
Robo-Sumo-Battle

Dokumentationsrapport

Gruppe #6

Villiam Holger Bo	201907166
Adam Ryager Høj	201803767
Rasmus Kahr	201803491
David Vestergaard Kristensen	201908226
Frederik William Lassen	201905905
Daniel Schultz	201709325
Simon Fogh Thomsen	201906472

Vejleder: John Rohde

Dato: 27. november 2020

Indhold

Ordliste	4
Forkortelser	5
1 Indledning	6
1.1 Koncept	6
1.1.1 Spilregler	6
2 Krav	10
2.1 Use Cases	10
2.2 Kravliste	10
2.2.1 Krav stillet af ASE	10
2.2.2 Overordnede krav til Robo-Sumo-Battle	11
2.2.3 Spillebane	11
2.2.4 SumoBot	11
2.2.5 Styringsenhed	12
3 Proces og metode	15
3.1 Udviklingsværktøjer	15
3.1.1 Software	15
3.1.2 Måleudstyr	15
4 Risk Assessment	16
5 Strukturering	18
5.1 Robo-Sumo-Battle Strukturering	18
5.2 Central Computer Strukturering	20
5.3 Styringsenhed Strukturering	22
5.4 SumoBot Strukturering	25
6 Analyse	28
6.1 Central Computer Analyse	28
6.1.1 Embedded Controller	29
6.1.2 Display	30
6.2 Styringsenhed Analyse	32
6.2.1 Analyse af lydgivere	32
6.2.2 Mikrofon	32
6.2.3 Analog signalbehandling	32
6.2.4 Joystick	34
6.2.5 Signalkonditionering	34
6.2.6 ADC	34
6.2.7 Digital signal processing	36
6.3 Sumobot Analyse	37
7 Design	43
7.1 Central Computer Design	43
7.1.1 Overordnet	43
7.1.2 Initialisering	43
7.2 Styringsenhed Design	45
7.2.1 ADC	45
7.3 SumoBot Design	46
8 Implementering	49
A Revisionshistorik	51

Tabeller

2.1	Fully dressed beskrivelse af Use Case 1: "Spil Robo-Sumo-Battle med joystick".	13
2.2	Fully dressed beskrivelse af Use Case 2: "Spil Robo-Sumo-Battle med mikrofon".	14
4.1	Risk Assessment	17
5.1	Blokbeskrivelse for Robo-Sumo-Battle	19
5.2	Blokbeskrivelse for Central Computer	21
5.3	Blokbeskrivelse for Styringsenhed [1/2]	23
5.4	Blokbeskrivelse for Styringsenhed [2/2]	24
5.5	Blokbeskrivelse for Sumobot [1/2]	26
5.6	Blokbeskrivelse for Sumobot [2/2]	27
6.1	Blokbeskrivelse for Central Computer: Embedded Controller	29
6.2	Blokbeskrivelse for Central Computer: Display	31
6.3	Blokbeskrivelse for Styringsenhed: Mikrofon	32
6.4	Blokbeskrivelse for Styringsenhed: Analog signalbehandling	33
6.5	Blokbeskrivelse for Styringsenhed: Joystick	34
6.6	Blokbeskrivelse for Styringsenhed: Signal konditionering	35
6.7	Blokbeskrivelse for Styringsenhed: ADC	36
6.8	Vurdering af ADC	36
6.9	Blokbeskrivelse for Central Computer IF	38
6.10	Blokbeskrivelse for Attack Sensor	38
6.11	Blokbeskrivelse for PSU	39
6.12	Blokbeskrivelse for Microcontroller]	40
6.13	Blokbeskrivelse for Batteri]	41
6.14	Blokbeskrivelse for Motor]	41
6.15	Blokbeskrivelse for Motorstyring	42

