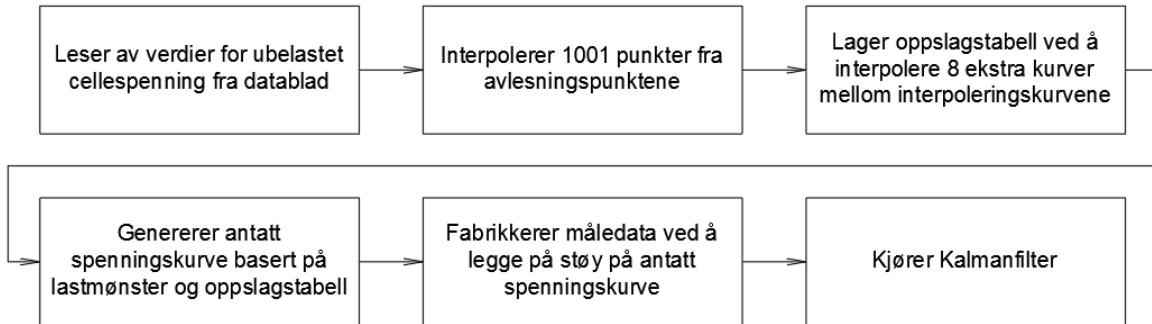
Dette fører til:

$$\hat{\mathbf{x}}_{a;k} = \begin{bmatrix} \hat{\mathbf{x}}_k \\ 0 \\ 0 \end{bmatrix} \quad (2.3)$$

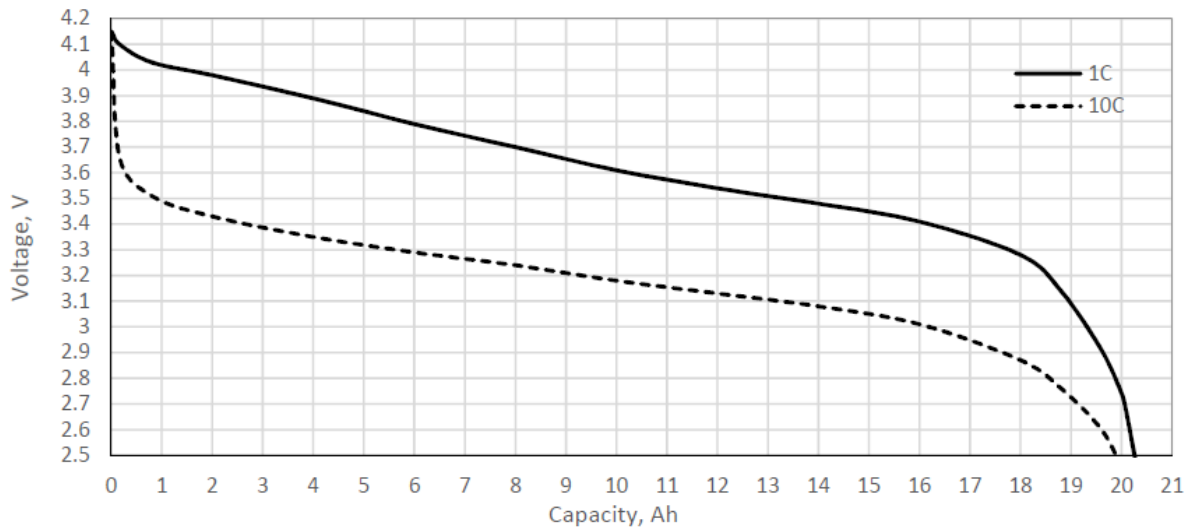
- Fra (1.8) må  $\Phi$  og  $\Gamma$  implementeres i oppdateringen av tilstandestimatet.
- Fra (1.9) må  $\mathbf{D}$  implementeres i oppdateringen av måleestimatet.
- Autokovariansmatrisen for prosesstøy,  $\mathbf{Q}$ , inngår i kovariansmatrisen for tilstandsestimatet,  $\mathbf{P}_{\hat{\mathbf{x}}_k}$ .
- Autokovariansmatrisen for målestøy,  $\mathbf{R}$ , inngår i kovariansmatrisen for måleestimatet,  $\mathbf{P}_{\mathbf{y}_k}$ .
- Det er regnemessig krevende å beregne kvadratroten av en matrise. Da er det bedre å benytte Cholesky-dekomposisjonen, som foreslått i både [6, side 478] og [19, side 52]. Cholesky-dekomposisjonen beregner Cholesky-faktoren,  $L$ , i  $A = LL^T$ , hvor  $A$  er en Hermitisk matrise. En Hermitisk matrise er en kvadratisk matrise hvis transponerte kompleks konjugerte matrise,  $A^{*T}$ , er lik  $A$ . På bakgrunn av dette gir Cholesky-dekomposisjonen  $L = \sqrt{A}$  [1, side 25][13].

Funksjonen  $V_{OC}(SOC)$  som inngår i  $\mathbf{H}(\mathbf{x}(k))$ , (1.17), må enten beregnes etter målte verdier, eller ut fra informasjon oppgitt i datablad. I databladet oppgir produsenten av cellemodulene to spenningskurver under utladning med forskjellig strømtrekk, vist i Figur 12. Det ble vurdert at en oppslagstabell med interpolerte punkter var en relativt enkel måte å estimere  $V_{OC}(SOC)$ . Tabellen er interpolert ut fra punkter på kurvene i Figur 12 samt at det er interpolert punkter mellom kurvene som gir antatte utladekurver for forskjellige belastninger mellom 1C og 10C. Oppslagstabellen er en  $10 \times 1001$  tabell, hvor radene er C-rate fra 1C til 10C og kolonnene er oppløsningen på kurvene. Det er bevisst valgt grov oppløsning på utladekurven, da en sjåfør ikke har behov for mer detaljert ladestatus enn i hele prosent, og  $1/1000$  oppløsning vil gi godt nok grunnlag

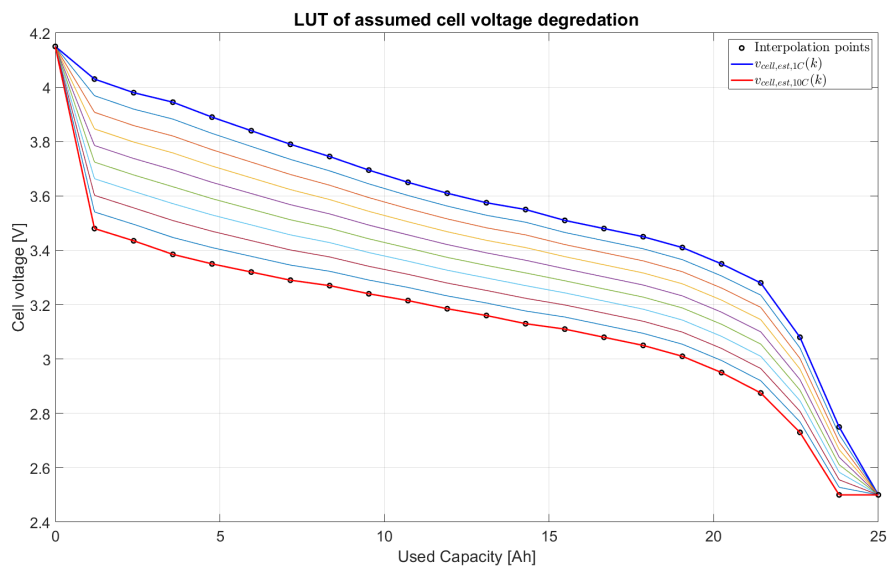
for estimering. De interpolerte kurvene vises i Figur 13. Flytskjema for prosessen vises i Figur 11.



Figur 11: Flytskjema over spenningsestimering.



Figur 12: Utladepkurver for cellemoduler fra datablad [3]. Kurven viser at det er forskjell i spenningen under forskjellige utladesituasjoner.



**Figur 13:** Interpolerte utladekurver for  $V_{OC}(SOC)$  under forskjellig last, basert på Figur 12. De opprinnelige avlesningspunktene er markert i figuren. Den øverste kurven representerer utladekurven under 1C last, den neste representerer 2C osv. til den nederste kurven som representerer 10C last.

---

**Algoritme 2:** Unscented Kalmanfilter med Sigma-punkter for cellemodell
 

---

**Initialiserer:**

$$\mathbf{x}_0, \mathbf{P}_{\mathbf{x}_0}, \mathbf{Q}, \mathbf{R}, \lambda, \gamma, \mathbf{w}_m, \mathbf{w}_c, \Phi, \Gamma, \mathbf{D}, E, \alpha, \beta, \kappa, L, V_{OC}(SOC)$$

**for** k from 1 to  $\infty$  **do**

Oppdaterer utvidede matriser:

$$\hat{\mathbf{x}}_{a;k-1} = \begin{bmatrix} \hat{\mathbf{x}}_{k-1}^T & 0 & 0 \end{bmatrix}^T$$

$$\mathbf{P}_{\mathbf{x}_a;k-1} = \text{diag}(\mathbf{P}_{\mathbf{x}_{k-1}}, \mathbf{Q}, \mathbf{R})$$

$$\mathbf{P}_{\mathbf{x}_{chol};k-1} = \text{chol}(\mathbf{P}_{\mathbf{x}_a;k-1})$$

Kalkulasjon av Sigma-punkter:

$$\mathcal{X}_{a;k-1} = \begin{bmatrix} \hat{\mathbf{x}}_{a;k-1} & \hat{\mathbf{x}}_{a;k-1} + \gamma \mathbf{P}_{\mathbf{x}_{chol};k-1} & \hat{\mathbf{x}}_{k-1} - \gamma \mathbf{P}_{\mathbf{x}_{chol};k-1} \end{bmatrix}$$

Oppdatering av tilstandsestimat:

$$\mathcal{X}_{\mathbf{x};k|k-1} = \Phi \mathcal{X}_{\mathbf{x};k-1} + \Gamma u(k) + \mathcal{X}_{\mathbf{v};k-1}$$

$$\hat{\mathbf{x}}_k = \sum_{i=0}^{2L} w_{m;i} \mathcal{X}_{\mathbf{x};k|k-1;i}$$

$$\mathbf{P}_{\hat{\mathbf{x}}_k} = \sum_{i=0}^{2L} w_{c;i} (\mathcal{X}_{\mathbf{x};k|k-1;i} - \hat{\mathbf{x}}_{k;i}) (\mathcal{X}_{\mathbf{x};k|k-1;i} - \hat{\mathbf{x}}_{k;i})^T + \mathbf{Q}$$

Oppdatering av måleestimat:

$$\mathbf{H}_{\mathbf{x};k|k-1} = V_{OC}(SOC(k)) - \mathbf{D} \mathcal{X}_{\mathbf{x};k|k-1}$$

$$\mathcal{Y}_{k|k-1} = \mathbf{H}_{\mathbf{x};k|k-1} + E u(k) + \mathcal{X}_{\mathbf{n};k|k-1}$$

$$\hat{\mathbf{y}}_k = \sum_{i=0}^{2L} w_{m;i} \mathcal{Y}_{k|k-1;i}$$

$$\mathbf{P}_{\mathbf{y}_k} = \sum_{i=0}^{2L} w_{c;i} (\mathcal{Y}_{k|k-1;i} - \hat{\mathbf{y}}_{k;i}) (\mathcal{Y}_{k|k-1;i} - \hat{\mathbf{y}}_{k;i})^T + \mathbf{R}$$

$$\mathbf{P}_{\mathbf{x}_k \mathbf{y}_k} = \sum_{i=0}^{2L} w_{c;i} (\mathcal{X}_{\mathbf{x};k|k-1;i} - \hat{\mathbf{x}}_{k;i}) (\mathcal{Y}_{k|k-1;i} - \hat{\mathbf{y}}_{k;i})^T$$

$$\mathbf{K}_k = \mathbf{P}_{\mathbf{x}_k \mathbf{y}_k} \mathbf{P}_{\mathbf{y}_k}^{-1}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k + \mathbf{K}_k (\mathbf{y}_k - \hat{\mathbf{y}}_k)$$

$$\mathbf{P}_{\mathbf{x}_k} = \mathbf{P}_{\hat{\mathbf{x}}_k} - \mathbf{K}_k \mathbf{P}_{\mathbf{y}_k} \mathbf{K}_k^T$$

**end**


---

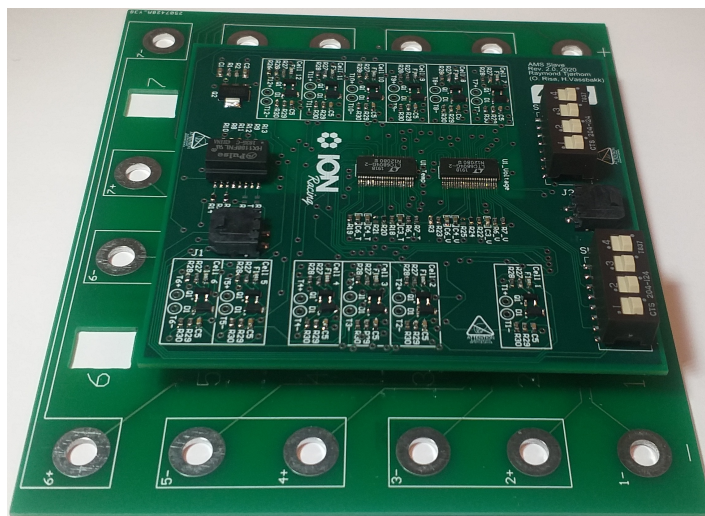
# Kapittel 3

## Resultater

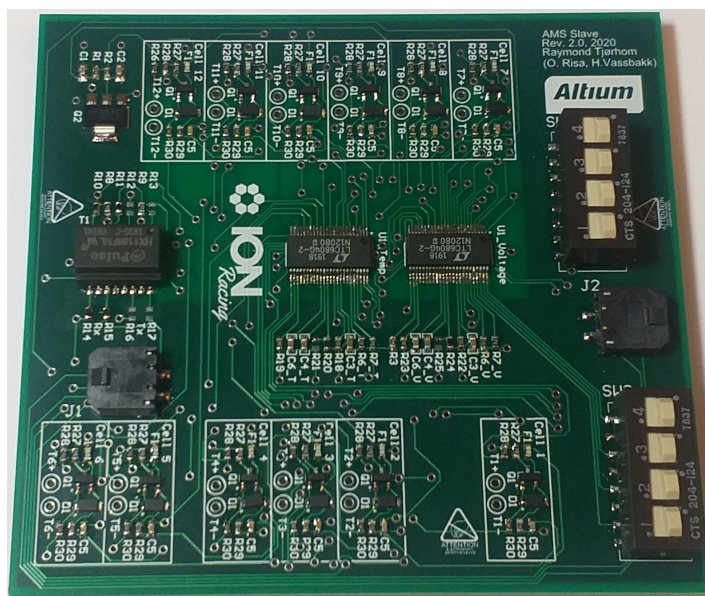
Dette kapittelet viser resultatene oppnådd i konstruksjonen av AMS-slavene og estimeringen av State of Charge med bruk av unscented kalmanfilter med sigma-punkter.

### 3.1 Kretskort i slave-modulen

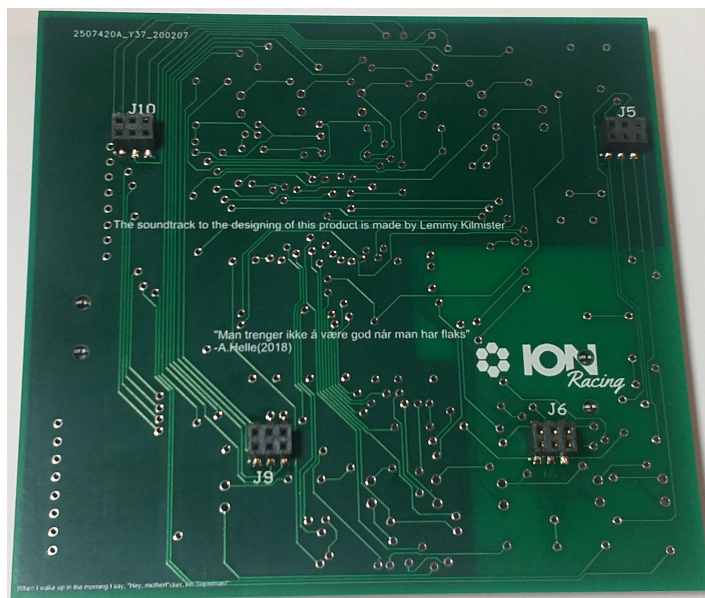
Kretskortene til slave-modulene ble designet i programmet Altium Designer 19, og deretter bestilt fra ekstern produsent. UiS har muligheten til å frese kretskort, men det forutsetter at det er et tolags-kort samt at topp og bunnlag ikke er isolert. Dette er ikke kompatibelt med bruksområdet til ION Racing. Ved å bestille kretskort fra en ekstern produsent får man også muligheten til et silketrykk på topplaget, med valgfri utforming. Dette laget er de hvite markeringene som har informasjon om komponentnummer og kortinfo, samt noen utvalgte morsomheter, som er en tradisjon i Formula Student-miljøet. Kretskortene er håndloddet.



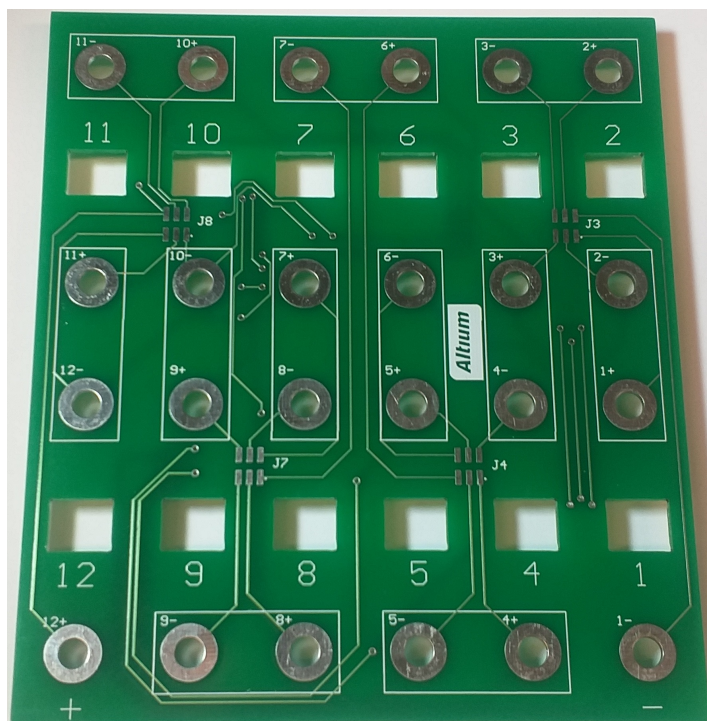
Figur 14: Skråperspektiv av sammenstilling av slave-PCB



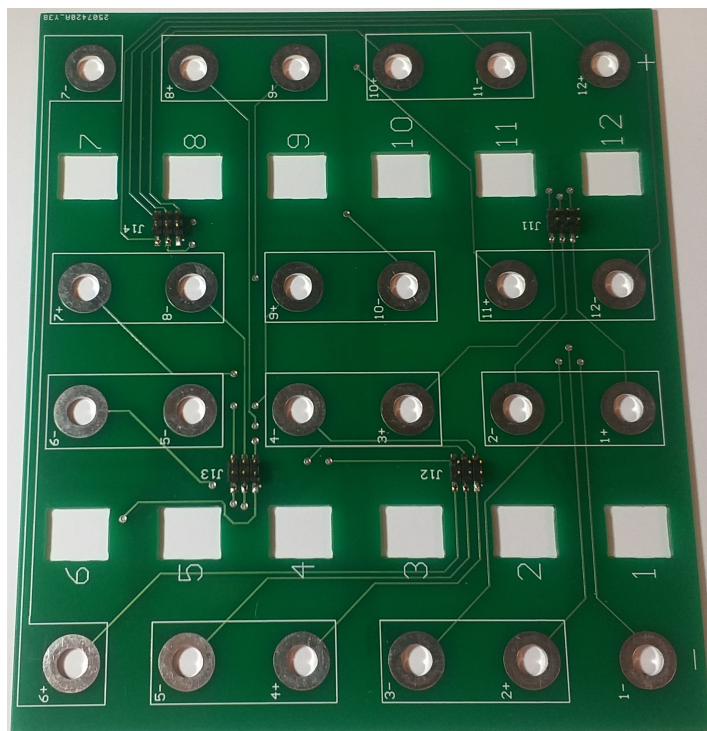
Figur 15: Topplaget til slave-PCB.



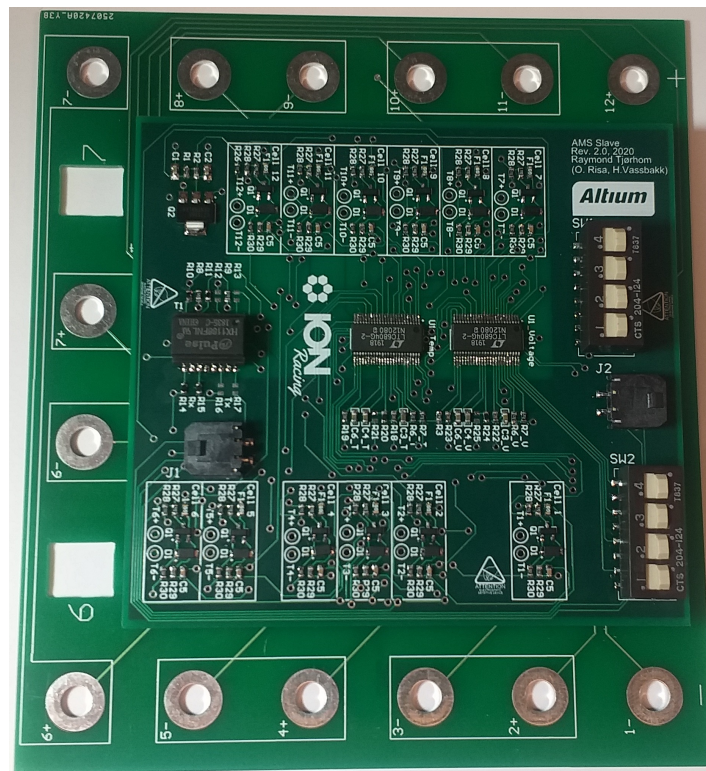
Figur 16: Bunnlaget til slave-PCB. Konnektorene er type Molex 79109-1002.



Figur 17: Fugleperspektiv av layout-kort: indre versjon.



Figur 18: Fugleperspektiv av layout-kort: ytre versjon.



Figur 19: Fugleperspektiv av sammenstilling av slave-PCB.

## 3.2 Simuleringer i MATLAB

Resultatene presentert i denne seksjonen kommer fra forskjellige simuleringer av estimate-ring av SOC under forskjellige lastmønstre. Simuleringer av ladestatusen i akkumulator-systemet er utført i Mathworks MATLAB. Tillegg C inneholder Matlab-scriptet brukt for simuleringer. Programmet bygger på Algoritme 2, med det krever litt forberedelser:

- Celldata må initialiseres.

$V_{max}$ ,  $V_{min}$ , Kapasitet.

- Lastdata må initialiseres.

Lastmønster,  $R_{Series}$ ,  $R'_{Short}$ ,  $C'_{Short}$ ,  $R'_{Long}$ ,  $C'_{Long}$  og  $T$ .

- Antall nødvendige iterasjoner beregnes. Dette beregnes ut fra formelen:

$$\text{Iterasjoner} = \frac{3600}{\text{Lastmønster} * T} \quad (3.1)$$

- Oppslagstabell (*Look-up table*, LUT) for degradering av cellespenning beregnes.



Denne interpoleres ut fra utladningskurve i datablad, [3].

- Antatt spenningsdata beregnes.

Dette beregnes fra lastmønster og LUT for degradering av celledspenning

- Måledata fabrikkere på bakgrunn av antatt spenningsdata.

Antatt spenningsdata pålegges hvit støy. Støymengden er satt til å ha en *signal-to-noise ratio*, SNR på 40.

- Implementerer Algoritme 2

Dersom spenningen over en cellemodul blir  $\leq 2.5$  V anses denne cellen som utladet, og programmet stopper.

## Valgte verdier

I matrisene i Kalmanfilteret er det visse verdier som er antatt:

$$R_{Series} = 2.7 \mu\Omega \quad (3.2)$$

$$R'_{Short} = 0.5 \Omega \quad (3.3)$$

$$C'_{Short} = 10 \text{ kF} \quad (3.4)$$

$$R'_{Long} = 0.7 \Omega \quad (3.5)$$

$$C'_{Long} = 10 \text{ kF} \quad (3.6)$$

$$T = 290 \mu\text{s} \quad (3.7)$$

Disse er antatt med bakgrunn i verdier fra [15] avlest ved 20 °C, med visse justeringer.

## Simuleringsresultater med tre cellemoduler

I simuleringene gjengitt i Figur 20 - 23 er det forskjellige lastmønstre:

- Konstant 1C last

Under konstant last skal SOC utlades lineært. Som det kommer frem i Figur 20b bruker modellen litt tid på å justere seg inn.

- Rolig lastmønster

I det rolige mønsteret er lastkurven lav under utladning, mens den gir høye regenereringer.

- Middels hardt lastmønster

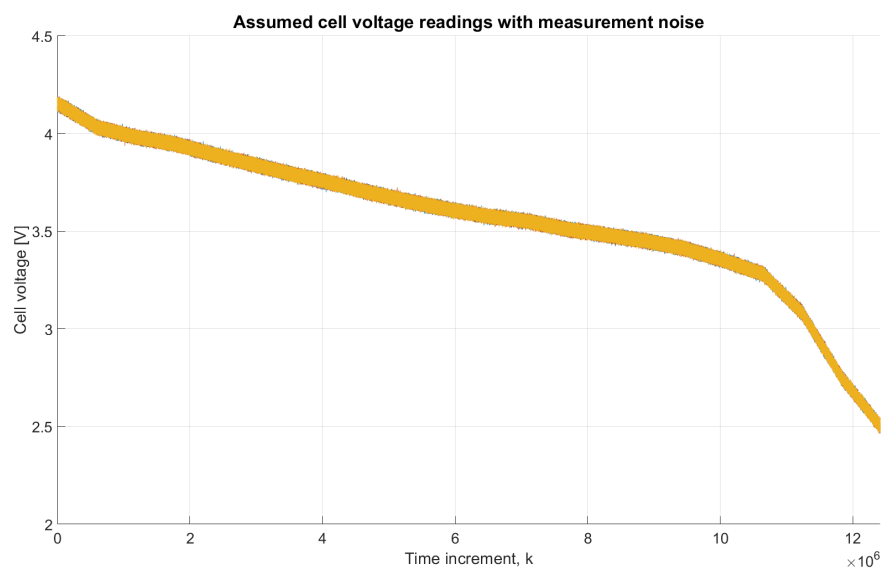
I det middels harde mønsteret er det antatt at maks last er 5C. Lastmønsteret går tidvis under null, som representerer regenerering av kinetisk energi under brems.