Figurer

1.1	Rigt billede der viser hvordan produktet er tiltænkt.	6
1.2	Illustration der viser tonaliteten af en blokfløjte. Her ses hvordan styretonerne A, B, C og D spilles	7
1.3	Selve spilpladen	8
1.4	Spillepladen set fra siden	8
1.5	Spillepladen set fra den ene spillers side	9
2.1	Use Case diagram der viser de tre use cases for systemet Robo-Sumo-Battle og aktørernes relation til disse.	10
2.2	Sekvensdiagram der viser samspillet mellem aktørerne for use cases "Styr SumoBot med joystick" og "Styr SumoBot med lyd".	11
5.1	Robo-Sumo-Battle BDD	18
5.2	Robo-Sumo-Battle IBD	18
5.3	Central Computer BDD	20
5.4	Central_Computer IBD	20
5.5	IBD for Styringsenhed	22
5.6	SumoBot BDD	25
5.7	SumoBot IBD	25
6.1	Central Computer System Sekvensdiagram	28
6.2	Embedded Controller I/O	29
6.3	Display I/O	30
6.4	Udsnit af mikrofonblokken	32
6.5	Udsnit af blokken til analog signalbehandling	32
6.6	Flow der viser funktionaliteten af filteret	33
6.7	Udsnit af joystickblokken	34
6.8	Udsnit af blokken til signalkonditionering	35
6.9	Udsnit af ADC-blokken	35
6.10	SumoBot System Sekvensdiagram	37
6.11	Central Computer IF IBD	37
6.12	Attack Sensor IBD	38
6.13	PSU IBD	39
6.14	Microcontroller IBD	39
6.15	Batteri IBD	40
6.16	Motor IBD	41
6.17	Attack Sensor IBD	42
7.1	Statemachine over Central Computer	43
7.2	Sekvensdiagram over UC1+2	44
7.3	Klassediagram	45
7.4	Domænemodel	46
7.5	Sekvensdiagram over initialisering	47
7.6	Forskellige input modes for Delta Sigma ADC	48

Ordliste

access point

Trådløst WiFi-netværk hvor IEEE 802.11[1] kompatible-enheder kan tilkoble sig.

aktuator

En enhed der tillader et system at interagere med den fysiske verden; f.eks. en motor.

attacksensor

Sensor på SumoBots til detektering af sammestød.

bil-kontrol

Central enhed der modtager inputs fra controller og sender signal til begge SumoBots.

controller

Enhed som spiller bruger til at styre en SumoBot.

DFT

Diskret Fourier-Transformation, en matematisk funktion som benyttes til at regne fra tids-til frekvensdomæne.

gameplay-modul

Beskrivelse af Gameplay-Modul.

joystick

En analog styringspind med variabel X- og Y-retning.

Robo-Sumo-Battle

En kamp mellem to SumoBots.

scoreboard

Pointtavle der viser hvilken spiller der er tættest på at vinde.

spilpladen

Området hvor spillet afvikles.

spiltilstand

Beskrivelse af Spiltilstand.

SumoBot

En lille robot der skal styres for at vinde spillet.

Forkortelser

DSP

Digital Signal Processing.

MFB

Multi-Feedback.

PSoC

Cypress PSoC 5LP.

PSU

Power Supply Unit.

RPi

Raspberry Pi.

SNR

Signal-Noise-Ratio.

TBD

To be defined - indikere at et afsnit har store mangler..

TBR

To be refined - indikere at et afsnit mangler mere information..

UI

User interface.

1. Indledning

1.1 Koncept

Kender vi ikke alle problemet med en drukmås der har drukket for mange øller og får lyst til at skribe i en mikrofon? Det hidtil eneste kendte produkt til at afhjælpe dette problem, har været karaoke, dog med den ulempe at alle andre på baren skal høre på det. Robo-Sumo-Battle vil specifikt afhjælpe dette problem ved i stedet at lade personen bruge sin stemme til at styre to små chubby robotter der dyrker sumo-brydning, som samtidig er sjov at se på - vi bruger således en gammel japansk tradition til at afhjælpe problemerne med en nyere japansk tradition.

Produktet er et spil for to spillere, hvor hver spiller styrer en to-hjulet robot rundt på en bane.