- Hardt lastmønster

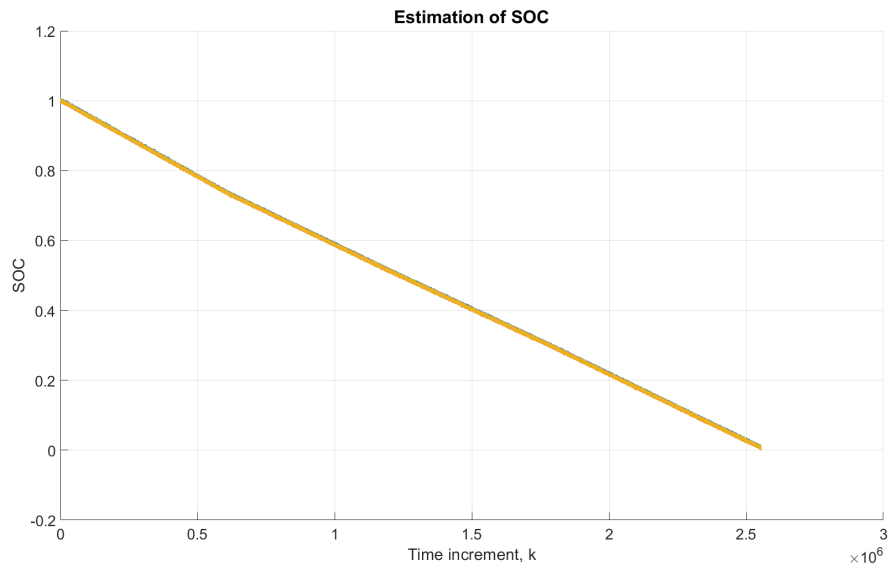
Det harde mønsteret har en dobling av verdiene fra der middels harde mønsteret.

Kun de viktigste kurvene fra simuleringene blir gjengitt her, men Tillegg D inneholder samtlige kurver fra testene.

### Konstant 1C last



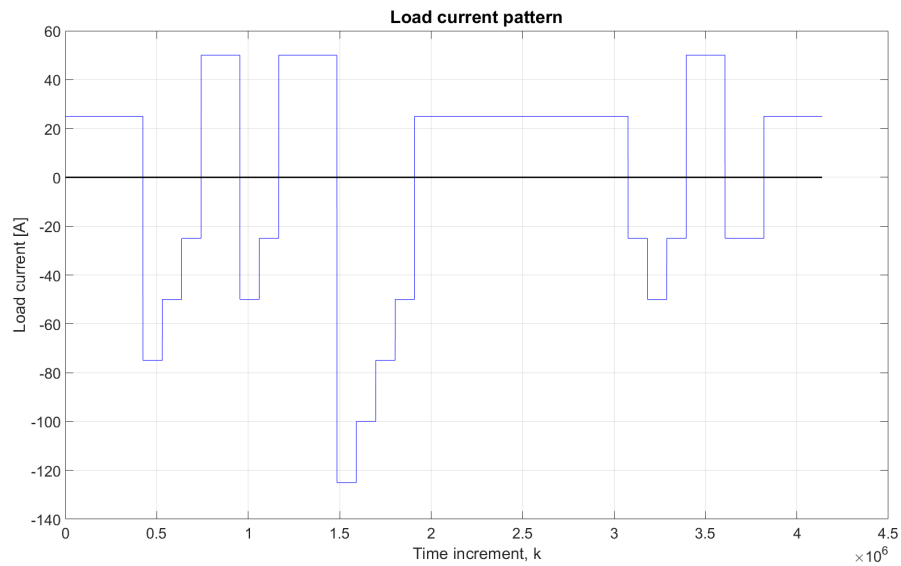
(a) Spenningskurve med hvit støy for de tre cellene.



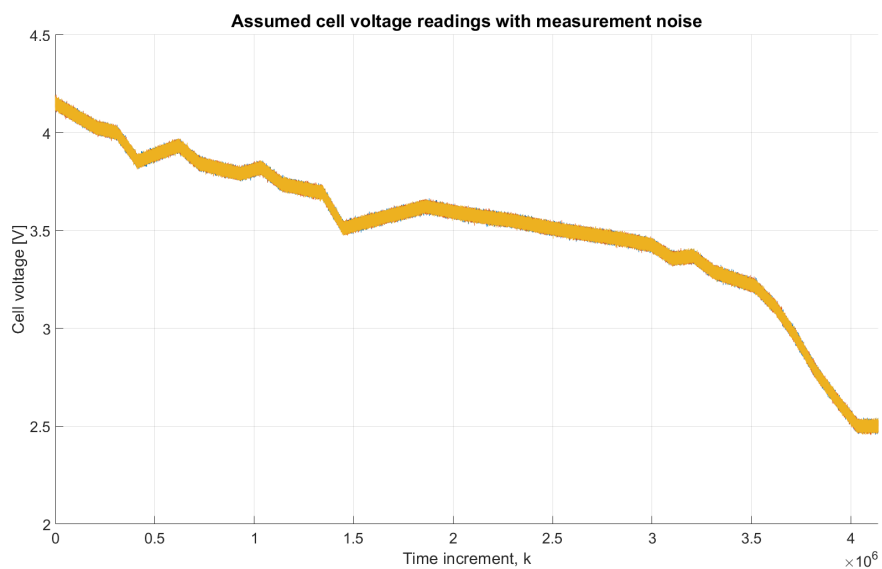
(b) Estimering av SOC for de tre cellene fra Kalmanfilter

**Figur 20:** Kurver fra simulering med konstant 1C last.

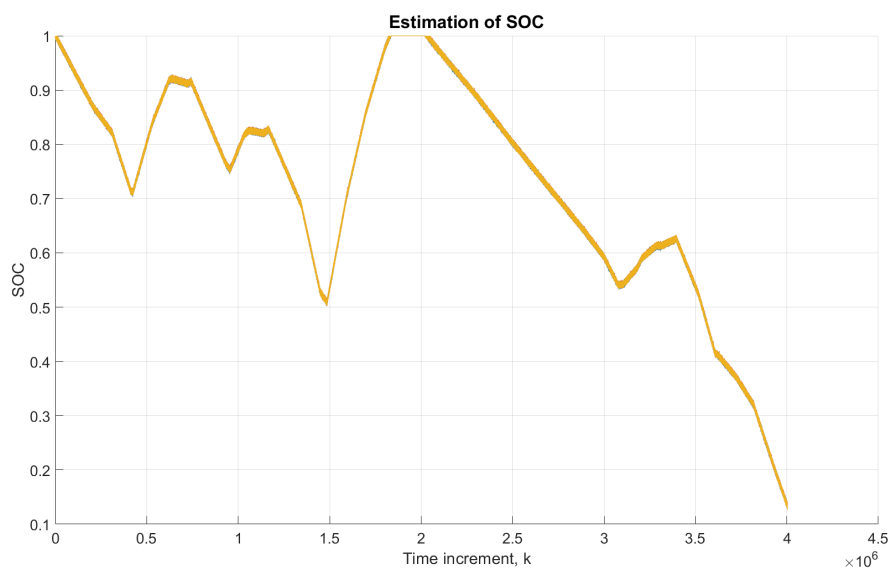
### Rolig lastmønster



(a) Lastkurve.



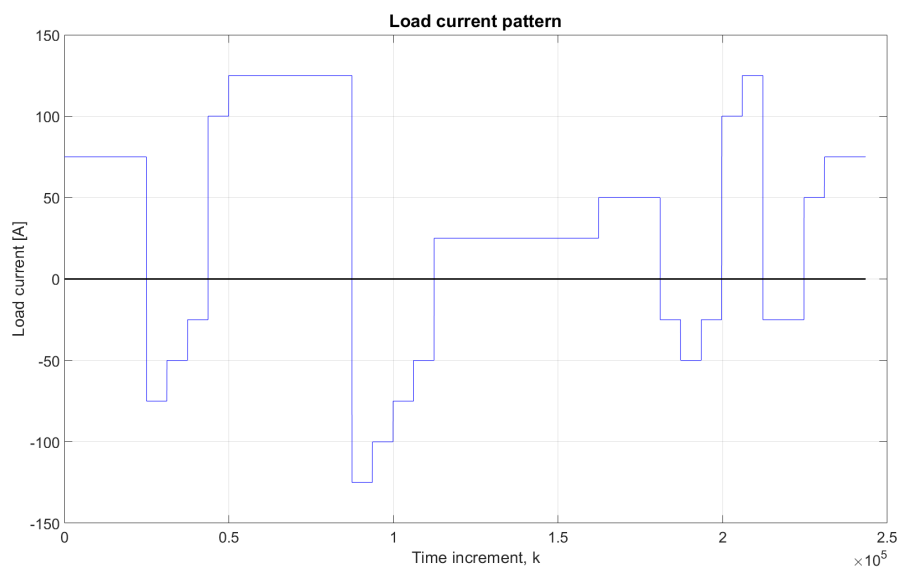
(b) Spenningskurve med hvit støy for de tre cellene.



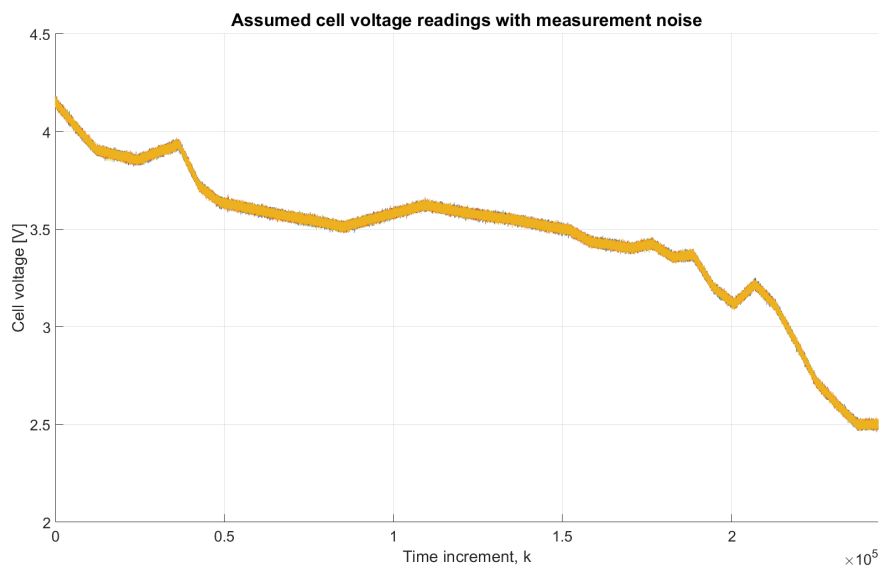
(c) Estimering av SOC for de tre cellene fra Kalmanfilter

**Figur 21:** Kurver fra simulering med rolig lastmønster.

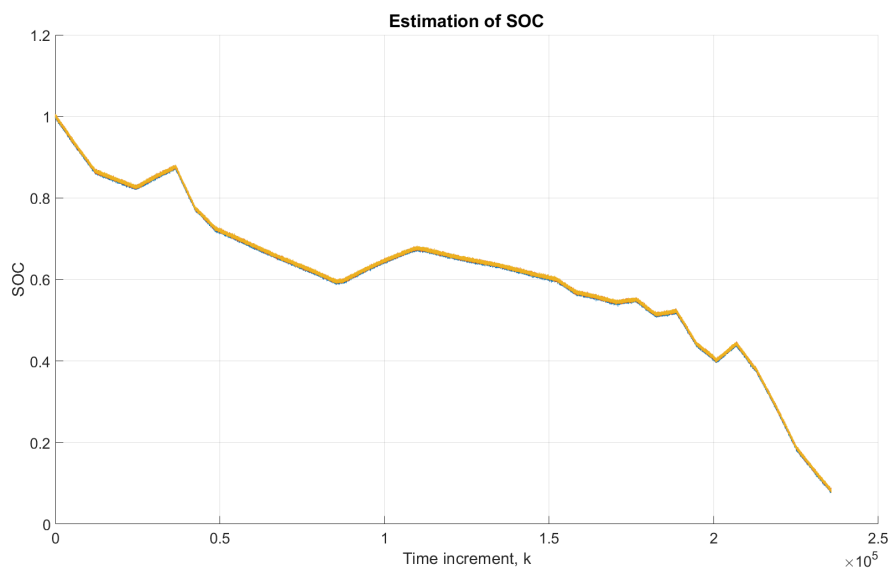
## Middels hardt lastmønster



(a) Lastkurve. Tilnærmet likt mønster som i Figur 23a, men med halverte størrelser.



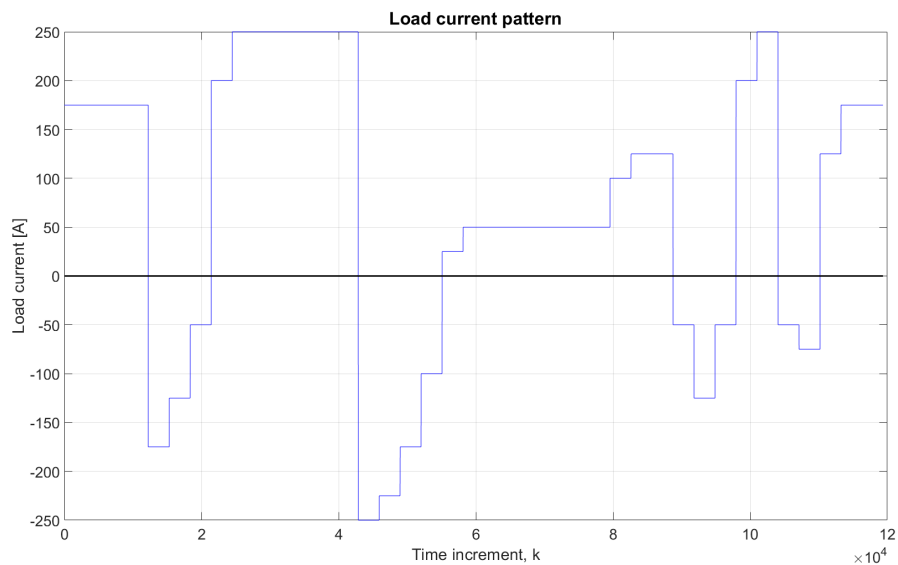
(b) Spenningskurve med hvit støy for de tre cellene.



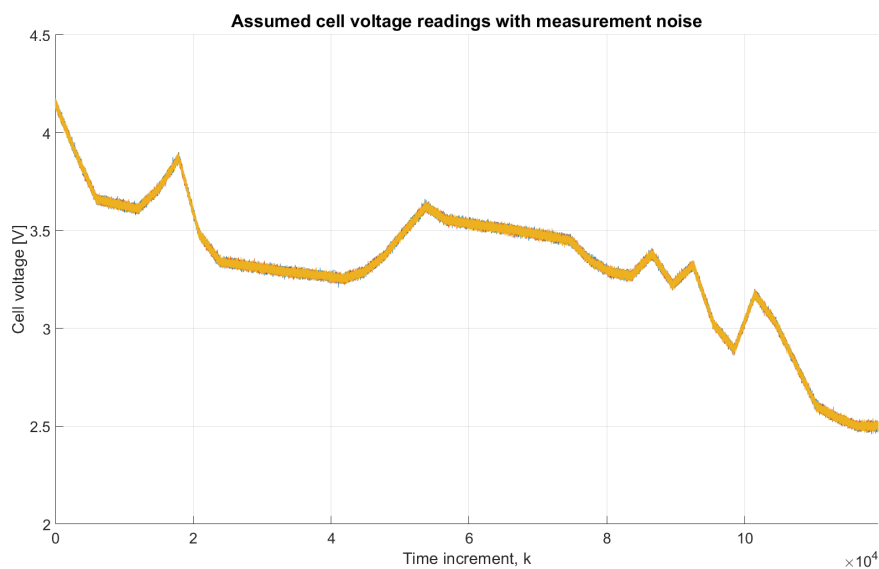
(c) Estimering av SOC for de tre cellene fra Kalmanfilter

**Figur 22:** Kurver fra simulering med middels hardt lastmønster. SOC rekker ikke å bli null før programmet avsluttes. Dette kommer av at estimeringen av SOC ikke rekker å svinge seg inn til riktig verdi i et så kort program.

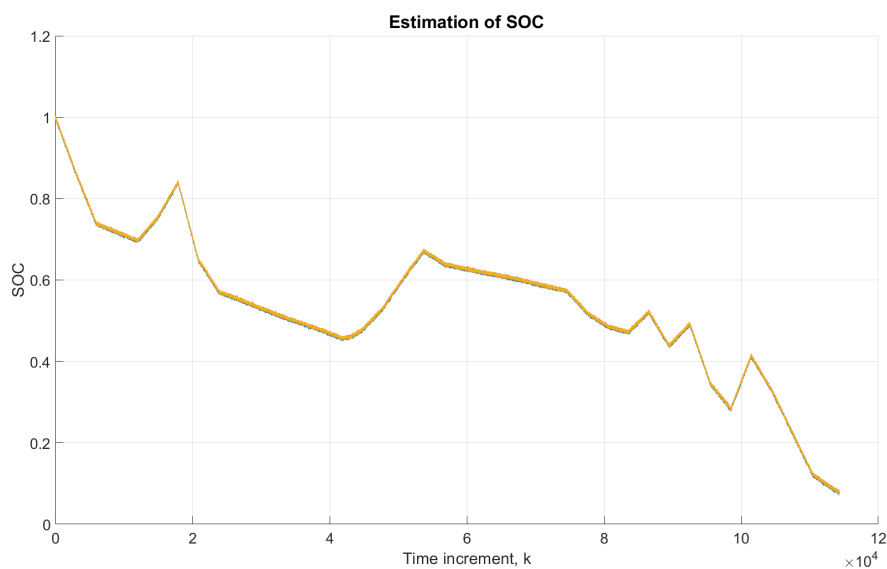
### Hardt lastmønster



(a) Lastkurve. Kurven er tenkt som et mulig lastmønster under banekjøring med implementert regenerering.



(b) Antatt spenningskurve med hvit støy for de tre cellene.



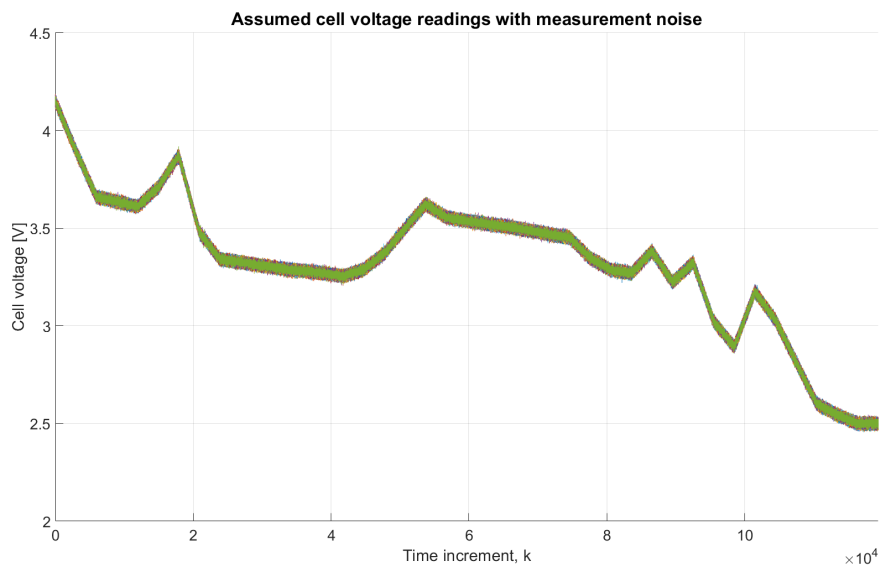
(c) Estimering av SOC for de tre cellene fra Kalmanfilter

**Figur 23:** Kurver fra simulering med hardt lastmønster. SOC rekker ikke å bli null før programmet avsluttes. Dette kommer av at estimeringen av SOC ikke rekker å svinge seg inn til riktig verdi i et så kort program.

## Simuleringsresultater med 96 celledmoduler

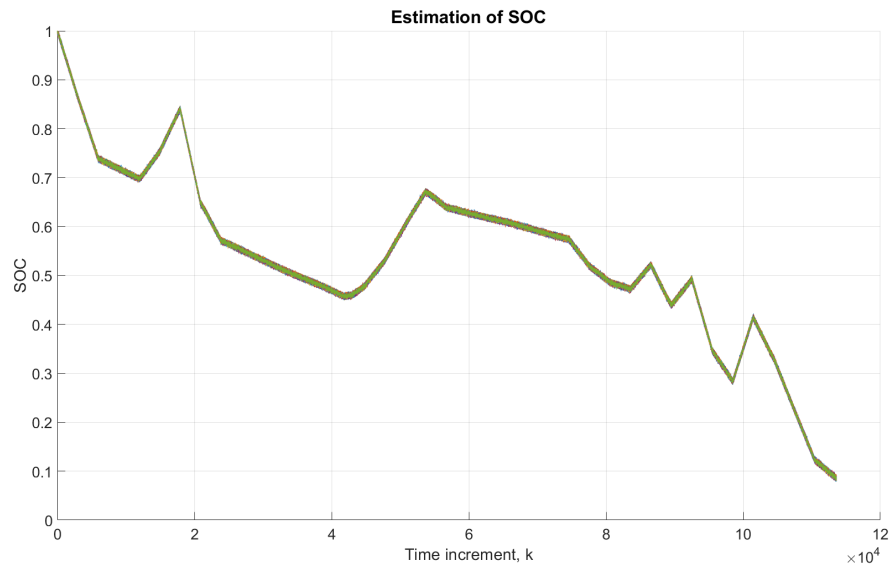
Denne testen er foretatt med hardt lastmønster, som vist i Figur 21a.

### Hardt lastmønster med 96 celledmoduler



(a) Spenningskurve med hvit støy.

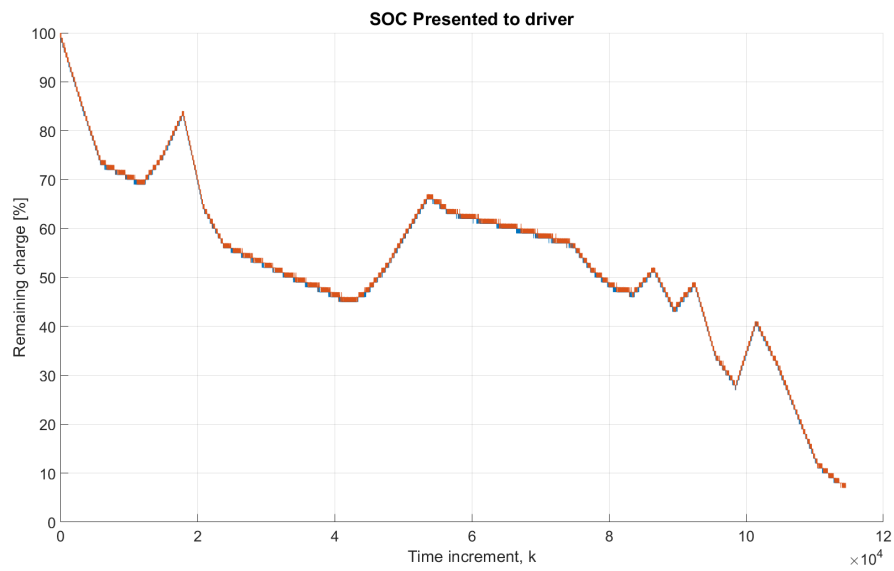




(b) Estimering av SOC fra Kalmanfilter

**Figur 24:** Kurver fra simulering med hardt lastmønster på 96 celledmoduler.

Testen er utført for å teste effektiviteten til Kalmanfilteret med full akkumulator. Under hardt lastmønster brukte MATLAB ca. 581 sekunder på Kalmanfiltreringen som gikk over 113605 iterasjoner. Dette tilsvarer 5.11 ms per gjennomgang i akkumulatoren. Eller  $53.3 \mu\text{s}$  per celle. IC-en avleser spenning pr.  $290 \mu\text{s}$  [8], hvilket betyr at det er et etterslep på beregningene. Et alternativ vil være å kun vurdere én celle, og anta uniform utladning. I Figur 25 vises det hvordan Kalmanfilteret estimerer laveste celle under estimering med 96 celler og 1 celle med samme datasett. Under estimering med én celle brukte MATLAB ca. 6.2 sekunder, hvilket tilsvarer ca  $54.3 \mu\text{s}$  per iterasjon. På bakgrunn av resultatene med 1 celle og 96 celler antas det at Kalmanfilteret kan brukes på et mindre utvalg celler, gitt at det ikke er for store forandringer mellom miljøet til cellene.



**Figur 25:** Estimering av ladestatus med 96 celler vs. 1 celle.

# Kapittel 4

## Diskusjon og Konklusjon

Kretskortene som inngår i AMS-slavene er ferdig designet, produsert og loddet, og simuleringer i MATLAB ble utført på estimering av SOC, men på bakgrunn av de ekstraordinære omstendighetene rundt covid-19 ble flere prosesser i oppgaven satt på vent, og det var ikke mulig å fullføre alle deloppgavel/-mål som planlagt.

### Batteriovervåkningssystemet

Fra tidligere var master-modulen til AMS ferdigstilt [7]. Slave-modulene ble konstruert som en del av denne masteroppgaven. Kretskortene har et relativt enkelt design med utskiftbarhet som et nøkkelmoment. Gjennomtenkte løsninger på layout-kortet, med forskjellig design på topp- og bunnlag, har gjort at samme kort kan brukes for flere ulike cellekonfigurasjoner og har redusert det nødvendige antall forskjellige PCB fra tre til to. Adresseringsløsningen til slavekortet gjør også at et eventuelt defekt kort raskt kan byttes ut med et reservekort uten å måtte foreta hardwareendringer på reservekortet før installasjon. Systemet ser ut til å være et godt utarbeidet system, som har få mangler.