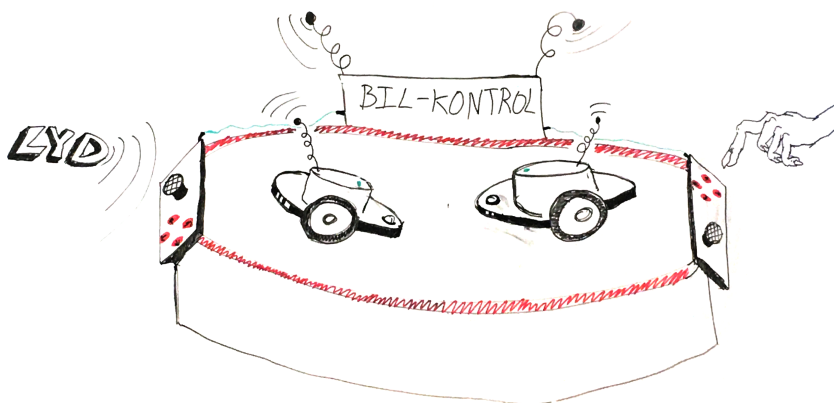
Styringen foregår via én kontrolenhed pr. spiller og vil være baseret på sensorinput i form af lyd igennem en mikrofon og bevægelse igennem et *d-pad*. Dertil kan man aktivere forskellige *gamemodes* hvor der er udviklet mere eller mindre arbitrære styringer for at skrue sværhedsgraden op. Disse signaler bliver digitalt behandlet og sendt til robotterne som styrekommandoer via den embedded software i spilplatformen. Spillet er tænkt udført som en traditionel japansk sumokamp — de to robotter mødes i en ring og kæmper. Vinderen er den der fratager alle modstanders liv først - eller den sidste robot i ringen.

1.1.1 Spilregler

- De to robotter anbringes på et markeret startfelt.
- Når robotterne er anbragt korrekt på deres startfelter, kan en nedtælling fra 3 sekunder startes ved, at begge spillere trykker på hver deres knap, hvorefter spillet går i gang.
- Hver robot har fra spillets start 3 liv.
- Hver runde varer maksimalt 1 minut.
- I en rundes varighed gælder:
 - Hvis modstanderen skubbes udover den markerede bane, vinder hele spillet.
 - Ved at påkøre modstanderen bagfra eller fra siden, mister modstanderen et liv.
 - Hvis en spiller selv bringer sig udover den markerede bane, vinder modstanderen.
- Ved tab af liv placeres hver robot ved deres respektive startfelt.

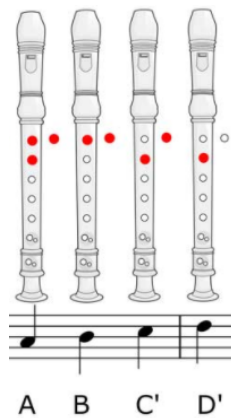
Til styringen af robotterne medfølger der til spillet et instrument med veldefinerede frekvensområder: en blokfløjte. Hertil også et *d-pad*.

Til disse instrumenter tilknytter der sig som udgangspunkt følgende kommandoer: frem, tilbage, højre og venstre. Disse kommandoer eksekveres med udgangspunkt i figur 1.2 som følgende:

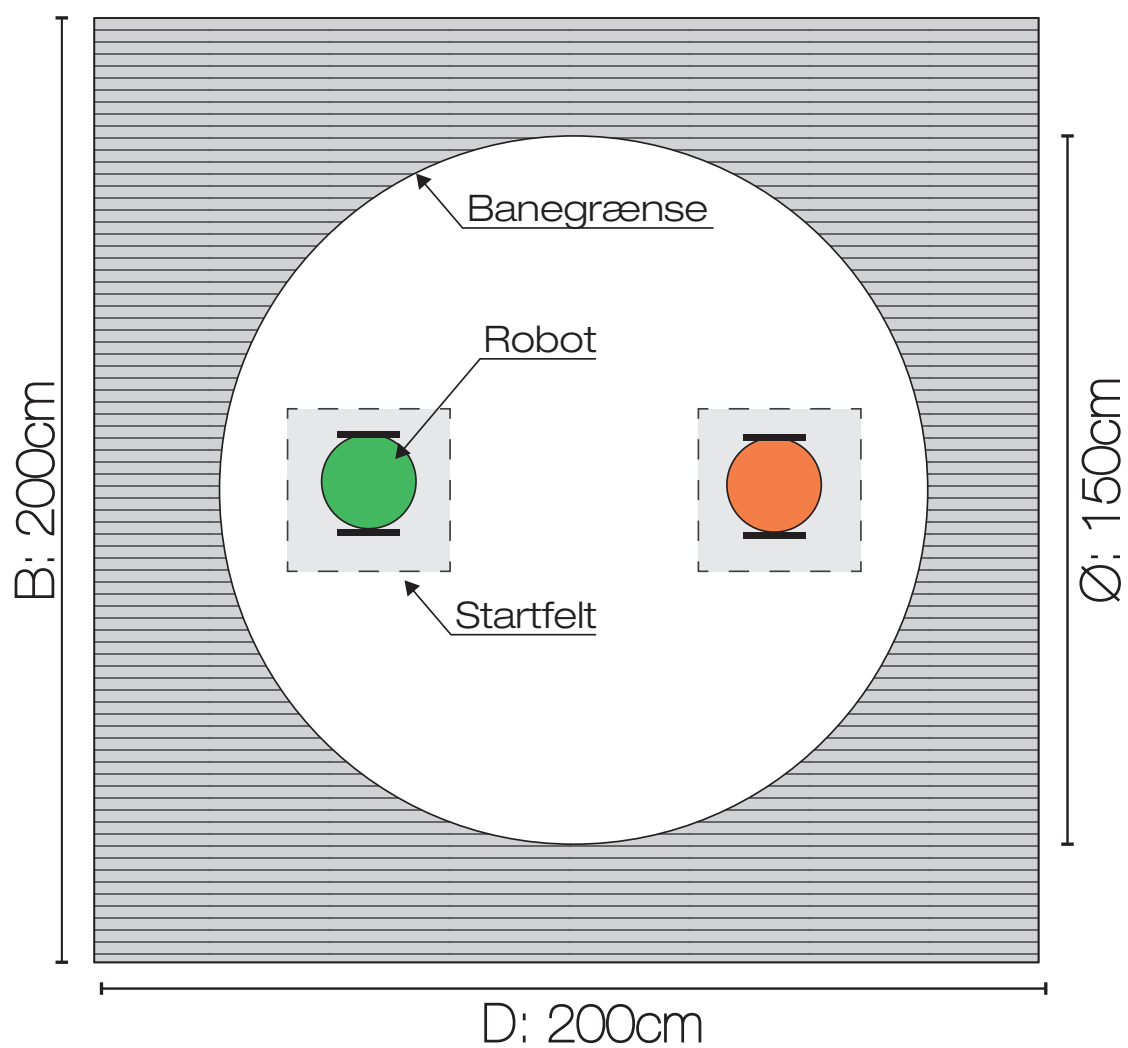


Figur 1.1: Rigt billede der viser hvordan produktet er tiltænkt.

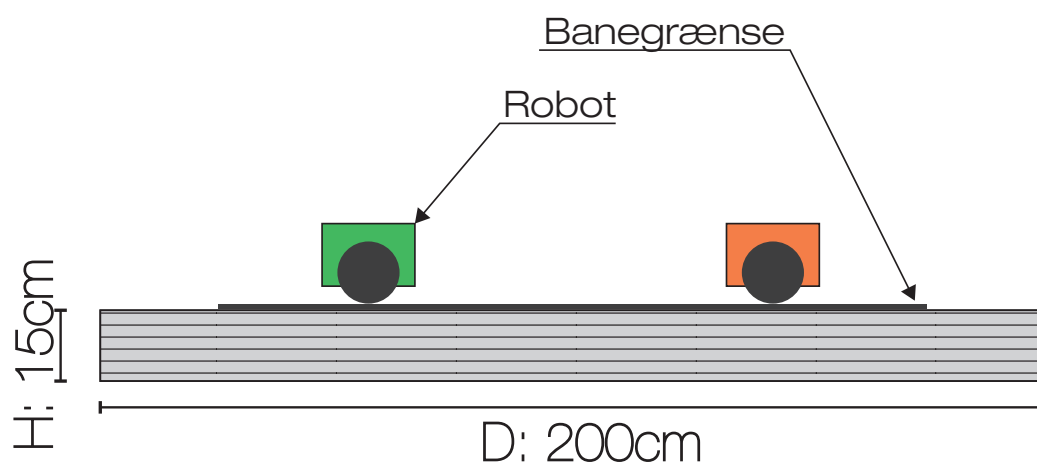
- A: Fremad
- B: Bagud
- C': Venstre
- D': Højre