### Estimering av ladestatus

Simuleringsresultatene fra MATLAB tyder på at fremgangsmåten for estimering av ladestatus med Sigma-punkter virker lovende. Kurvene fra Figur 20 - 24 viser at algoritmen er i stand til å estimere ladestatus til forventede verdier ut fra kjøremønster. Under testen med 96 celledmoduler på hardt kjøremønster hadde algoritmen ikke tilfredsstillende beregningshastighet. Det var det derimot når det bare ble beregnet én celle, og det var akseptable avvik i estimering av ladestatus fra full akkumulator. Det kan også vurderes å estimere et utvalg fra hvert celsegment. Eventuelt kan også estimeringen foretas på full akkumulator hvert  $n$ -te tidsinkrement, hvor  $n$  bestemmes etter nærmere testing.

### Koronavirus

Covid-19 påvirket fremgangen i denne oppgaven i svært stor grad. Til tross for at kretskortene var ferdig produsert i starten av februar, ble leveransen av kretskortene til slave-modulene utsatt i mer enn en måned, og de kom frem kun få dager før regjeringen innførte "...de mest inngripende Norges befolkning har opplevd i fredstid." [Helseminister Bent Høie, 12. mars 2020].

#### Påvirkningen på slave-modulene

På de få dagene mellom leveranse av slave-modulene og nedstengningen av det norske samfunnet ble en prototype av slavemodulene håndloddet. Planen var å foreta en sammenstilling av et cellesegment og etablere kommunikasjon med master-modulen så snart det var mulig, men med manglende komponentleveranse var det ikke anledning for å teste systemet. Det er dermed usikkert hvorvidt resultatene er mindre støybefengte enn tidligere år.

#### Påvirkningen på utviklingen av estimeringsalgoritmen

I cellemodellen fra Figur 5, i Thevenin-modellen er det visse motstands- og kondensatorverdier som trengs for å foreta estimering av ladestatus. Etter planen skulle disse verdiene måles ved hjelp av sponset testing fra et eksternt firma i slutten av mars, men dette lot seg naturligvis ikke gjøre. Etter oppheving av koronatiltak har det foreløpig ikke vært aktuelt for firmaet å foreta sponsede tester. Motstands- og kondensatorverdiene brukt i simuleringene er derfor antatt ut fra verdier fra [15], med visse justeringer.

### Sammenstilling og veien videre

Dessverre medførte omstendighetene rundt koronaviruset til at delene som trengtes for å ferdigstille cellesegmentene ikke kom frem i tide, og systemet har ikke blitt testet opp mot AMS-master. Det neste steget i prosessen ville vært å etablere kommunikasjon mellom master og slave, for deretter å vurdert avlesningsresultatene, før algoritmen for estimering av ladestatus blir implementert som et sanntidsprogram i AMS-masteren. Det er forventet at denne overgangen ikke er elementær, og at spesielt sanntidsoppdateringen av måledata vil være utfordrende.

### Konklusjon

På bakgrunn av den enkle, oversiktlige sammenstillingen av kretskortene og resultatene fra ladestatus-simuleringene vurderes det til at oppgaven med å ferdigstille batteriovervåkningssystem og utvikle en effektiv algoritme for estimering av ladestatus i en racerbil er vellykket, til den grad det er mulig på tidspunktet for denne rapporten.

# Bibliografi

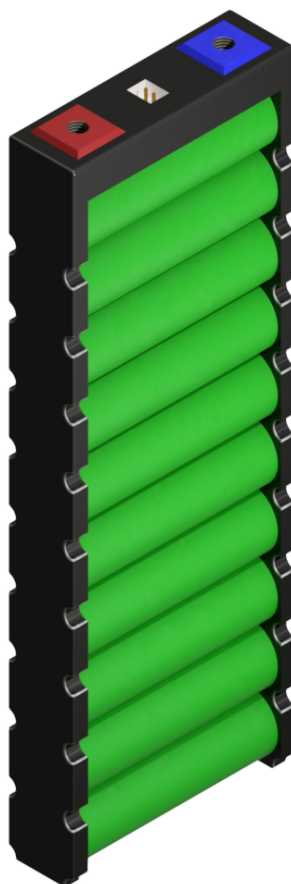
- [1] E. Bédos. Rom og lineæritet, 2012. URL <https://www.uio.no/studier/emner/matnat/math/nedlagte-emner/MAT4000/v13/undervisningsmateriale/RL-1og2-1.pdf>. Accessed: 2020-06-11.
- [2] M. Chen and G. A. Rincón-Mora. Accurate electrical battery model capable of predicting runtime and I-V performance. *IEEE Transactions on Energy Conversion*, 21(2):504–511, 2006. doi: 10.1109/TEC.2006.874229.
- [3] Energus Power Solutions. *Li-ion building block Li8P25RT*. Energus Power Solutions, k edition, Oktober 2016. Generelt datablad for cellemoduler fra Energus, mottatt pr. epost.
- [4] Formula Student Germany. Formula student rules 2020, 2020. URL [https://www.formulastudent.de/fileadmin/user\\_upload/all/2020/rules/FS-Rules\\_2020\\_V1.0.pdf](https://www.formulastudent.de/fileadmin/user_upload/all/2020/rules/FS-Rules_2020_V1.0.pdf).
- [5] J.SCHAFFER. *TEST SUMMARY 2 mm DUAL ROW VERTICAL RECEPTACLE ASSEMBLIES*. Molex Inc., a edition, Desember 1998.
- [6] S. J. Julier and J. K. Uhlmann. New extension of the Kalman filter to nonlinear systems. *Proceedings of the SPIE*, 3068:182–193, Juli 1997. doi: <https://doi.org/10.1117/12.280797>.
- [7] E. B. Knudsen and R. Tjørhom. Hardware, communication and graphical user interface for the accumulator management system in an electric race car. Report, University of Stavanger, Stavanger, 2019.
- [8] Linear Technology Corporation. *LTC6804-1/LTC6804-2 Multicell Battery Monitors*. Linear Technology Corporation, 2013. URL <https://www.analog.com/media/en/technical-documentation/data-sheets/680412fc.pdf>.
- [9] MIT Electric Vehicle Team. A guide to understanding battery specifications, December 2008. URL [http://web.mit.edu/evt/summary\\_battery\\_specifications.pdf](http://web.mit.edu/evt/summary_battery_specifications.pdf). Accessed: 2020-02-21.
- [10] Norsk Elbilforening. Feirer 10 prosent elbiler i Norge, 2020. URL <https://elbil.no/feirer-10-prosent-elbiler-i-norge/>. Accessed: 2020-02-17.

- [11] Norsk Elbilforening. Elbilbestand, 2020. URL <https://elbil.no/elbilstatistikk/elbilbestand/>. Accessed: 2020-02-17.
- [12] Opplysningsrådet for veitrafikken. Kjøretøybestanden per 31. desember 2019, 2020. URL <https://ofv.no/kjoretoybestanden/>. Accessed: 2020-02-17.
- [13] Rosetta Code. Cholesky decomposition, 2020. URL [https://rosettacode.org/wiki/Cholesky\\_decomposition](https://rosettacode.org/wiki/Cholesky_decomposition). Accessed: 2020-06-11.
- [14] Samsung SDI Co., Ltd. Energy Business Division. *Lithium-ion rechargeable cell for power tools - Model name : INR18650-25R*. Samsung SDI Co., Ltd., 1.0 edition, March 2014. Accessed via <https://www.energusps.com/shop/product/li1x10p25rt-li-ion-building-block-with-temp-sensor-3-6v-25ah-15c-198?category=3>: 2019-10-28.
- [15] Y. Shen. Accurate electrical battery model capable of predicting runtime and I-V performance. *Electrochimica Acta*, 283(1):1432–1440, 2018. doi: <https://doi.org/10.1016/j.electacta.2018.07.078>.
- [16] K. Skretting. Notat om tidsdiskrete systemer., December 2008. URL <http://www.ux.uis.no/~karlsk/ELE620/notat1.pdf>. Accessed: 2020-02-24.
- [17] Språkrådet. Årets ord: sakte-tv, 2013. URL <https://www.sprakradet.no/Vi-og-vart/hva-skjer/Aktuelt/2013/Arets-ord-sakte-tv/>. Accessed: 2020-02-17.
- [18] UniTek. *Manual, Digital Three-Phase Servo Amplifier, BAMOCAR-PG-D3*. UniTek, v05 edition, 2019. URL [https://www.unitek-industrie-elektronik.de/images/pdf/BAMOCAR%20Digital/BAMOCAR-PG-D3\\_EN.pdf](https://www.unitek-industrie-elektronik.de/images/pdf/BAMOCAR%20Digital/BAMOCAR-PG-D3_EN.pdf).
- [19] R. van der Merwe. *Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models*. PhD thesis, Oregon Health & Science University, April 2004.

# Tillegg A

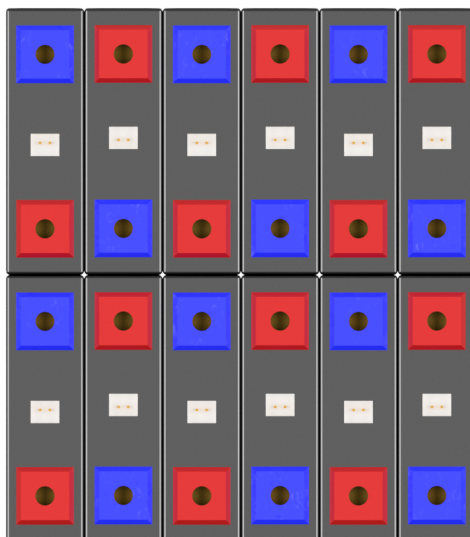
## Oppbygning av akkumulatorsystemet

I dette tillegget blir oppbygningen til akkumulatorsystemet presentert via bilder.

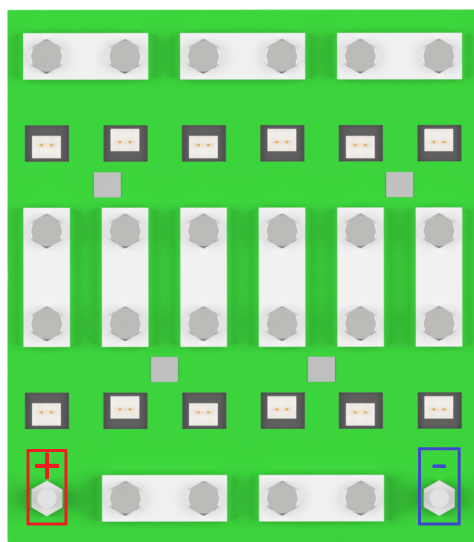


**Figur A.1:** CAD-modell av cellemodulene brukt i akkumulatorssystemet. På toppen av cellemodulen er det tilkoblinger for pluss og minus, samt konnektor for temperaturavlesning. Cellemodulene er fra Enerigus Power Solutions, produktnr. Li1x10p25RT, og består av 10 parallellkoblede Samsung INR18650-25R celler. <https://www.enerigusps.com/shop/product/li1x10p25rt-li-ion-building-block-with-temp-sensor-3-6v-25ah-15c-198?category=3>

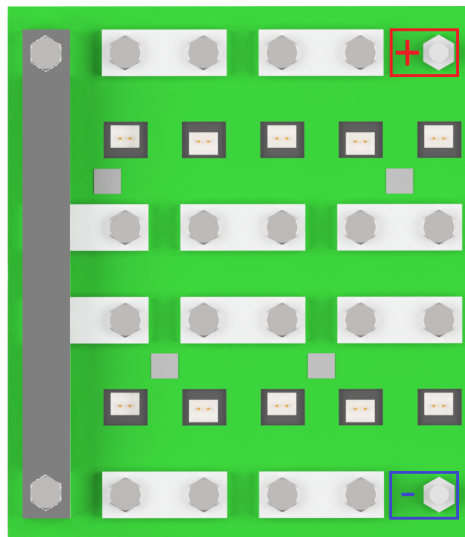




(a) Oppsettet av celleduler i segmentene.

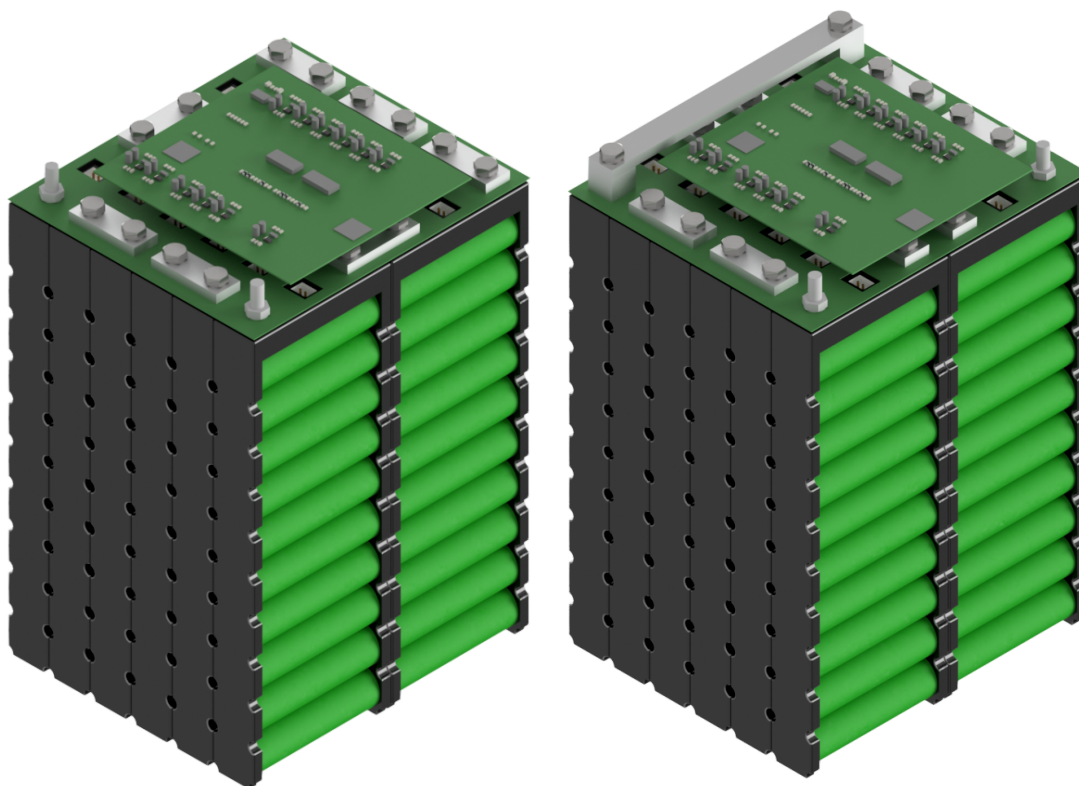


(b) Konfigurasjon brukt på cellesegmenter på sidene av akkumulatorsystemet. Den lengste busbaren er uthevet i figuren for å indikere at den krysser over de andre.



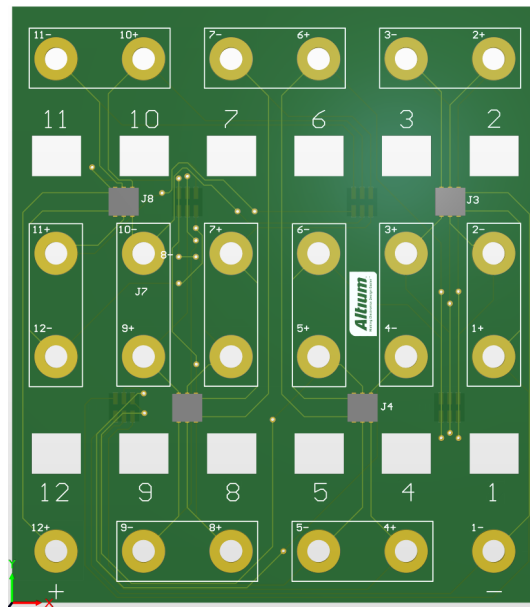
(c) Konfigurasjon brukt på cellesegmentene på sidene av akkumulatorsystemet.

**Figur A.2:** CAD-sammenstillinger av de to forskjellige konfigurasjonene av cellesegmenter, med aluminiumsskinner som kobler celledulerne i serie. I både Figur A.2b og A.2c er celledulerne som i A.2a, og har det mest negative punktet i cellesegmentet nederst til høyre i figuren. Grunnen til at det er lagt opp til to forskjellige design på layout-kortet er at det gir en strømvei som vist i Figur A.7, og da unngår man å måtte krysse internt i akkumulatorsystemet med HV-kabler. Figurer hentet fra Autodesk Inventor.

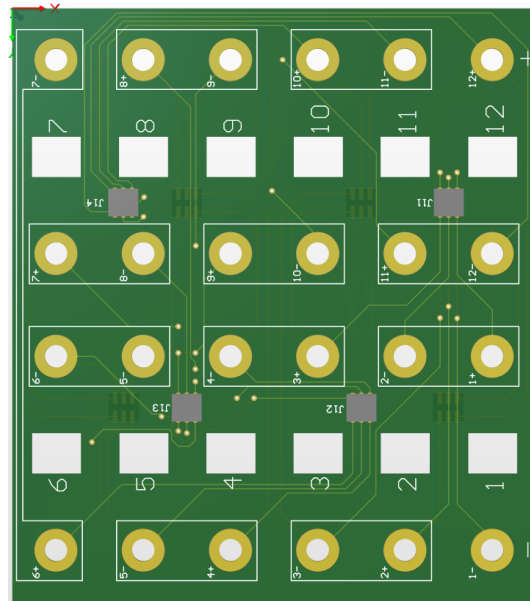


- (a) Konfigurasjon brukt på cellesegmentene i midten av akkumulatorsystemet.      (b) Konfigurasjon brukt på cellesegmenter på sidene av akkumulatorsystemet.

**Figur A.3:** CAD-sammenstillinger av de forskjellige konfigurasjonene av cellesegmenter. På toppen av cellemodulene er det et PCB, som har konnektorer til det øverste kretskortet. Det nederste kretskortet har forskjellig utforming på hver side, og kan brukes for begge cellesegment-konfigurasjonene. Det øverste PCB er ansett som slavekortet, og foretar kommunikasjon med mastermodulen. Alle slavene er utskiftbare med hverandre, og adresseres med DIP-brytere. Se Figur A.6 for plassering av segmentene innad i akkumulatorkassen. Figurer hentet fra Autodesk Inventor.

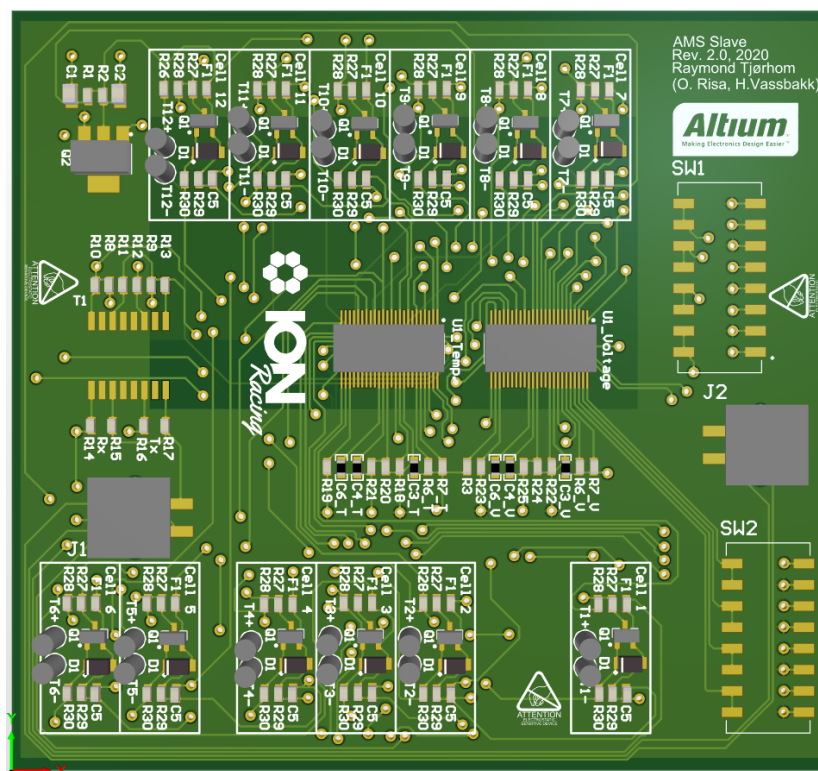


(a) Topplag

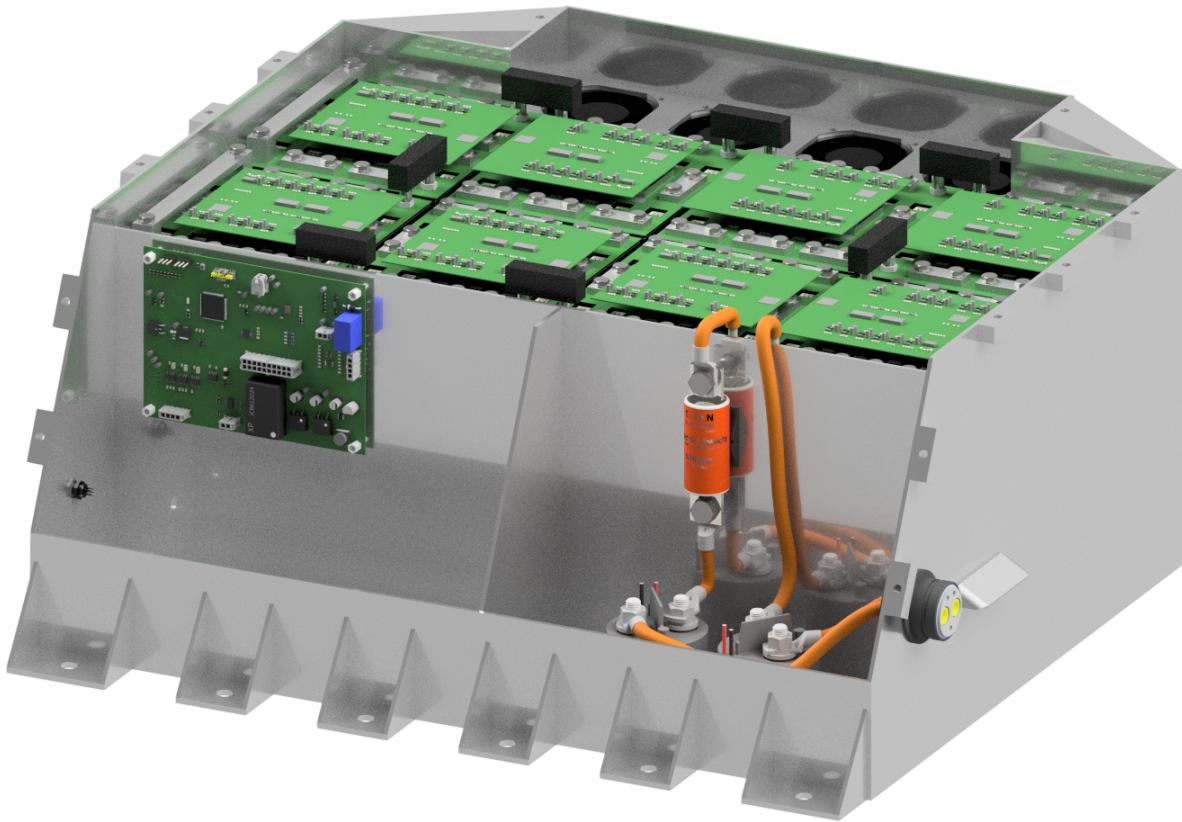


(b) Bunnlag

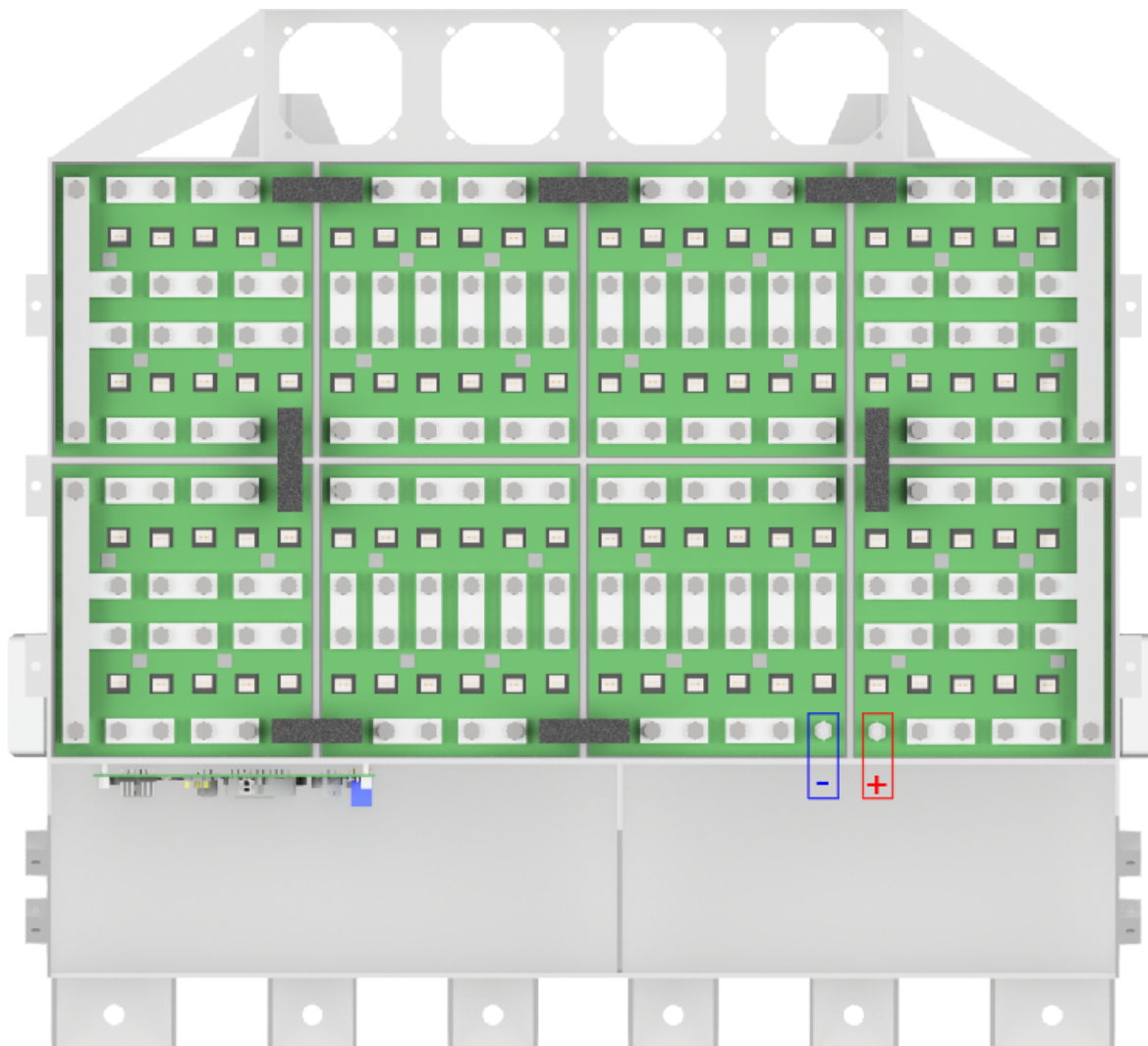
**Figur A.4:** CAD-sammenstillinger av topp- og bunnlag på layout-kortet. Nummerringen fra en til tolv på kortet viser cellenummer i segmentet, hvor en er den mest negative og tolv den mest positive. PCB er koblet opp slik at de samme signalene går i de ekvivalente konnektorene på hver sin side. Figurer hentet fra Altium Designer.