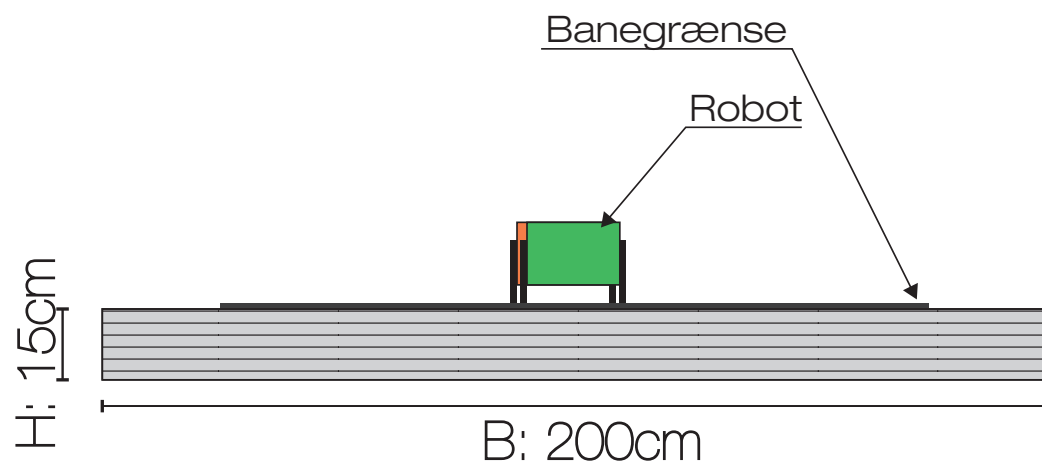
Figur 1.2: Illustration der viser tonaliteten af en blokfløjte. Her ses hvordan styretonerne A, B, C og D spilles



Figur 1.3: Selve spilpladen



Figur 1.4: Spillepladen set fra siden

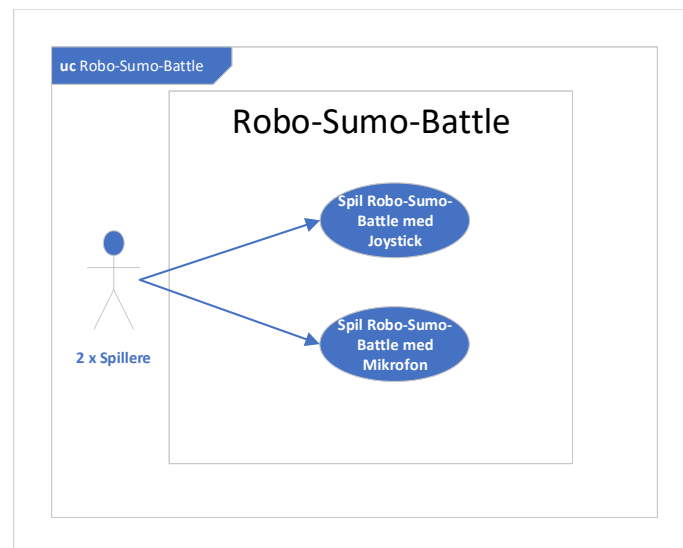


Figur 1.5: Spillepladen set fra den ene spillers side

2. Krav

2.1 Use Cases

I det følgende beskrives de use cases der er defineret for systemet, således at de opstillede krav kan opnås. Der er defineret to use cases som fremgår i figur ?? . For de to use cases er der opstillet et sekvensdiagram i figur 2.2 der beskriver sammenspillet mellem aktørerne for at udføre casen. Efterfølgende er de to use cases beskrevet på *fully-dressed* form.



Figur 2.1: Use Case diagram der viser de tre use cases for systemet Robo-Sumo-Battle og aktørernes relation til disse.

Use Case 1

Spil Robo-Sumo-Battles med joystick. Beskrevet i tabel 2.1.