**Figur A.5:** CAD-sammenstillinger av slavekortet. Kortet benytter to IC-er, én for spenningsmåling og én for temperaturmåling. De tolv markerte områdene er måledata for hver celle. Kortet adresseres ved hjelp av to DIP brytere, SW1 og SW2. Disse stilles inn likt, men har motsatt logikk. Segmentene nummereres 0-7, så for segment nr. 2 vil dette bety at spenning-IC har adresse 0010 og temperatur-IC har adresse 1101. Figur hentet fra Altium Designer.



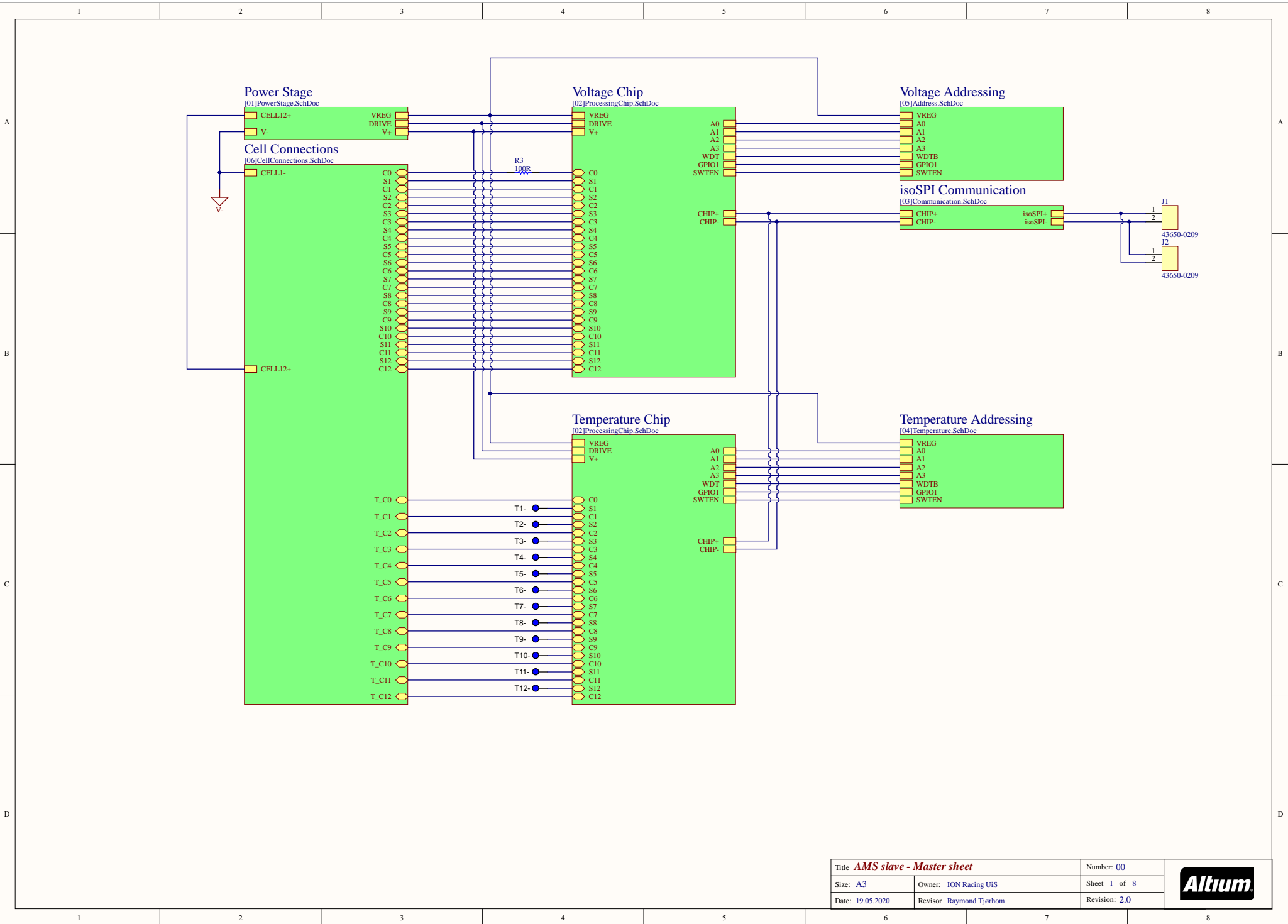
**Figur A.6:** CAD-sammenstilling av akkumulatorsystemet med cellesegmenter, koblinger mellom segmentene og AMS-systemet. AMS masteren er PCB i venstre frontkammer. Høyre frontkammer er ment for HV-komponenter som bl.a. hovedsikring. De svarte pluggene mellom cellesegmentene, som kalles *maintenance plugs*, er de elektriske koblingene som er tilkoblet under drift. Disse pluggene vil bli fastlåst av lokket, og fungerer både som elektrisk ledende, i tillegg til å bidra til å holde segmentene på plass ved en eventuell velt. Figur hentet fra Autodesk Inventor.



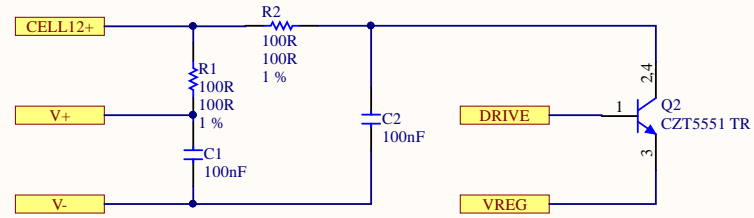
**Figur A.7:** Fugleperspektiv av CAD-sammenstilling av akkumulatorsystemet, uten AMS-slaver, for å vise strømveien i akkumulatorsystemet. Mest negativ og mest positiv tilkobling er markert. Figur hentet fra Autodesk Inventor.

# Tillegg B

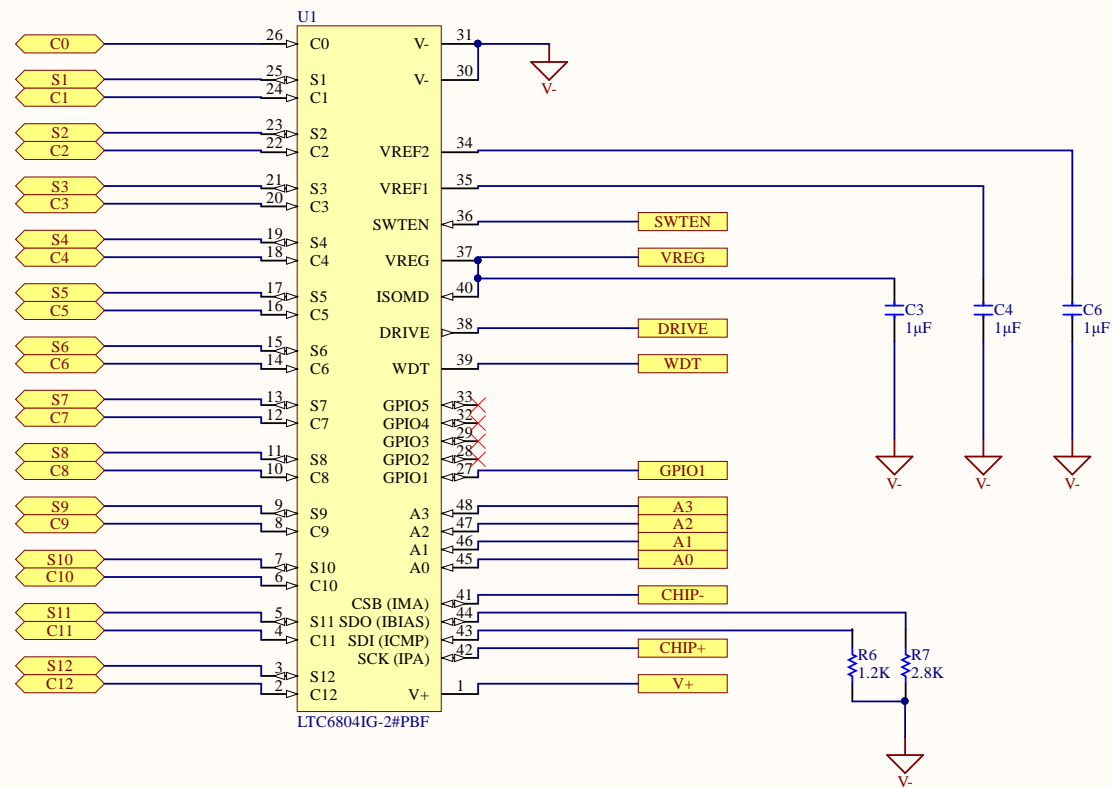
## Skjematikk: AMS-slave







Title <i>AMS slave - Power stage</i>		Number: 01	
Size: A4	Owner: ION Racing UiS	Sheet 2 of 8	
Date: 19.05.2020	Revisor: Raymond Tjørhom	Revision: 2.0	



Title <b>AMS slave - Processing chip</b>		Number: 02
Size: A4	Owner: ION Racing UiS	Sheet 3 of 8
Date: 19.05.2020	Revisor: Raymond Tjørhom	Revision: 2.0



1

2

3

4

A

A

B

B

C

C

D

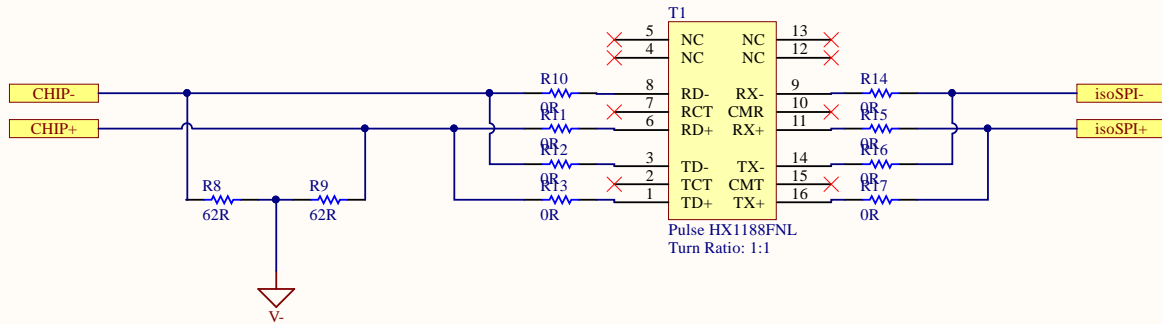
D

1

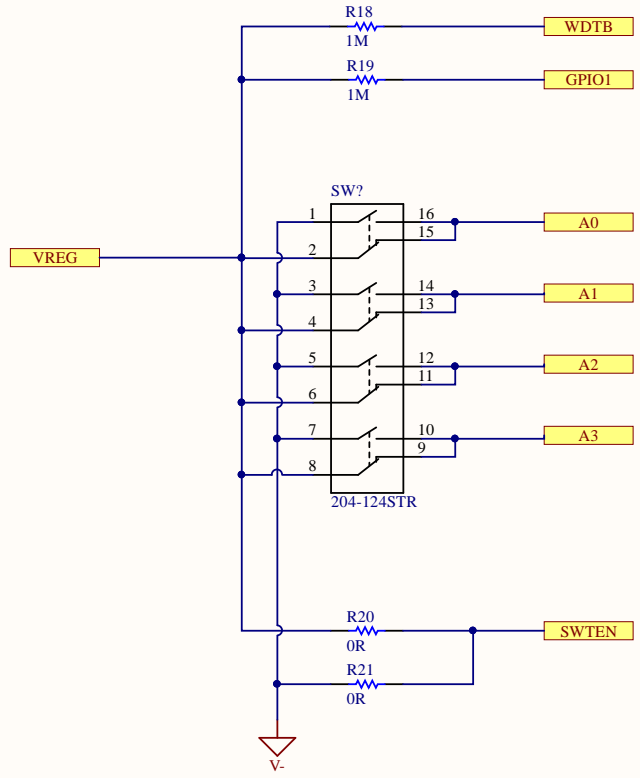
2

3

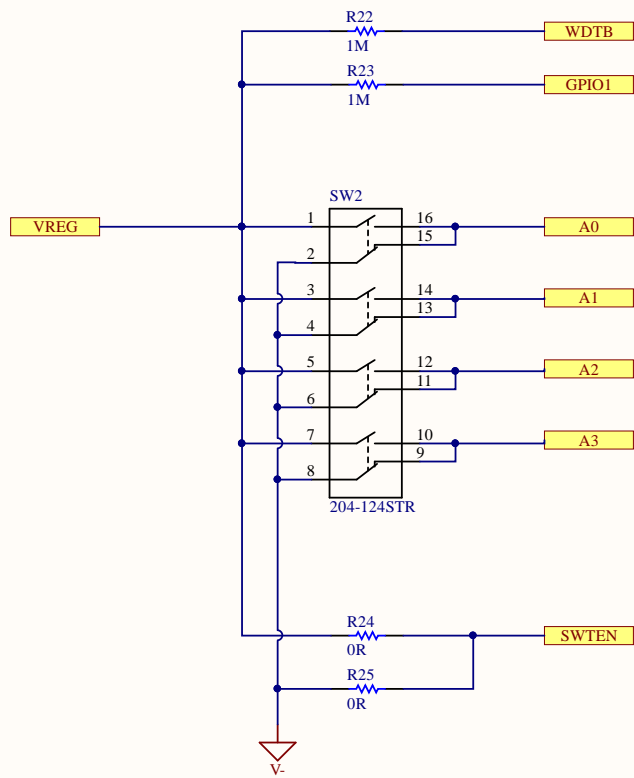
4



Title <b>AMS slave - Communication</b>		Number: 03	
Size: A4	Owner: ION Racing UiS	Sheet 4 of 8	
Date: 19.05.2020	Revisor: Raymond Tjørhom	Revision: 2.0	



Title <b>AMS slave - Temperature addressing</b>		Number: 04	
Size: A4	Owner: ION Racing UiS	Sheet 5 of 8	
Date: 19.05.2020	Revisor: Raymond Tjørhom	Revision: 2.0	



Title **AMS slave - Voltage addressing**

Number: 05

Size: A4

Owner: ION Racing UiS

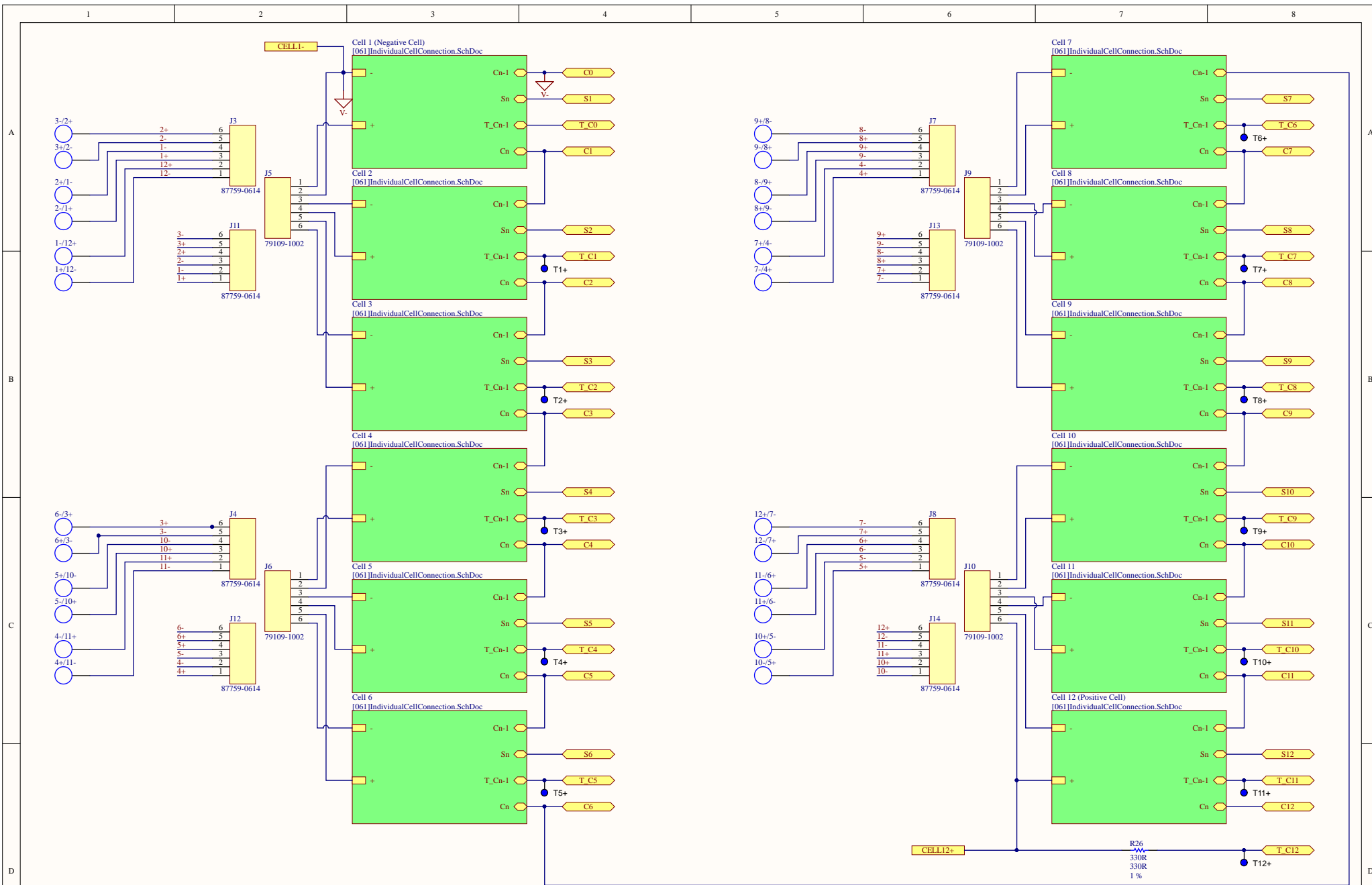
Sheet 6 of 8

Date: 19.05.2020

Revisor Raymond Tjørhom

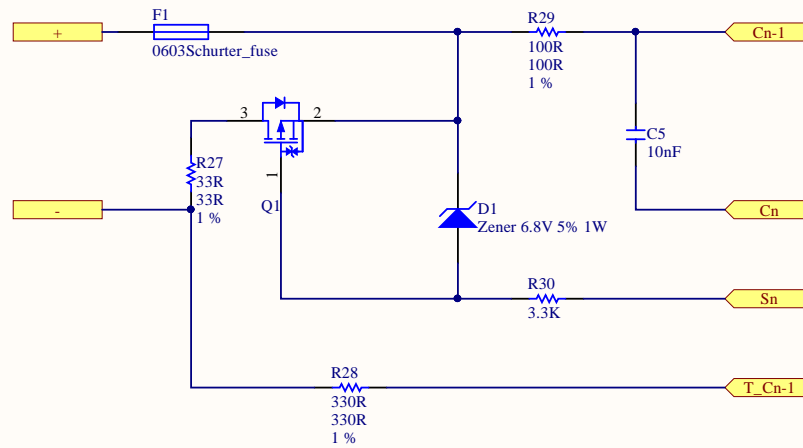
Revision: 2.0



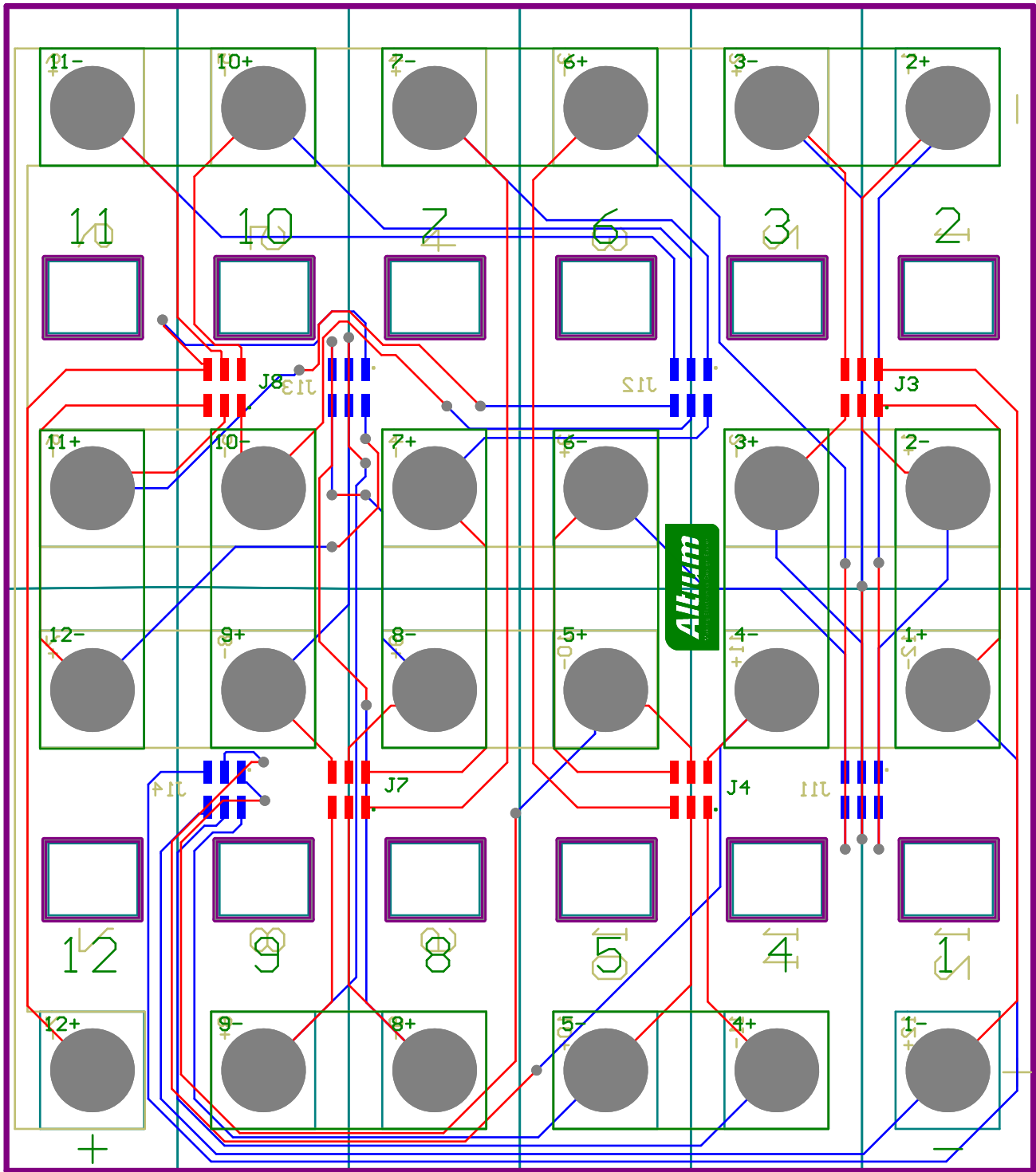


Title <b>AMS slave - Cell connections</b>		Number: 06
Size: A3	Owner: ION Racing UiS	Sheet 7 of 8
Date: 19.05.2020	Revisor: Raymond Tjørhom	Revision: 2.0

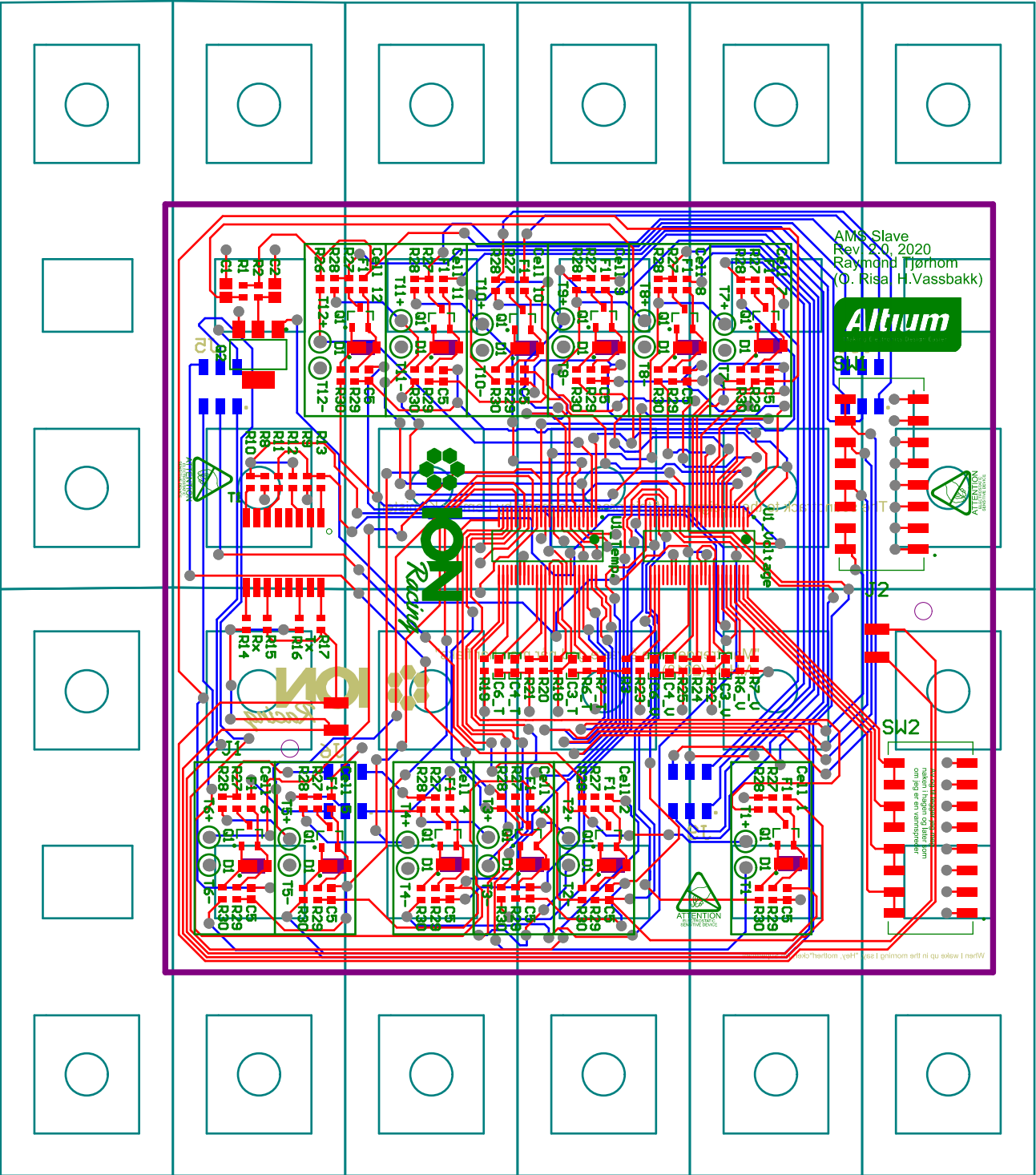




Title <b>AMS slave - Individual cell circuit</b>		Number: 061	
Size: A4	Owner: ION Racing UiS	Sheet 8 of 8	
Date: 19.05.2020	Revisor: Raymond Tjørhom	Revision: 2.0	







AMS Slave  
Rev 2.0, 2020  
Raymond Tjørhom  
(O. Risa, H.Vassbakk)



radiation (høyden og fallet) som  
fungerer på en vanlig måte

radiation (høyden og fallet) som  
fungerer på en vanlig måte

Tillegg C

**MATLAB-script**

```

% "UKF_MasterThesis"
% Raymond Tjorhom
% Spring 2020
%
% This program utilises the algorithm presented in the "Konstruksjon"-
% chapter of my thesis. The algorithm is based on the general algorithm
% from R. van der Merwe in his doctoral dissertation, and implements the
% state-space model presented by Y. Shen. This program has multiple
% multiple choice menus for various options. After having saved the
% initial
% variables in the .mat-file "UKF.mat", the user has the option of doing
% a
% "quick run", which skips the initial choices and only does the
% estimation
% of State of Charge via Kalman filter.

clear; clc; close all;
tic;

QuickRun = menu(...
    'Have you done previous runs of the program, and want a quick run?'
    ,...
    'Yes, do a quick run',...
    'No, do a full run',...
    'Exit program');
TimeMenu = toc;

if QuickRun == 1
    load('UKF_MasterThesis.mat')
elseif QuickRun == 2

    %-----
    % Creating a new estimation set
    %-----

    % Battery cell data
    Vmax = 4.15;
    Vmin = 2.5;
    Vnom = 3.6;
    CellCapacity = 25;

    % Load data
    Time1 = toc;
    LoadPatternMenu = menu('Which load pattern would you like?',...
        'Hard pattern', 'Medium pattern', 'Soft pattern',...
        'Constant 1C', 'Manual input');
    Time2 = toc;
    TimeMenu = TimeMenu + (Time2 - Time1);
    if LoadPatternMenu == 1
        LoadPattern =
        [7,7,7,7, -7, -5, -2,8,10,10,10,10,10,10, -10, -9, -7, -4,...

```

```

1,2,2,2,2,2,2,2,4,5,5,-2,-5,-2,8,10,-2,-3,5,7,7];
elseif LoadPatternMenu == 2
    LoadPattern =
[3,3,3,3,-3,-2,-1,4,5,5,5,5,5,-5,-4,-3,-2,1,1,1,...
1,1,1,1,1,2,2,2,-1,-2,-1,4,5,-1,-1,2,3,3];
elseif LoadPatternMenu == 3
    LoadPattern =
[1,1,1,1,-3,-2,-1,2,2,-2,-1,2,2,2,-5,-4,-3,-2,1,1,...
1,1,1,1,1,1,1,1,1,-1,-2,-1,2,2,-1,-1,1,1,1];
elseif LoadPatternMenu == 4
    LoadPattern = 1;
elseif LoadPatternMenu == 5
    Time1 = toc;
    LoadPattern = input('Input load pattern: ','s');
    Time2 = toc;
    TimeMenu = TimeMenu + (Time2 - Time1);
elseif LoadPatternMenu == 0
    return
end
figure()
plot(LoadPattern(:),'-b')
line([0,length(LoadPattern)],[0,0],'color','black','Linewidth',2)
grid on
ax=gca;
ax.FontSize=20;
xlabel('Run section')
ylabel('Energy usage [C]')
title('\fontsize{24}Load pattern')

RunTime = ceil(36000/sum(LoadPattern)); % Time in 1/10 seconds
I_step = LoadPattern * CellCapacity; % Assume constant step load
current.
I = imresize(I_step, [1,RunTime],'nearest'); % Resize to run time
StepRunTime = floor(RunTime/length(I_step));

% Create assumed cell voltage degredation look-up table and plot it
x = CellCapacity * linspace(0,1,22);
SOC_range = linspace(x(1),x(end),RunTime);
v_1C = [4.15 4.03 3.98 3.945 3.89 3.84 3.79 3.745 3.695 3.65 3.61...
3.575 3.55 3.51 3.48 3.45 3.41 3.35 3.28 3.08 2.75 2.5];
v_10C = [4.15 3.48 3.435 3.385 3.35 3.32 3.29 3.27 3.24 3.215...
3.185 3.16 3.13 3.11 3.08 3.05 3.01 2.95 2.875 2.73 2.5 2.5];
V_ass_1C = interp1(x,v_1C,SOC_range);
V_ass_10C = interp1(x,v_10C,SOC_range);
for k = 1:RunTime
    V_ass_range(:,k) = linspace(V_ass_1C(k),V_ass_10C(k),10);
end
SOC_lut = linspace(0,1,1001);
for k = 1:length(SOC_lut)
    [minValue,closestIndex] = min(abs(SOC_lut(k)-SOC_range/
CellCapacity));
    V_lut(:,k) = V_ass_range(:,closestIndex);

```

```

end

figure()
p1 = plot(x,v_1C,'ok',SOC_range,V_ass_1C,'-b','Linewidth',2);
hold on
for j = 2:9
    p3 = plot(x,v_10C,'ok',SOC_range,V_ass_range(j,:), 'Linewidth',1);
end
p2 = plot(x,v_10C,'ok',SOC_range,V_ass_10C,'-r','Linewidth',2);
hold off
grid on
ax=gca;
ax.FontSize=20;
xlabel('Used Capacity [Ah]')
ylabel('Cell voltage [V]')
title('\fontsize{24}LUT of assumed cell voltage degradation')
legend([p1(1) p1(2) p2(2)],...
    {'Interpolation points','$v_{cell,est,1C}(k)$',...
    '$v_{cell,est,10C}(k)$'},'Interpreter','latex')

% Claculate assumed cell voltage degradation and plot it
for l = 1:length(I_step)
    for k = 1:StepRunTime
        V_ass(StepRunTime*(l-1)+k) = ...
            V_ass_range(abs(I(StepRunTime*(l-1)+k)/CellCapacity),...
            StepRunTime*(l-1)+k);
    end
end
for k = 1:RunTime
    V_ass(k) = V_ass_range(abs(I(k)/CellCapacity),k);
end

if length(LoadPattern)>1
    figure()
    plot(V_ass(:),'-b')
    grid on
    ax=gca;
    ax.FontSize=20;
    xlabel('Run section')
    ylabel('Voltage [V]')
    title(['\fontsize{24}Assumed cell voltage from LUT',...
        ' based on abs(load)'])
    V_ass = [];
    for l = 1:(length(I_step)+2)
        if StepRunTime*(l-1)==0
            V_ass(l) = V_ass_range(abs(I_step(1))/CellCapacity,1);
        elseif l>=(length(I_step)+1)
            V_ass(l) = Vmin;
        else
            V_ass(l) = V_ass_range(abs(I_step(1))/CellCapacity,...
                StepRunTime*(l));
        end
    end
end

```

```

end

figure()
plot(V_ass(:), '-b')
grid on
ax=gca;
ax.FontSize=20;
xlabel('Time increment, k')
ylabel('Cell voltage [V]')
title(['\fontsize{24}Assumed cell voltage ',...
       'interpolation of cell voltages based on abs(load)'])

V_range = linspace(V_ass(1),V_ass(end),RunTime);
SamplePoints = round(linspace(1,RunTime,1));
V_ass = interp1(SamplePoints,V_ass,1:RunTime);
figure()
plot(V_ass(:), '-b')
grid on
ax=gca;
ax.FontSize=20;
xlabel('Run section')
ylabel('Voltage [V]')
title(['\fontsize{24}Assumed cell voltage based on ',...
       'interpolation points'])

end

% Choose number of cells for estimation set
Time1 = toc;
NoOfCells = menu('How many datasets would you like?',...
                '1','2','3','12','96','Manual input');
Time2 = toc;
TimeMenu = TimeMenu + (Time2 - Time1);

if NoOfCells == 4
    NoOfCells = 12;
elseif NoOfCells == 5
    NoOfCells = 96;
elseif NoOfCells == 6
    Time1 = toc;
    NoOfCells = input('How many datasets would you like? ');
    Time2 = toc;
    TimeMenu = TimeMenu + (Time2 - Time1);
elseif NoOfCells == 0
    return
end

% Apply distortion to voltage sets
for i = 1:NoOfCells
    V(i,:) = awgn(V_ass,40); % Chose SNR to 40
end

% Plot voltage readings with noise

```

```

figure()
hold on
for i = 1:NoOfCells
    plot(V(i,:))
end
hold off
grid on
ax=gca;
ax.FontSize=20;
axis([0 RunTime 2 4.5])
xlabel('Time increment, k')
ylabel('Cell voltage [V]')
title('\fontsize{24}Assumed cell voltage readings with measurement
noise')

% Gives the option of saving new data, if the user wants to
Time1 = toc;
SaveNewData = menu('Do you want to keep the new data?',...
    'Yes, and to save in a .mat-file',...
    'Yes, but NOT to save in .mat-file');
Time2 = toc;
TimeMenu = TimeMenu + (Time2 - Time1);

if SaveNewData==1
    Time1 = toc;
    FileName = 'UKF_MasterThesis.mat';
    NewFileName = menu('Do you want to overwrite "UKF_MasterThesis.
mat"?',...
        'Yes',...
        'No, I want to choose a new name');
    Time2 = toc;
    TimeMenu = TimeMenu + (Time2 - Time1);

    if NewFileName == 2
        Time1 = toc;
        FileName=input('File name: ','s');
        Time2 = toc;
        TimeMenu = TimeMenu + (Time2 - Time1);
    elseif NewFileName == 0
        return
    end
    save(FileName, '-regexp',...
        '^(!k|NewData|NewFileName|run|SaveNewData|Time1|Time2|TimeMenu|
ax)$)')
    elseif SaveNewData == 0
        return
    end
elseif QuickRun == 3 || QuickRun == 0
    clear;
    return
end

```

```

%-----
% Kalman Filter
%-----

% Set certain values for the model
T = 290e-6; % Measurement intervall [s]
R_series = 2.7e-6;
R_short = 2e-3;
C_short = 500;
R_long = 1.1e-3;
C_long = 5000;
C_capacity = sum(I_step/length(I_step))*(RunTime/10)/Vmax;

% Assume initial values from cell model
x = [1; 0; 0]; % [SOC, V'_short, V'_long]
u = I;
Phi = [1, 1-T/(R_short*C_short), 1-T/(R_long*C_long)];
Gamma = [-T/C_capacity; T/C_short; T/C_long];
y = V;
D = [0, -1, -1];
E = R_series;

% Assume initial covariance matrices
Px = [1.7, 0.04, 0.04]; % Covariance matrix for the estimation deviations
Q = 1e-3; % Autocovariance matrix for the input
R = 0.04; % Autocovariance matrix for the output

% Creation of augmented matrices
Px_aug = [Px, Q, R]; % Block diagonal of [Px,Q,R]
x_aug = [x; 0; 0]; % This vector is the mean of [x,v,w]
Phi_aug = [Phi 0 0];
Gamma_aug = [Gamma; 0; 0];

% Expand matrices to accomodate number of cells
x_aug = repmat(x_aug, NoOfCells, 1);
Phi_aug = repmat(diag(Phi_aug), NoOfCells, 1);
Gamma_aug = repmat(Gamma_aug, NoOfCells, 1);
Px_aug = repmat(diag(Px_aug), NoOfCells, 1);

% Initial conditions of the Scaled UKF
alpha = 0.1; % 0 < alpha < 1
beta = 2; % beta >= 0
kappa = 0; % Tuning variable
L = length(x) + 2; % Scaling constant
lambda = alpha^2*(L+kappa)-L; % Scaling constant
gamma = sqrt(L+lambda); % Scaling constant
w0_mean = lambda/(L+lambda); % Weight of mean SP 0
w0_cov = w0_mean+(1-alpha^2+beta); % Weight of covariance SP 0
w = 1/(2*(L+lambda)); % Weight of remaining points
w_mean = [w0_mean repmat(w,1,2*L)]; % Weighting vector of mean SPs
w_cov = [w0_cov repmat(w,1,2*L)]; % Weighting vector of covariance SPs

```



```

PreKal = toc;
k = 0;
run = 1;
y_est_store(:,1) = y(:,1);

% Scaled Unscented Kalman Filter
while(run && k < RunTime)
k = k + 1;
for i = 1:NoOfCells

%-----
% Sigma points
%-----

% Cholesky decomposition of augmented x-covariance matrix
Px_chol = chol(Px_aug(i*5-4:i*5,:), 'lower');

% Creation of Sigma Points
SP = x_aug(i*5-4:i*5, ones([1 2*L+1]))...
    + gamma*[zeros(L,1), Px_chol, -Px_chol];

% Defining Sigma points for input, process- and measurement noise
X_SP = SP(1:3,:);
V_SP = SP(end-1,:);
W_SP = SP(end,:);

%-----
% Time-update
%-----

% Find x-points as a function of Sigma points
X = Phi_aug(i*5-4:i*5-2,1:3)*X_SP...
    + Gamma_aug(i*5-4:i*5-2,:)*u(:,k) + V_SP;

% Find estimate for input vector from weighted Sigma points
x_est = X * w_mean';

% Find estimate for x-covariance matrix from weighted Sigma points
Px_est = 0;
for N = 1:(2*L+1)
    Px_est = Px_est + w_cov(N) * ((X(:,N)-x_est)*(X(:,N)-x_est)') + Q;
end

%-----
% Measurement-update
%-----

% Find Y-points as a function of Sigma points and Voc(SOC)
calc = abs(y(i,k)-y_est_store(i,k));
[minValue, closestIndex] = min(min(calc));
[minRow, minCol] = ind2sub(size(V_lut), closestIndex);

```

```

Voc_SOC(i,k) = V_lut(minRow,minCol);
Y = Voc_SOC(i,k) - D * X_SP - E * u(k) + W_SP;

% Find estimate for measurement vector from weighted Sigma points
y_est = Y * w_mean';

% Find estimate for y-covariance matrix from weighted Sigma points
Py = 0;
for N = 1:(2*L+1)
    Py = Py + w_cov(N) * ((Y(:,N)-y_est)*(Y(:,N)-y_est)') + R;
end

% Find estimate for xy-covariance matrix from weighted Sigma points
Pxy = 0;
for N = 1:(2*L+1)
    Pxy = Pxy + w_cov(N) * ((X(:,N)-x_est)*(Y(:,N)-y_est)');
end

% Calculate Kalman gain from covariance matrices
K = Pxy * (Py)^(-1);

% Update augmented x-vector with new estimates of input values
x_aug(i*5-4:i*5-2,:) = x_est + K*(y(i,k) - y_est);

% Update augmented x-covariance matrix from Kalman gain
Px_aug(i*5-4:i*5-2,1:3) = Px_est - (K * Py * K');

% Store values of significance
y_est_store(i,k+1) = y_est;
SOC_store(i,k) = x_aug(i*5-4,:);
SOC_permille(i,k) = round(SOC_store(i,k)*1000);
if SOC_store(i,k)<0
    run = 0;
    break
end
end
end
end
PostKal = toc;
TimeKalman = PostKal - PreKal

figure()
hold on
for i = 1:NoOfCells
    plot(SOC_store(i,:), 'Linewidth', 0.2)
end
hold off
grid on
ax=gca;
ax.FontSize=20;
xlabel('Time increment, k')
ylabel('SOC')

```

```

title('\fontsize{24}Estimation of SOC')
%%
% SOC resolution for the driver
averaging = 100; % Number of samples included in the moving average of
SOC
for j = 1:k
    for i = 1:NoOfCells
        if j >= averaging
            SOC_percent(i,j) = floor(sum(SOC_permille(i,...
                (j-averaging+1):j)/10)/averaging);
        else
            SOC_percent(i,j) = floor(sum(SOC_permille(i,1:j)/10)/j);
        end
    end
end
end
figure()
hold on
for i = 1:NoOfCells
    plot(SOC_percent(i,:), 'Linewidth', 0.2)
end
hold off
grid on
ax=gca;
ax.FontSize=20;
xlabel('Time increment, k')
ylabel('Remaining charge [%]')
title('\fontsize{24}SOC in all cells')

SOC_percent = min(SOC_percent);
figure()
hold on
plot(SOC_percent(:), '-b', 'Linewidth', 0.5)
hold off
grid on
ax=gca;
ax.FontSize=20;
xlabel('Time increment, k')
ylabel('Remaining charge [%]')
title('\fontsize{24}SOC Presented to driver')

TimeEnd = toc;
TimeTotal = TimeEnd - TimeMenu

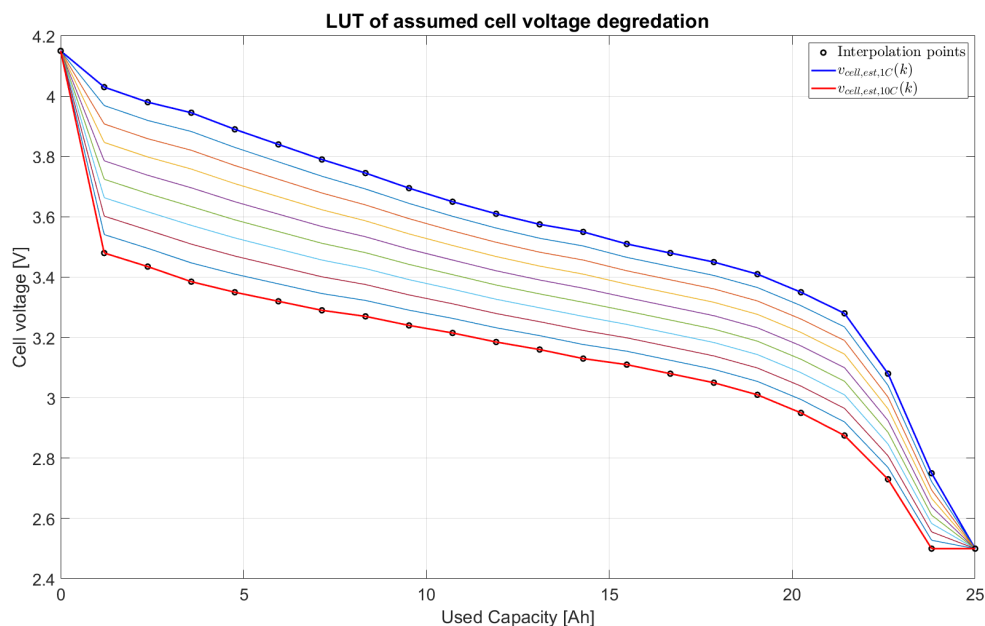
```

# Tillegg D

## Simuleringer i MATLAB

### Oppslagstabell for degradering av cellespenninger

I databladet til cellemodulene er det tegnet opp to utladekurver, én for 1C og én for 10C [3]. Avlesninger av disse kurvene ligger til grunn for oppslagstabellen for degradering av cellespenninger. Degraderingskurvene for mellomliggende laster er så interpolert.

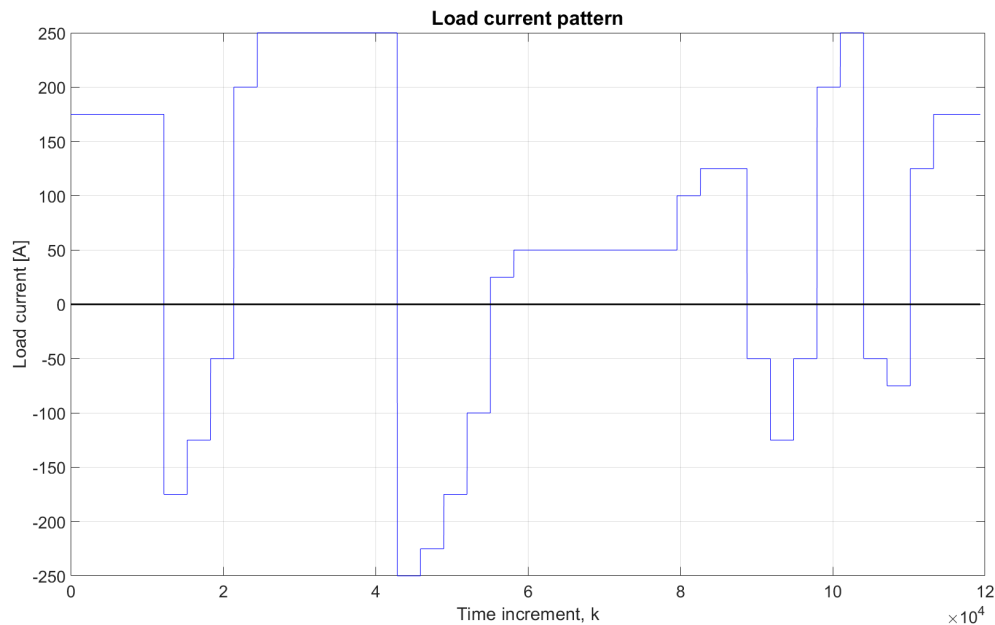


**Figur D.1:** Degraderingskurver for cellespenning under forskjellig last. De opprinnelige avlesningspunktene er markert i figuren.

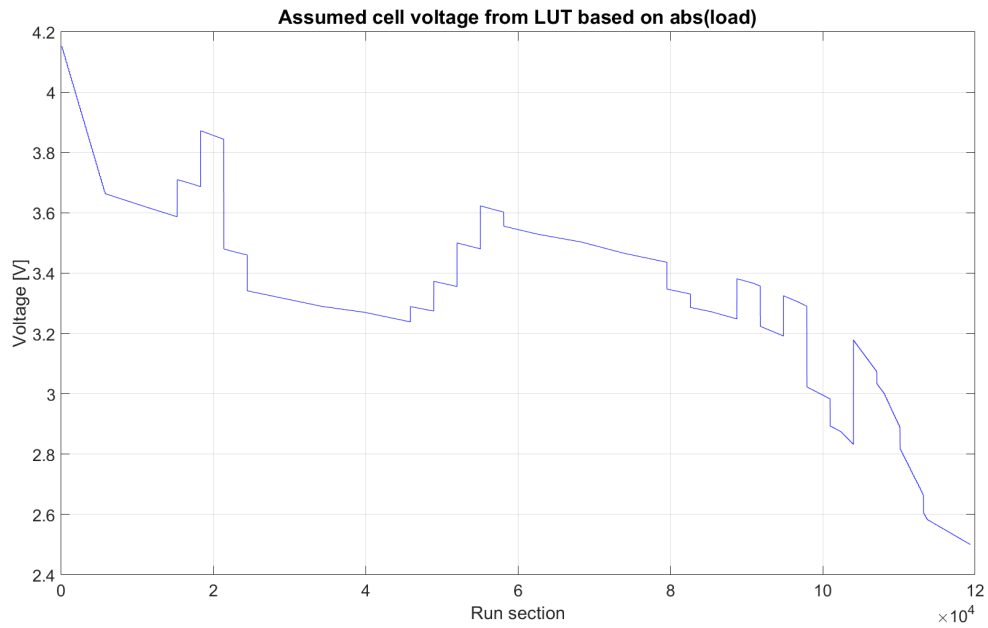
## Simuleringsresultater

Samtlige simuleringer utenom de presentert i Figur D.6 er kjørt med tre cellemoduler. Figur D.6 er kjørt med 96 celler.

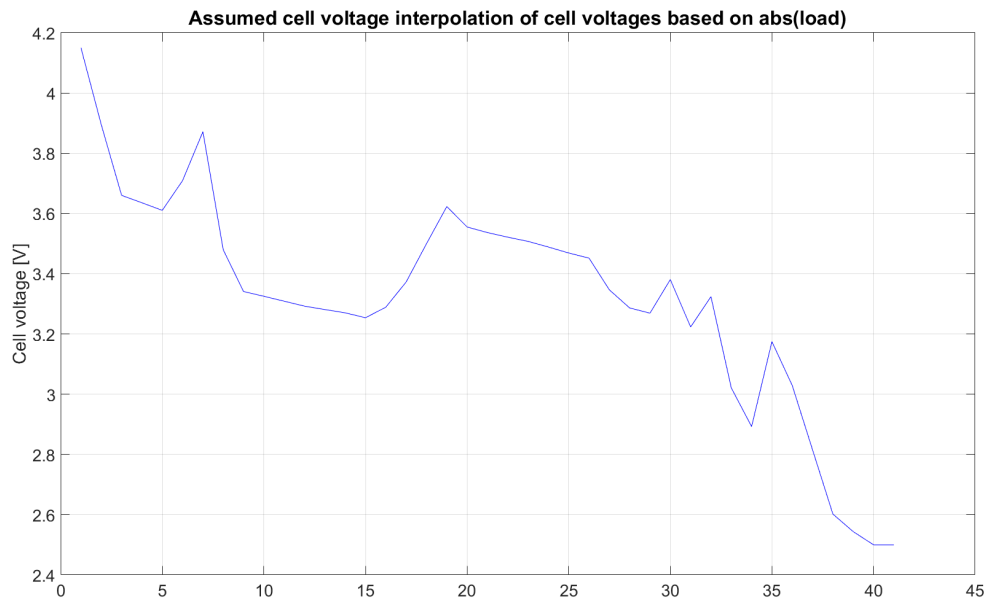
### Hardt lastmønster



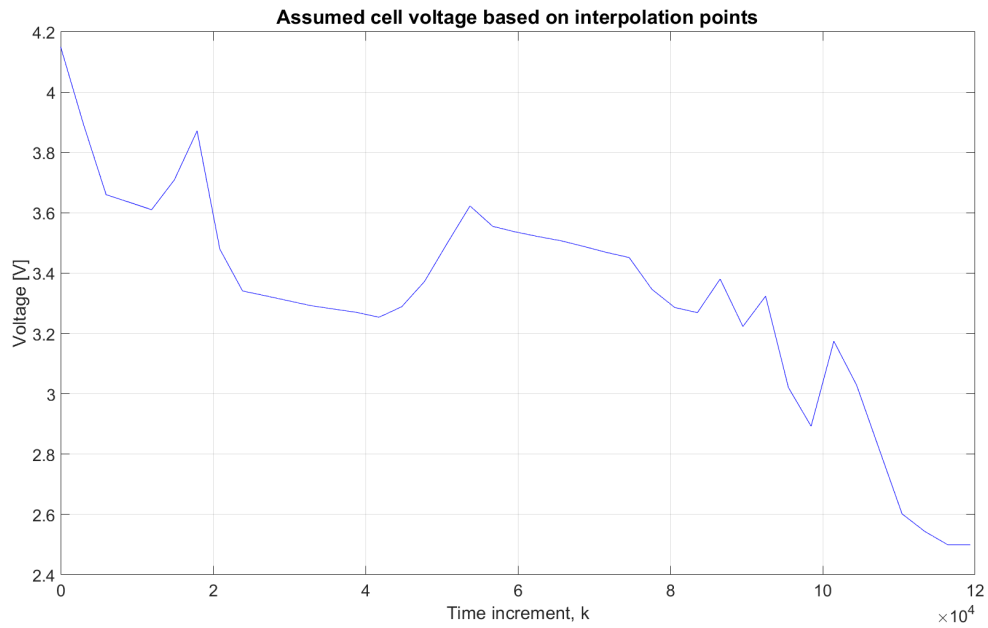
(a) Lastkurve. Kurven er laget etter en tenkt kjøresituasjon.



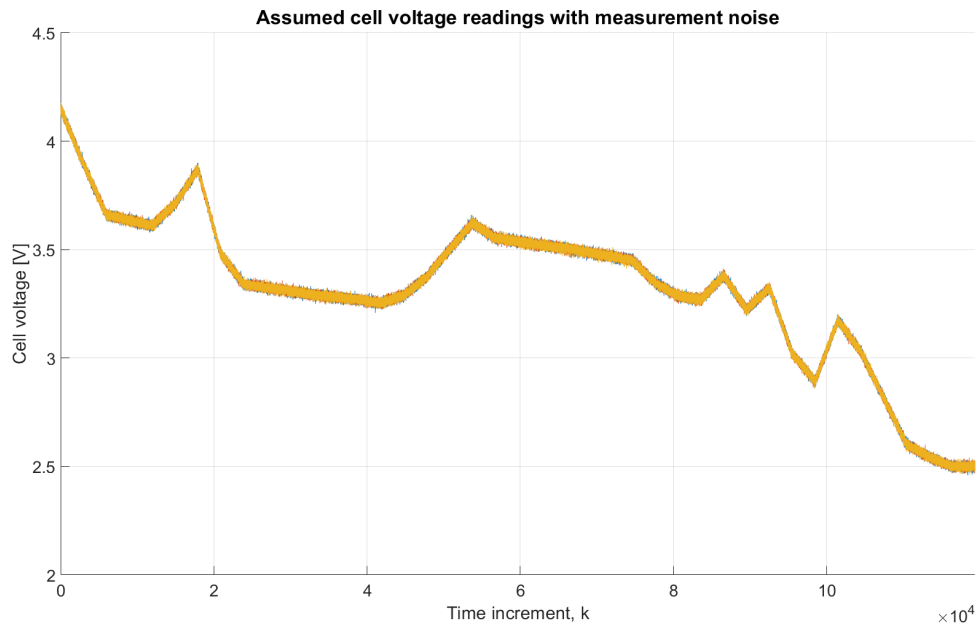
(b) Antatt spenningsmåling fra LUT.



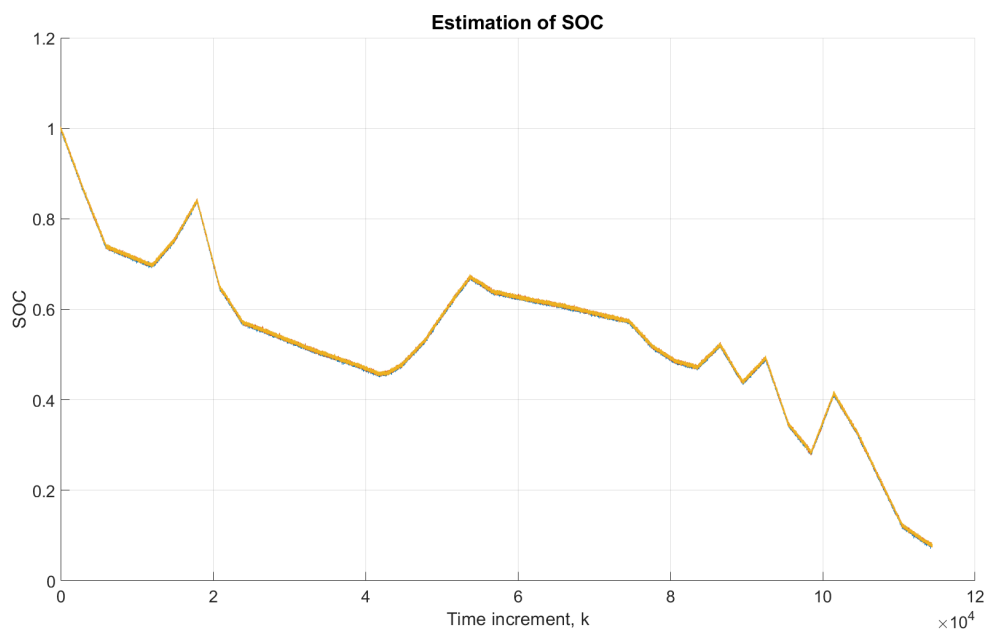
(c) Interpolert spenningsendring fra Figur D.2b.



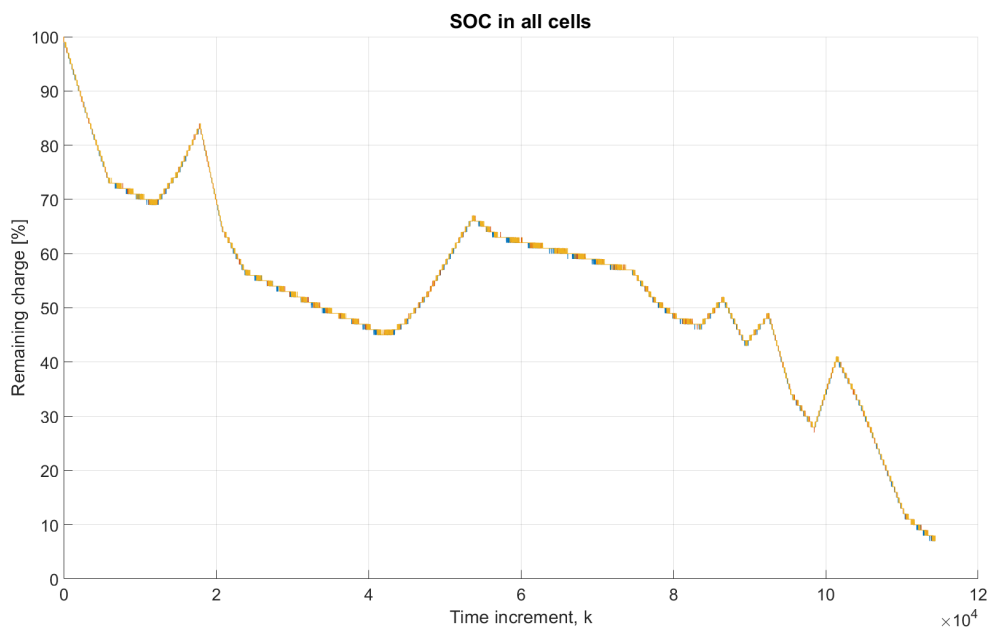
(d) Utvidet spenningskurve fra Figur D.2c som strekker seg over kjøretid.



(e) Spenningskurve fra Figur D.2d med hvit støy.

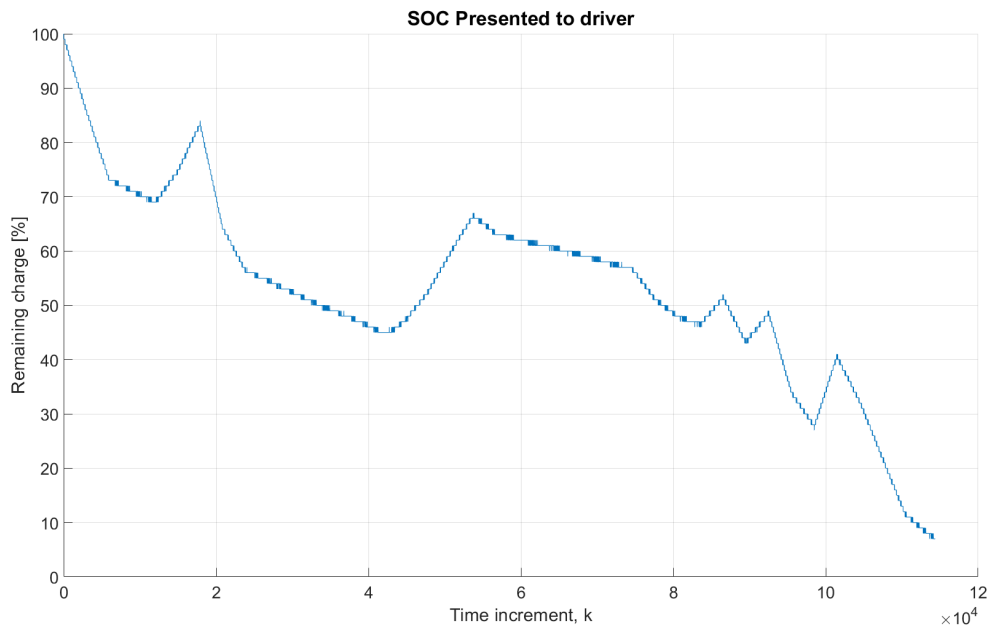


(f) Estimering av SOC fra Kalmanfilter

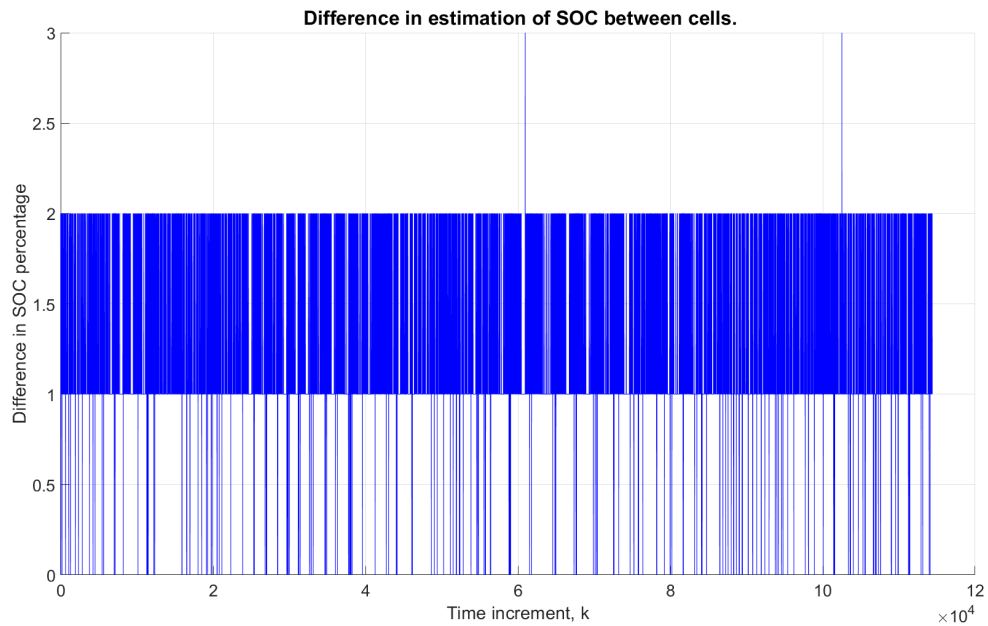


(g) SOC i hele prosent for alle celler i testen.





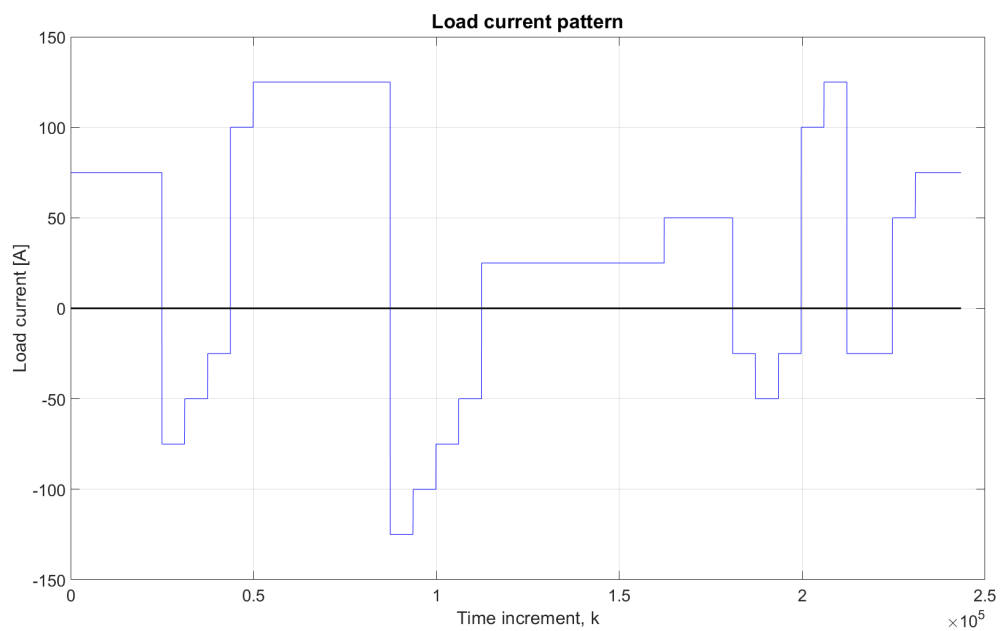
(h) Celle med lavest SOC. Dette er hva føreren får angitt som SOC.



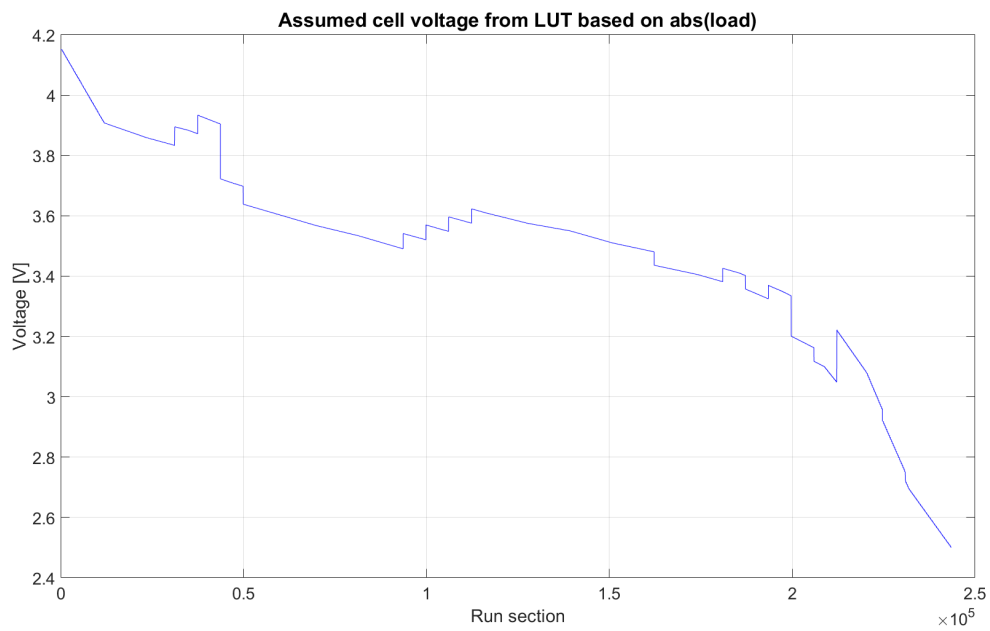
(i) Maksimum spredning i SOC-estimat.

**Figur D.2:** Kurver fra simulering med hardt lastmønster.

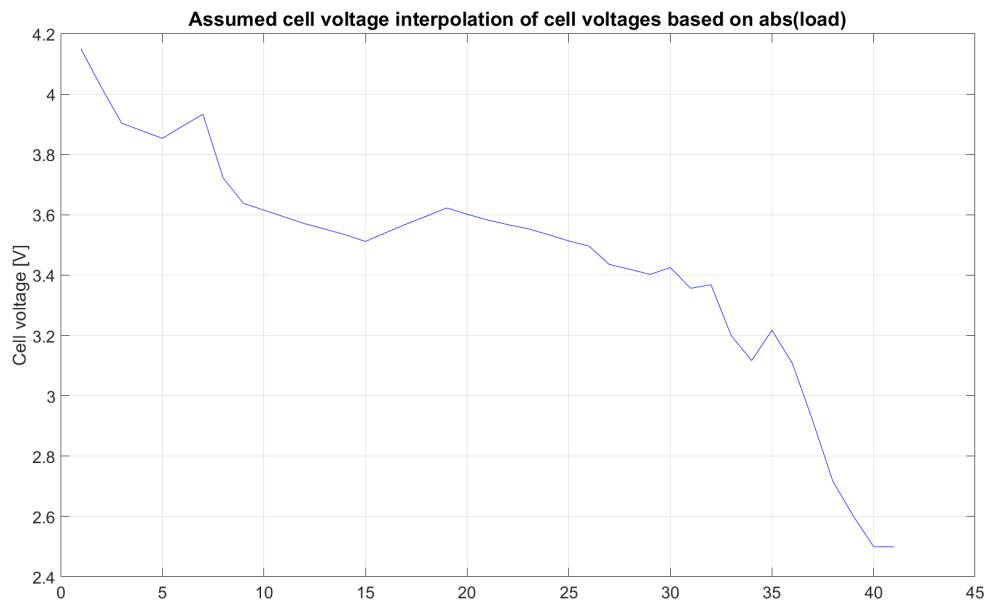
### Middels hardt lastmønster



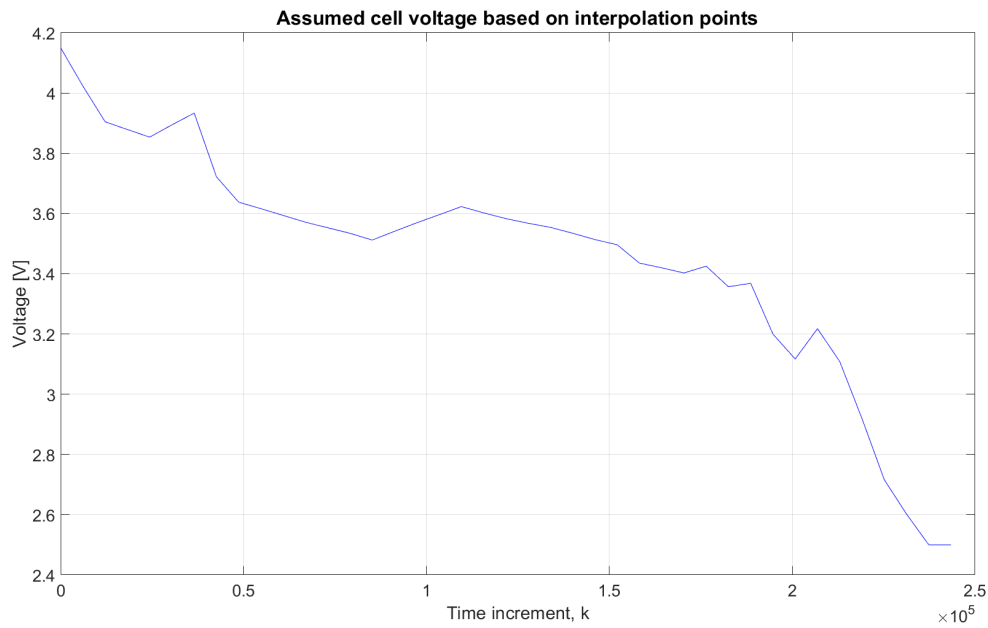
(a) Lastkurve.



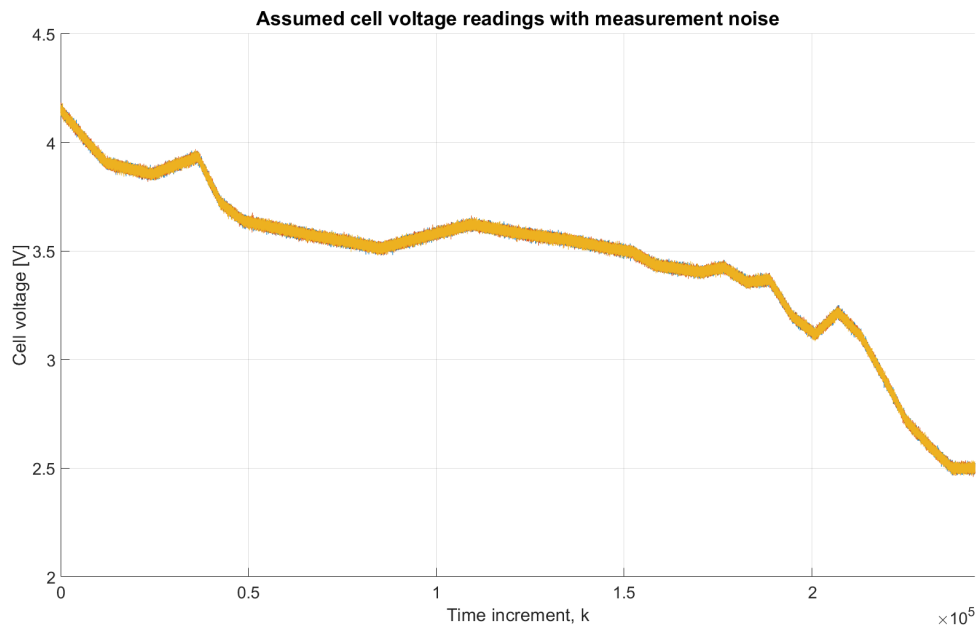
(b) Antatt spenningsmåling fra LUT.



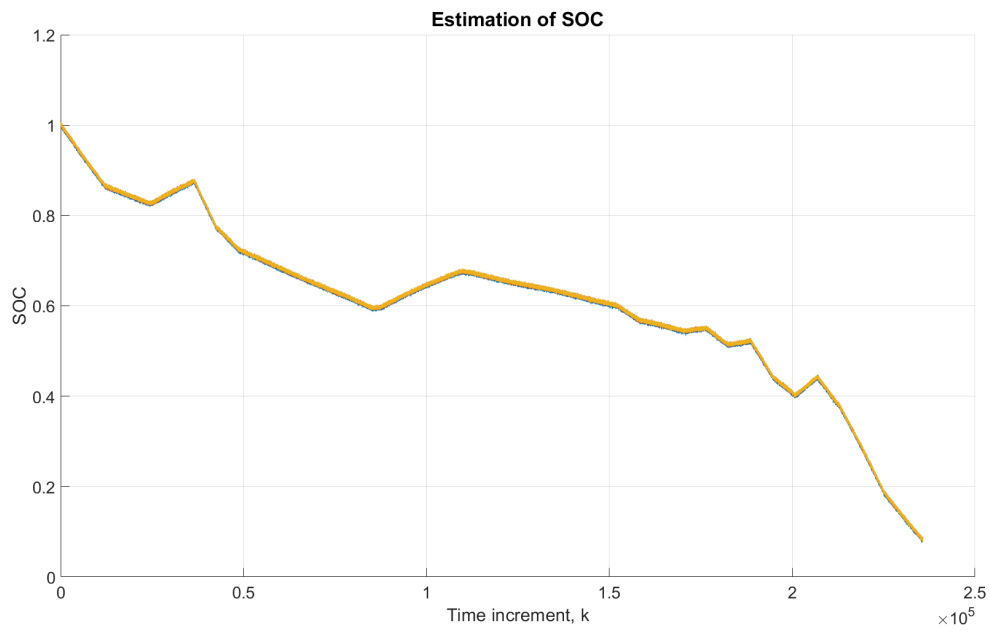
(c) Interpolert spenningsendring fra Figur D.3b.



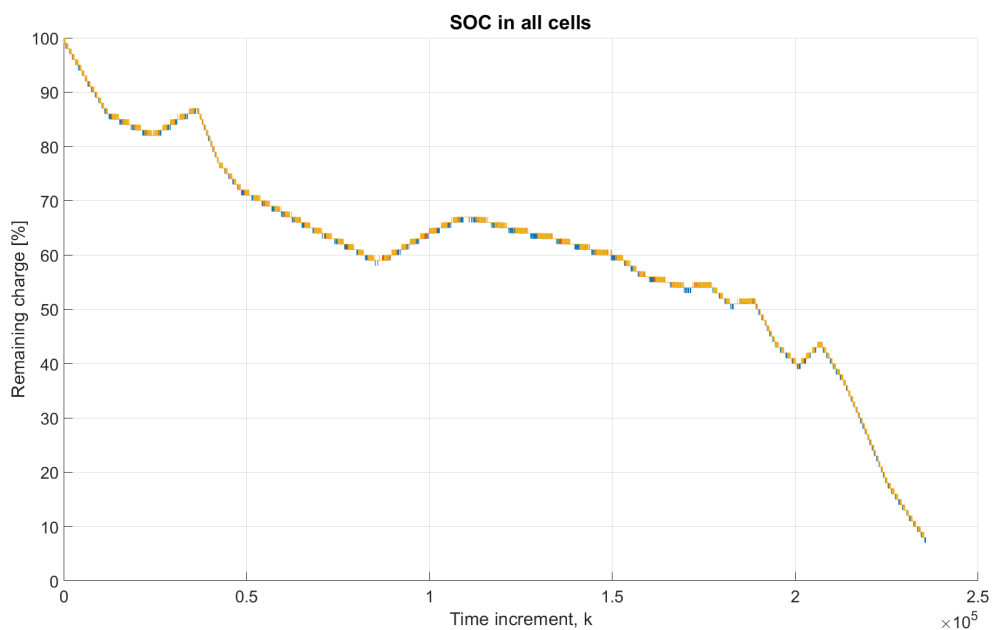
(d) Utvidet spenningskurve fra Figur D.3c som strekker seg over kjøretid.



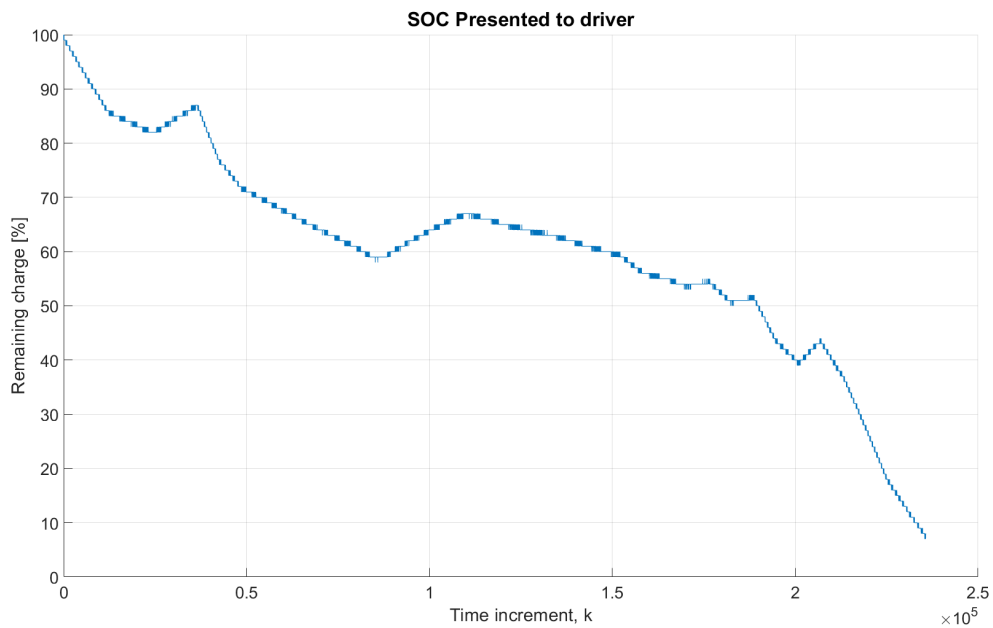
(e) Spenningskurve fra Figur D.3d med hvit støy.



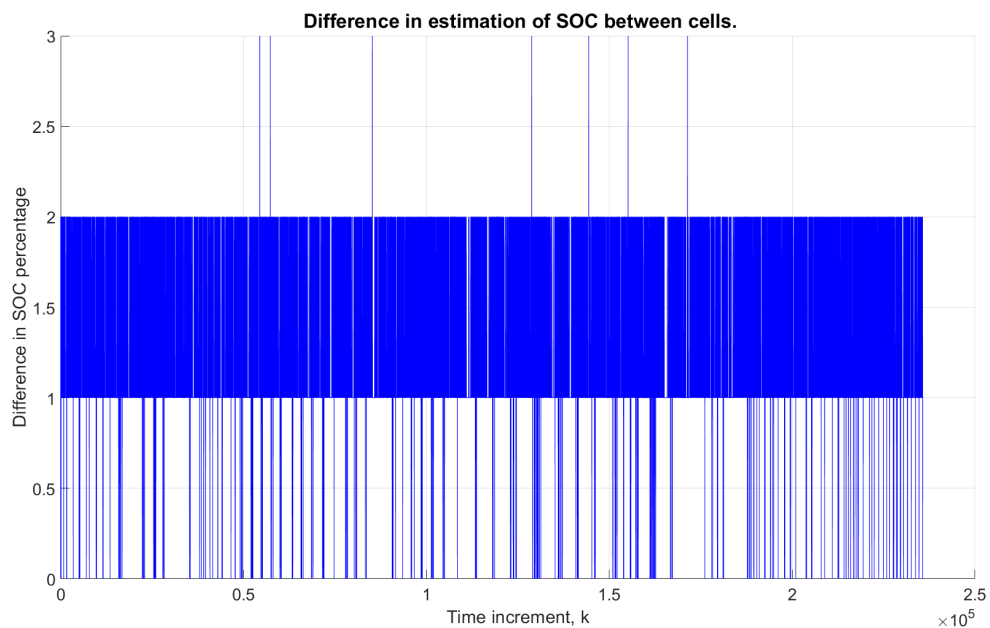
(f) Estimering av SOC fra Kalmanfilter



(g) SOC i hele prosent for alle celler i testen.



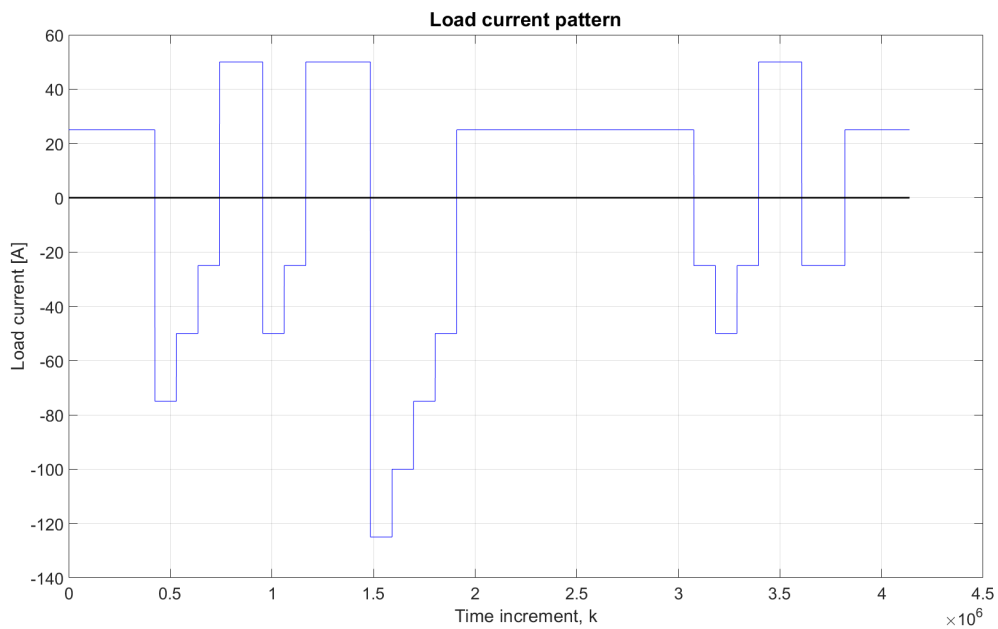
(h) Celle med lavest SOC. Dette er hva føreren får angitt som SOC.



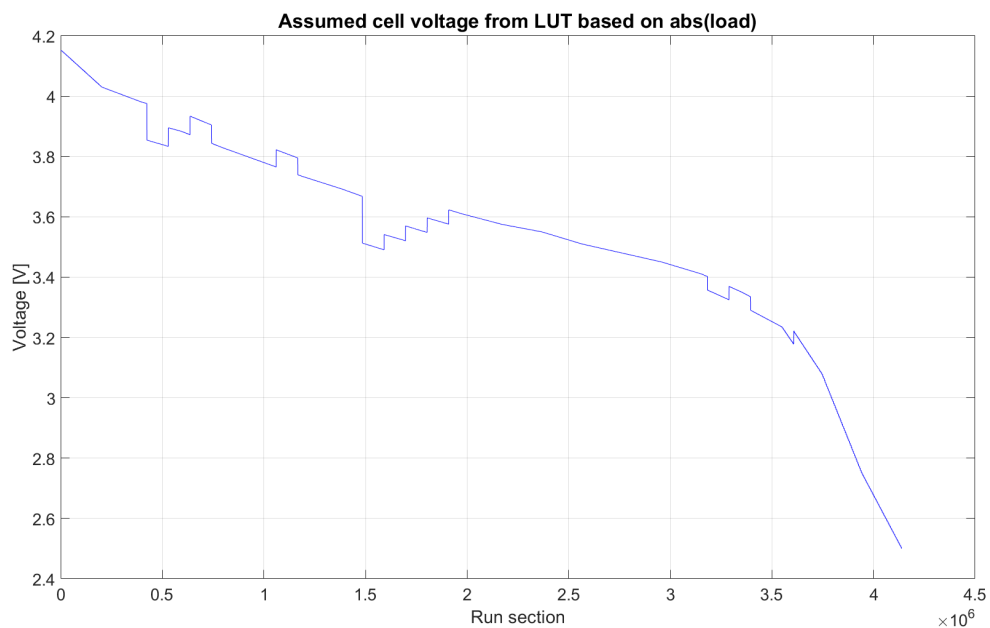
(i) Maksimum spredning i SOC-estimat.

**Figur D.3:** Kurver fra simulering med middels hardt lastmønster.

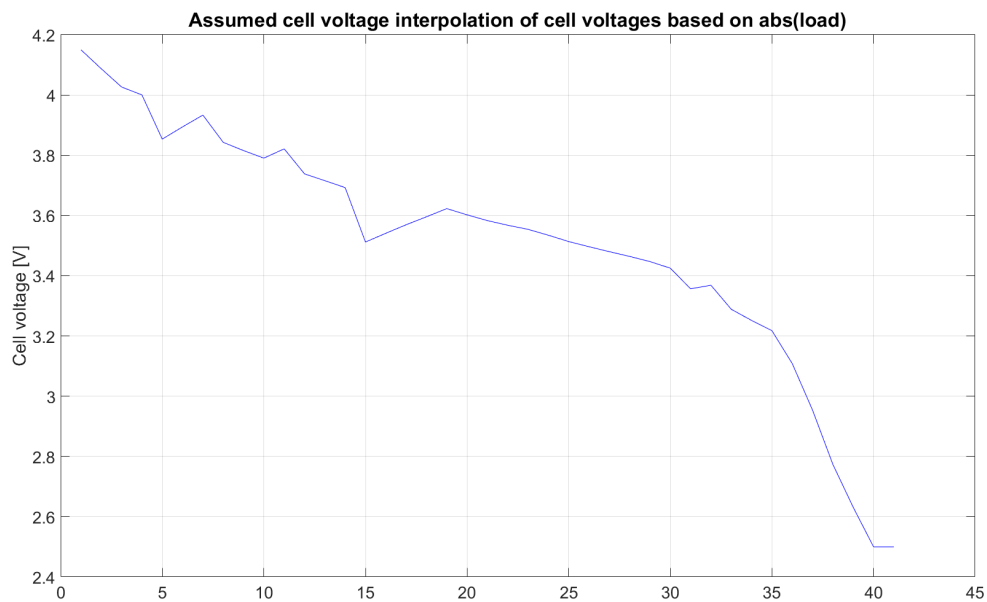
### Rolig lastmønster



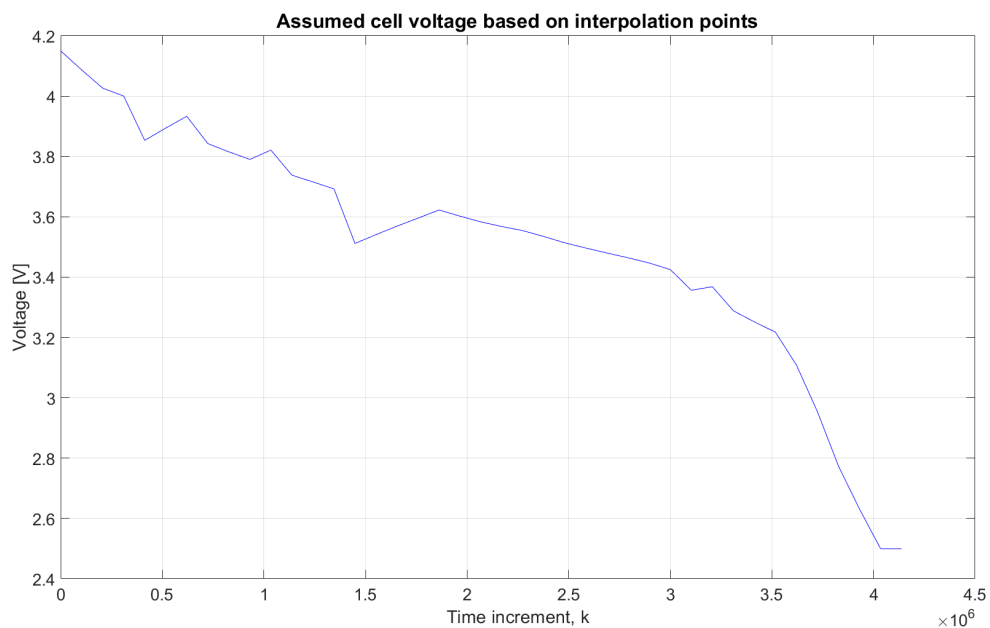
(a) Lastkurve.



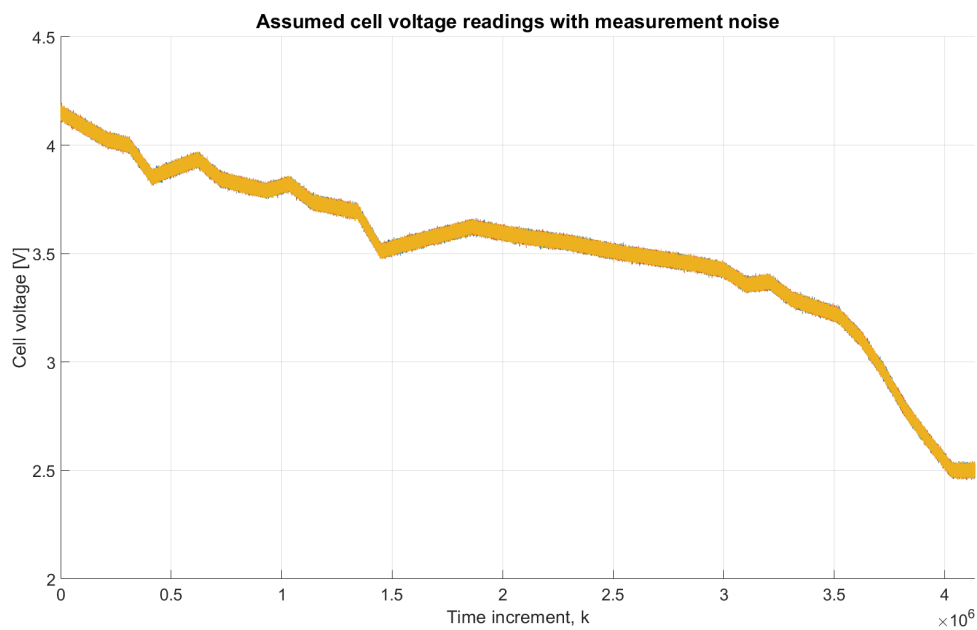
(b) Antatt spenningsmåling fra LUT.



(c) Interpolert spenningsendring fra Figur D.4b.

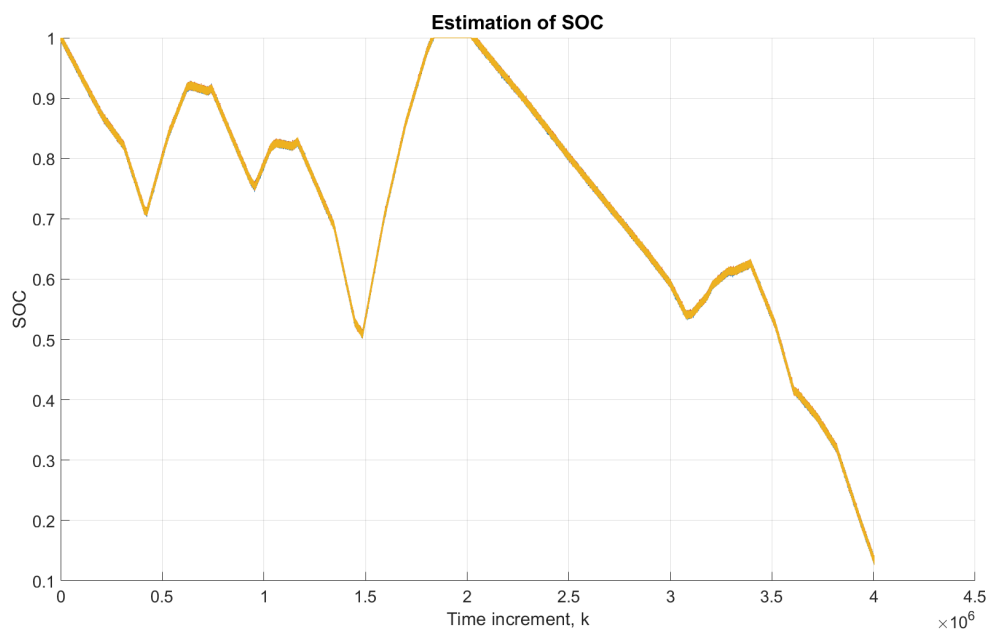


(d) Utvidet spenningskurve fra Figur D.4c som strekker seg over kjøretid.

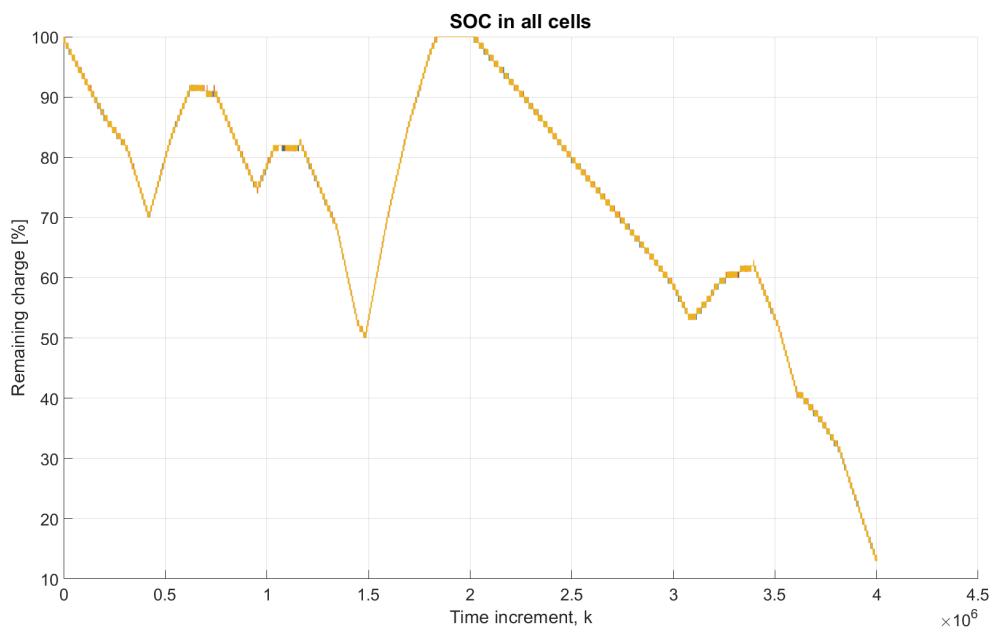


(e) Spenningskurve fra Figur D.4d med hvit støy.

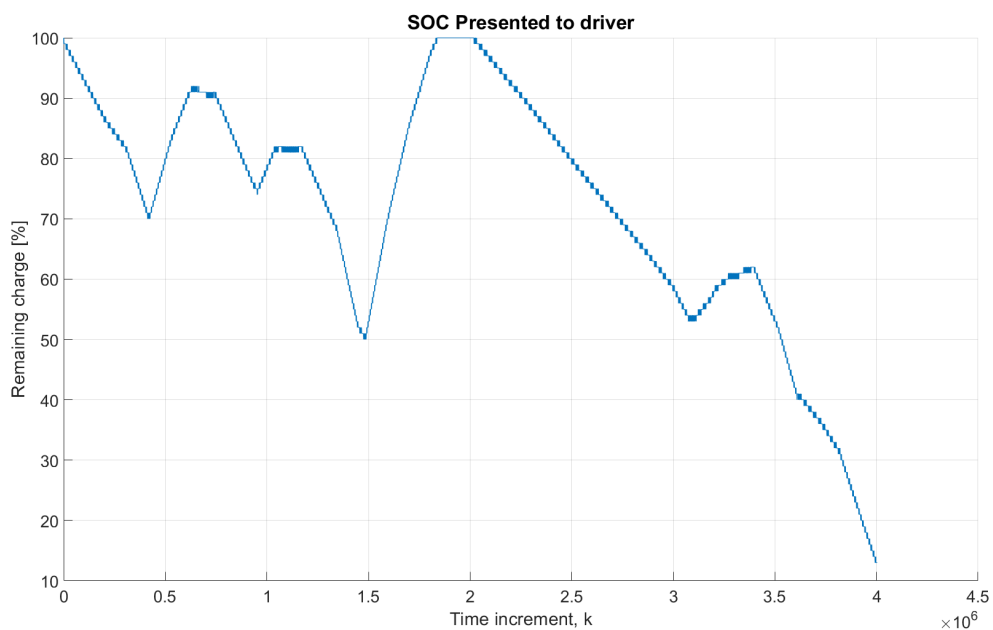




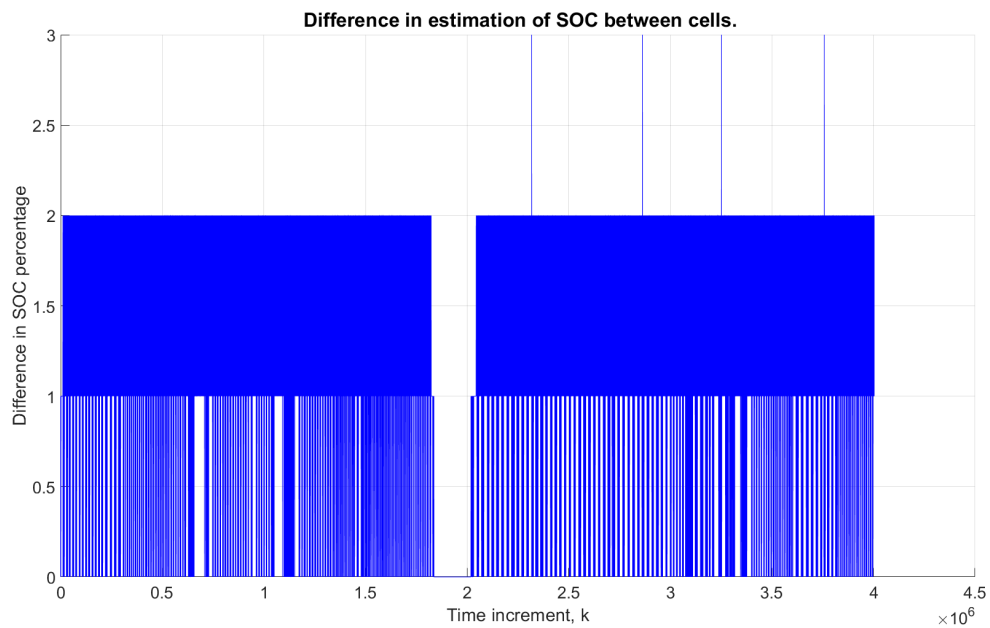
(f) Estimering av SOC fra Kalmanfilter



(g) SOC i hele present for alle celler i testen.



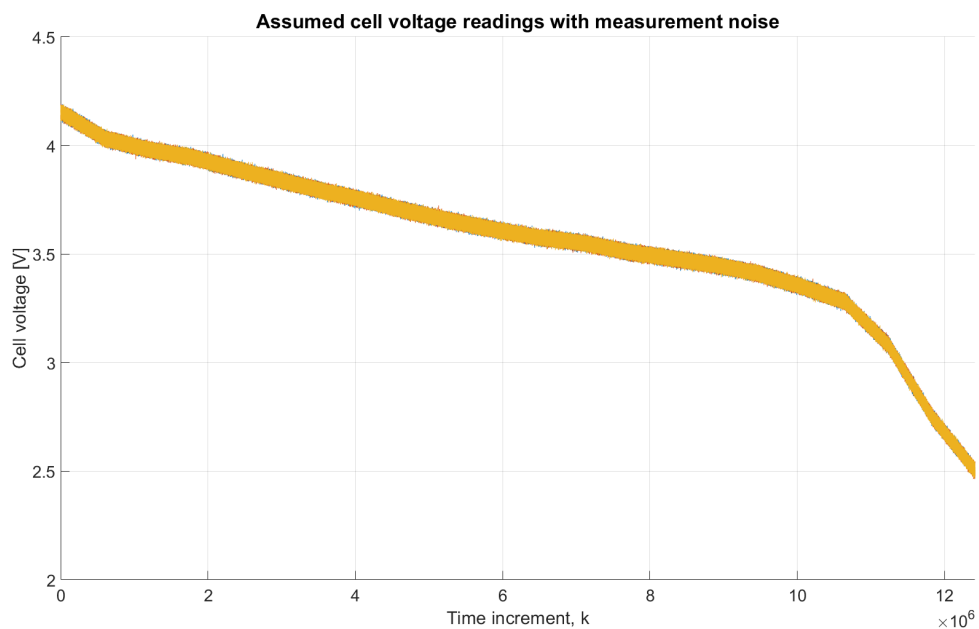
(h) Celle med lavest SOC. Dette er hva føreren får angitt som SOC.



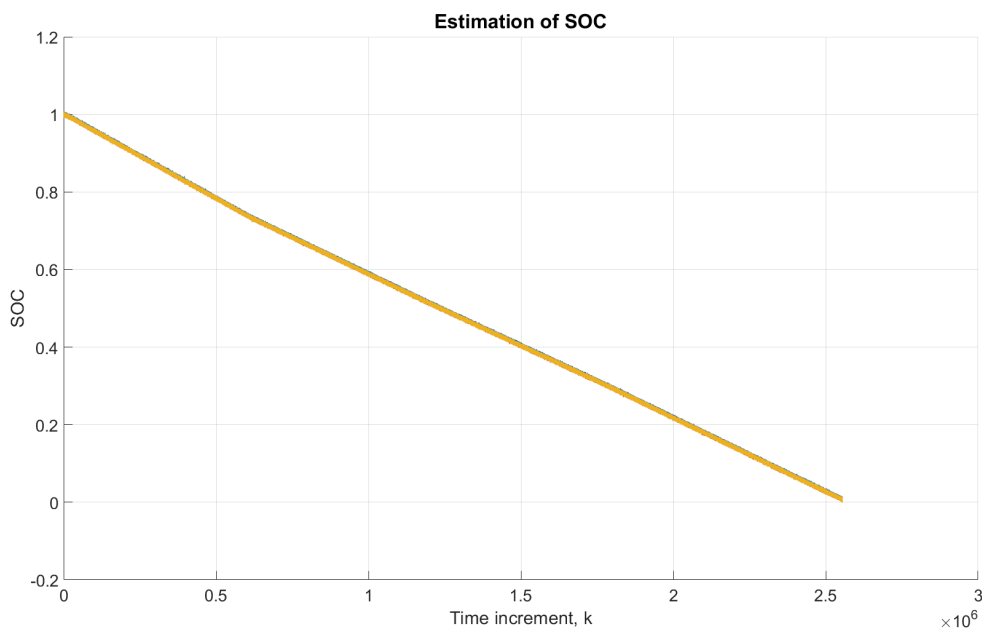
(i) Maksimum spredning i SOC-estimat.

**Figur D.4:** Kurver fra simulering med rolig lastmønster.

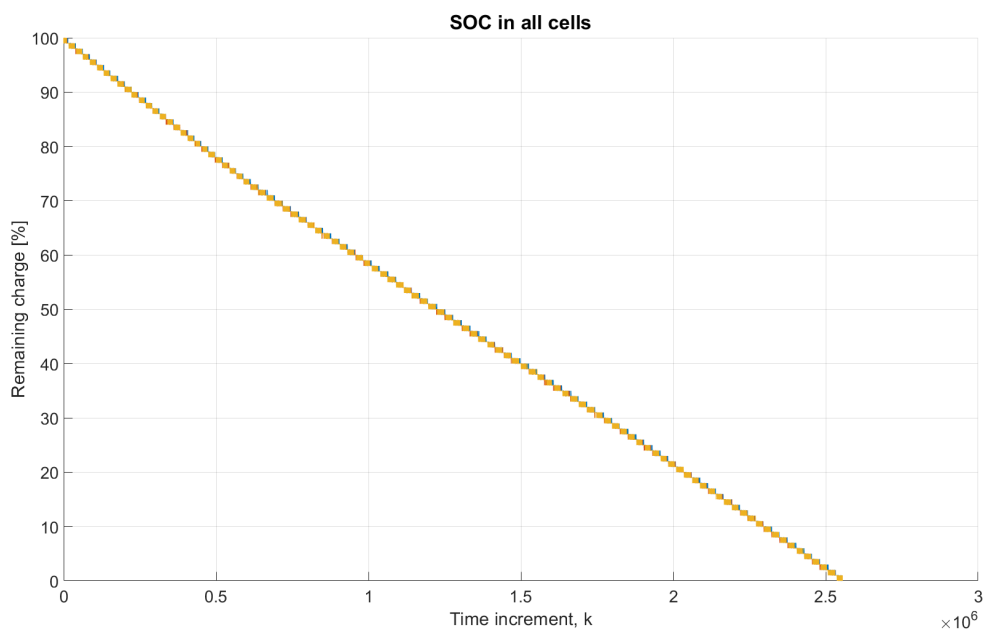
### Konstant lastmønster



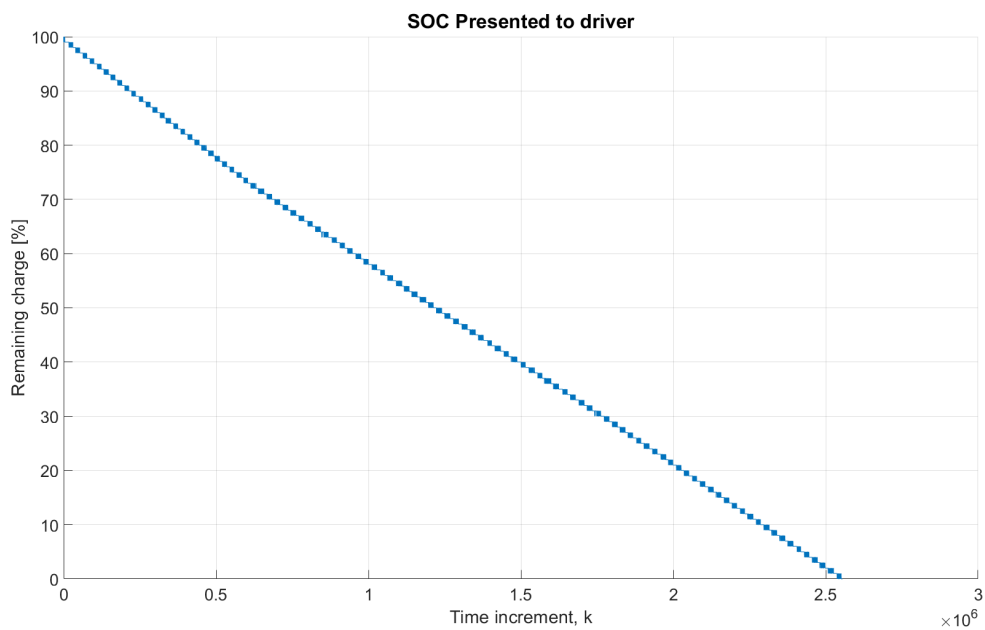
(a) Spenningskurve med hvit støy.



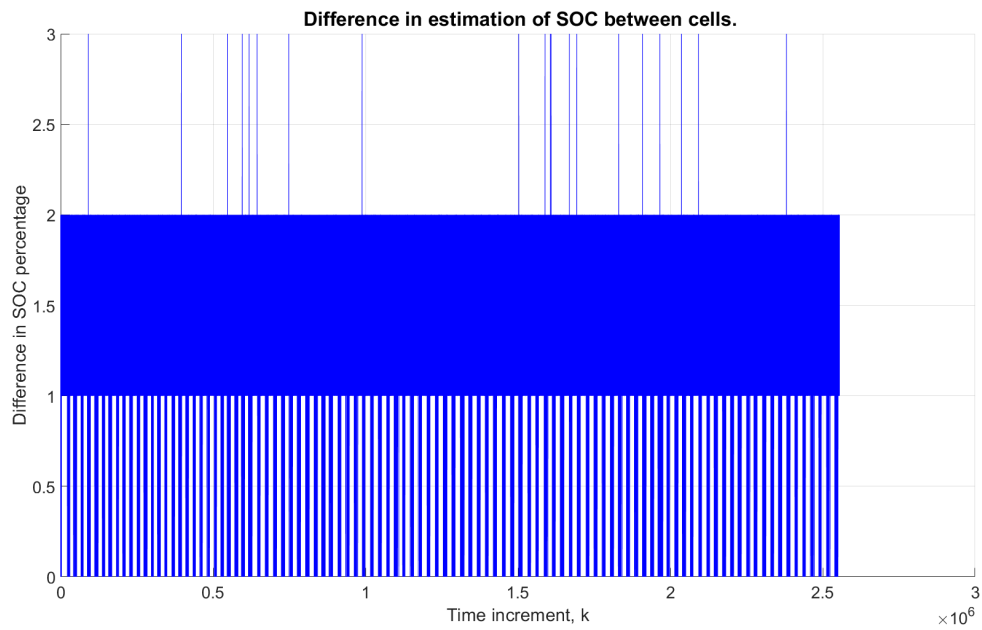
(b) Estimering av SOC fra Kalmanfilter



(c) SOC i hele prosent for alle celler i testen.



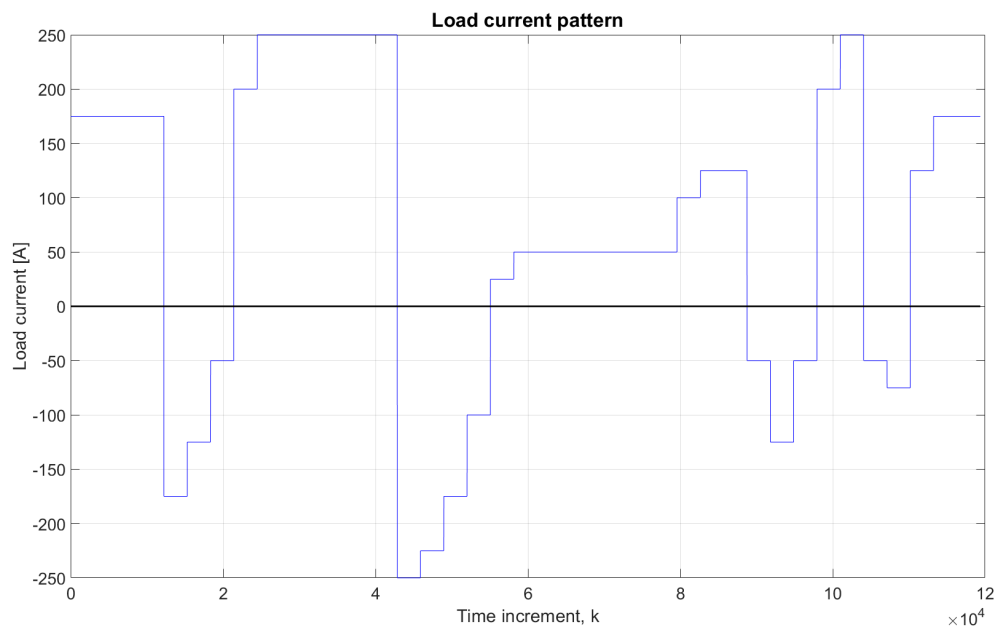
(d) Celle med lavest SOC. Dette er hva føreren får angitt som SOC.



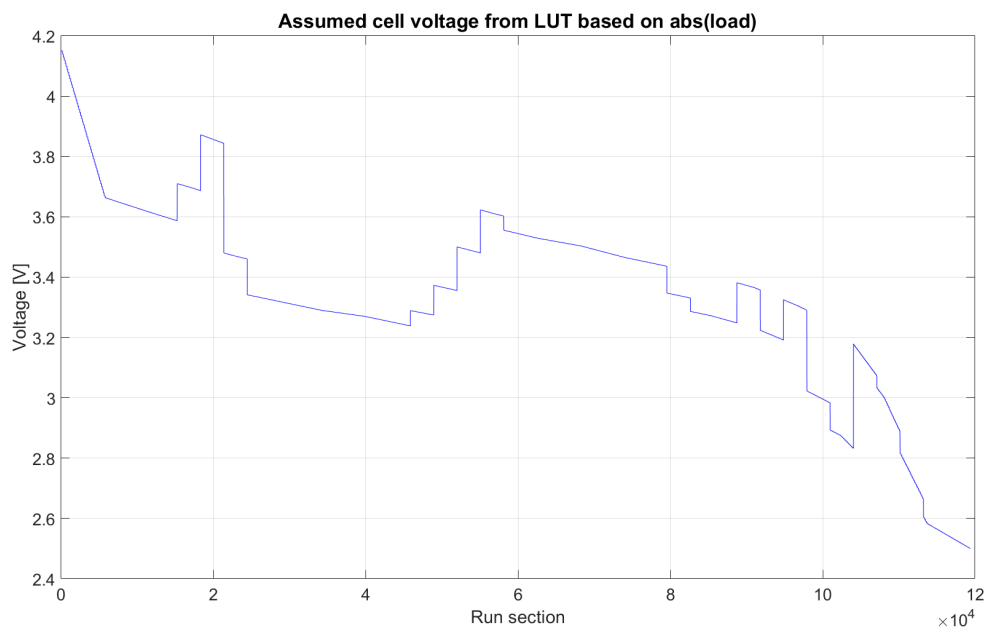
(e) Maksimum spredning i SOC-estimat.

Figur D.5: Kurver fra simulering med konstant 1C.

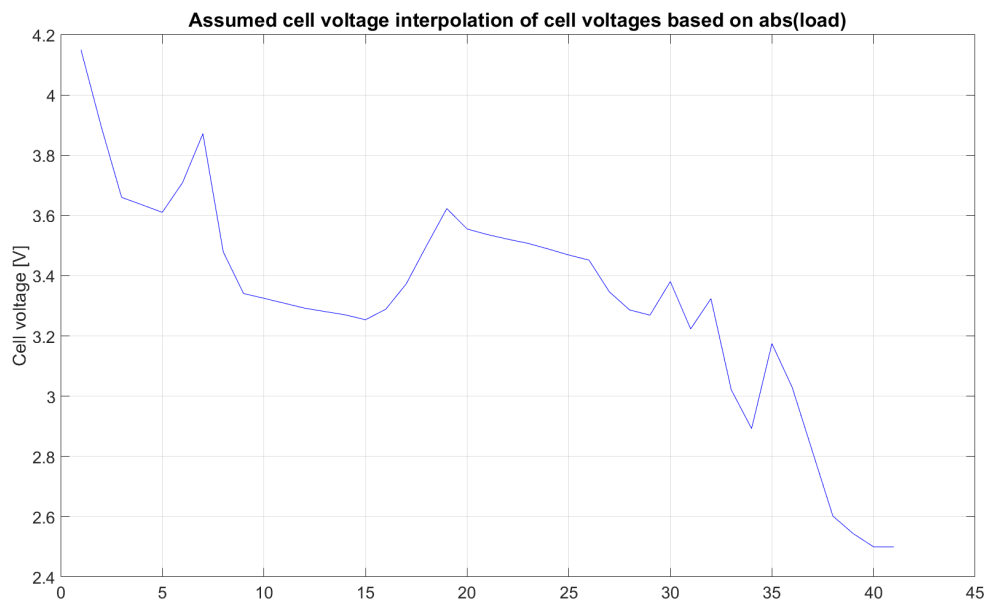
### Hardt lastmønster med 96 celledmoduler



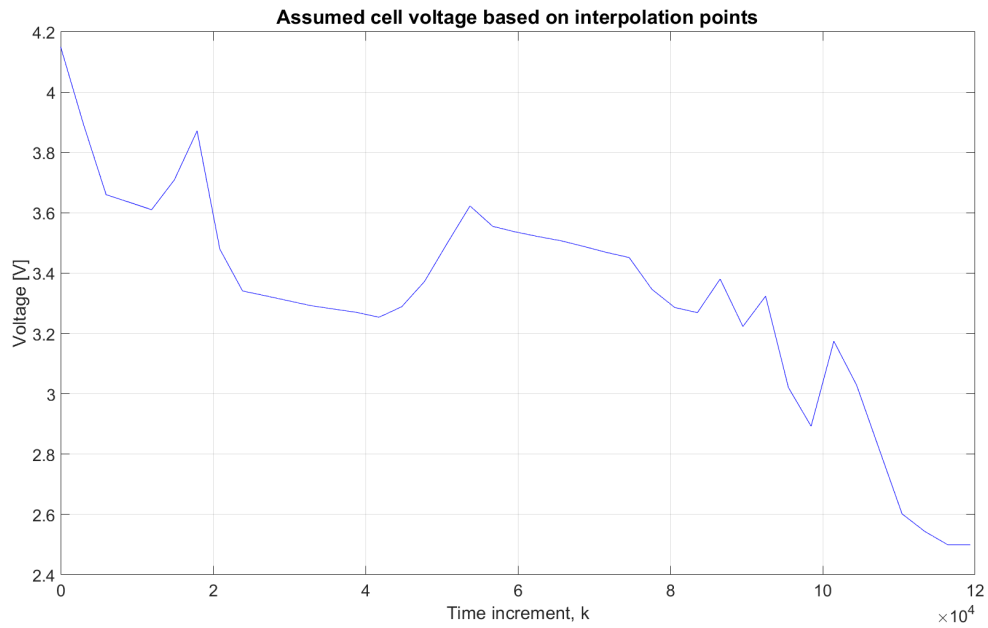
(a) Lastkurve.



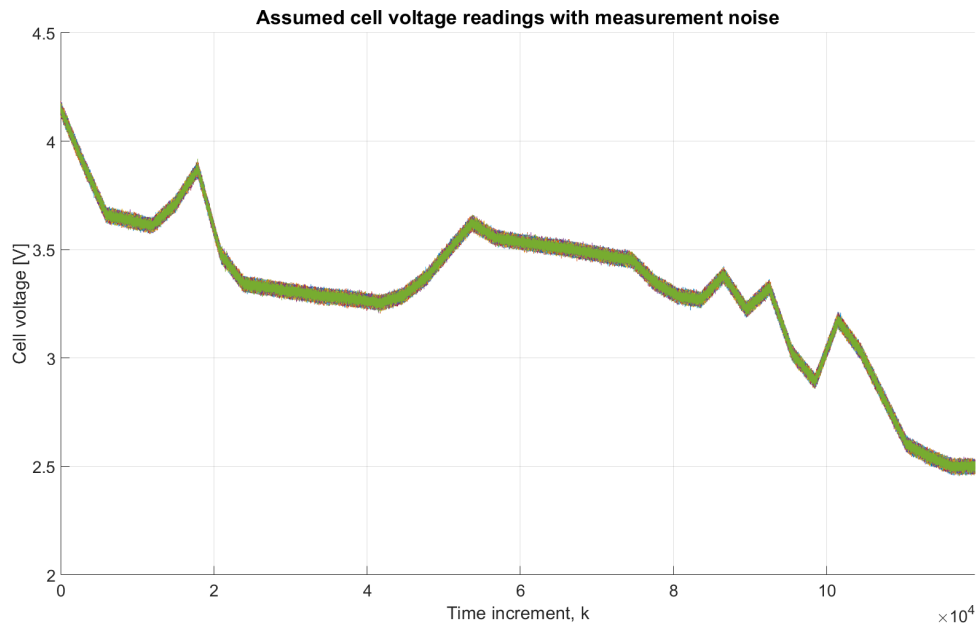
(b) Antatt spenningsmåling fra LUT.



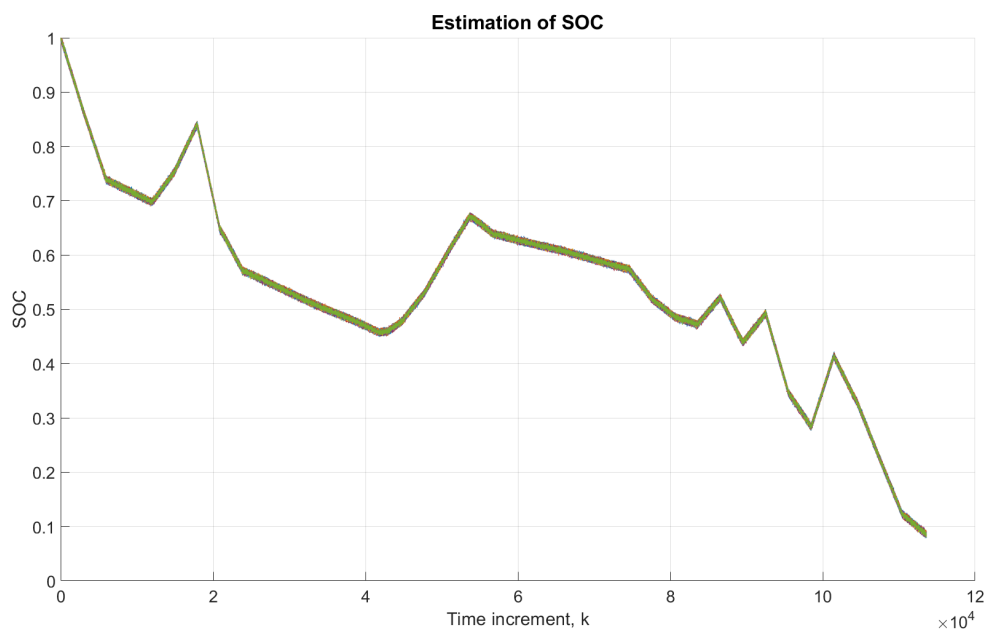
(c) Interpolert spenningsendring fra Figur D.6b.



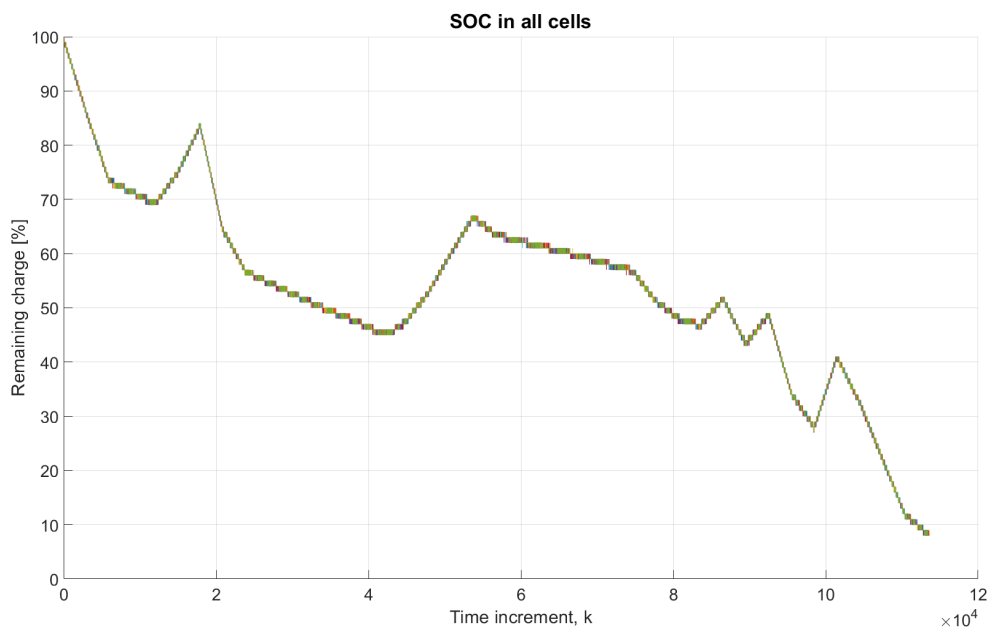
(d) Utvidet spenningskurve fra Figur D.6c som strekker seg over kjøretid.



(e) Spenningskurve fra Figur D.6d med hvit støy.

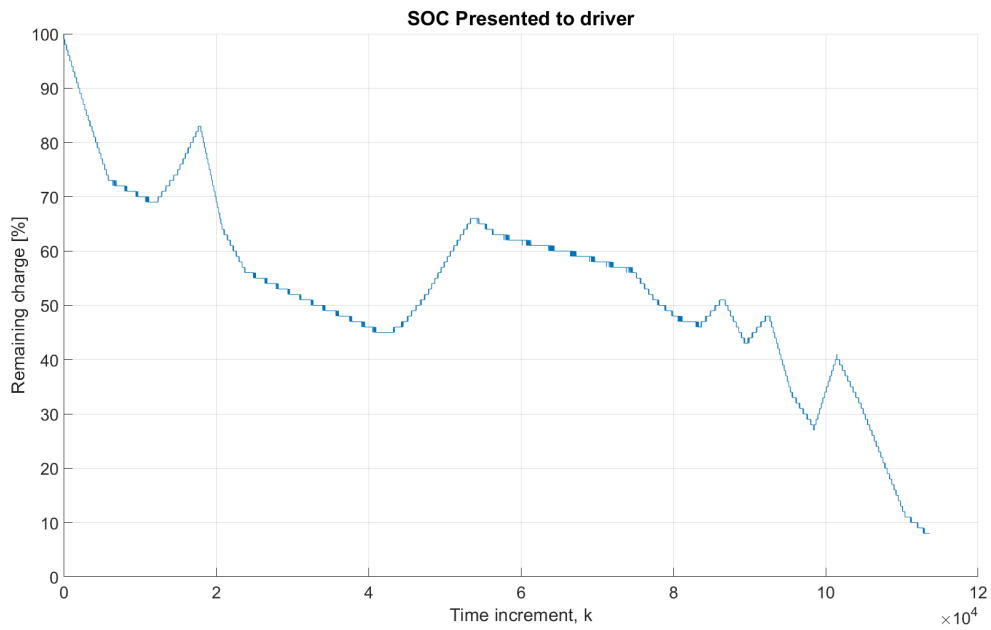


(f) Estimering av SOC fra Kalmanfilter

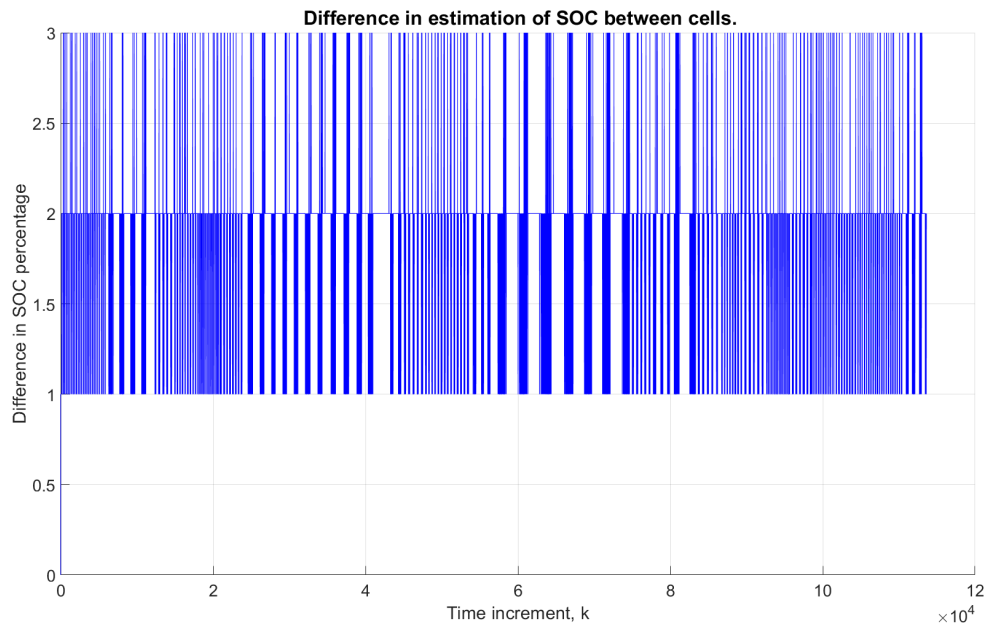


(g) SOC i hele present for alle celler i testen.





(h) Celle med lavest SOC. Dette er hva føreren får angitt som SOC.



(i) Maksimum spredning i SOC-estimat.

**Figur D.6:** Kurver fra simulering med 96 celler på hardt lastmønster.