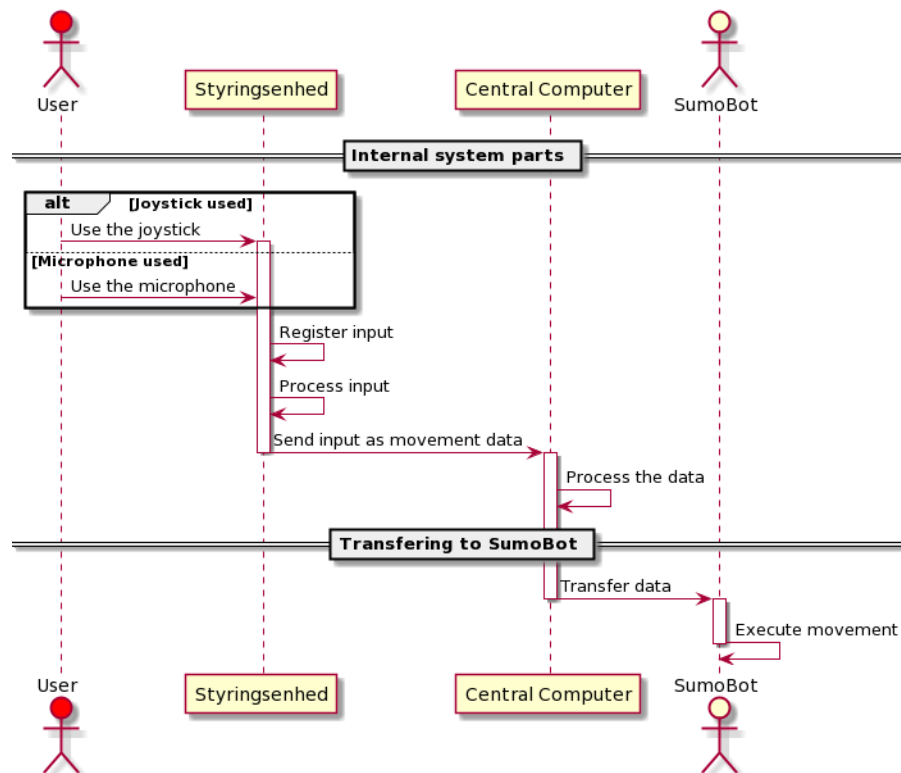
Use Case 2

Spil Robo-Sumo-Battles med mikrofon. Beskrevet i tabel 2.2.

2.2 Kravliste

2.2.1 Krav stillet af ASE

1. Systemet **SKAL** indeholde embedded software.
2. Systemet **SKAL** indeholde aktorater.
3. Systemet **SKAL** indeholde trådløs kommunikation.
4. Systemet **SKAL** indeholde et Linux system
5. Systemet **SKAL** gøre brug af PSoC
6. Systemet **SKAL** gøre brug af sensor(er)
7. Systemet **SKAL** indeholde et UI



Figur 2.2: Sekvensdiagram der viser samspillet mellem aktørerne for use cases “Styr SumoBot med joystick” og “Styr SumoBot med lyd”.

2.2.2 Overordnede krav til Robo-Sumo-Battle

1. systemet **SKAL** kunne understøtte minimum 2 spillere.
2. Systemet **SKAL** kunne registrere når en SumoBot overskrider spillebanens afgrænsning.
3. Systemet **BØR** kunne understøtte flere end 2 spillere.
4. Systemet **BØR** understøtte flere spiltilstande, end den allerede beskrevet under spilregler 1.1.1

2.2.3 Spillebane

1. Spillebanen **SKAL** maksimalt være Ø1.5 meter.
2. Spillebanen **SKAL** være et jævnt plan.
3. Spillebanen **BØR** veje mindre end 10 kg.

2.2.4 SumoBot

1. SumoBot **SKAL** maksimalt være 20 × 20 × 20cm.
2. SumoBot **SKAL** kunne kollideres med hinanden fra alle sider, ved max hastighed, uden at tage skade.
3. SumoBot **SKAL** kunne tilgås trådløst fra minimum tre meters afstand.
4. SumoBot **SKAL** veje under 2.5 kg med alt monteret.
5. SumoBot **BØR** kunne køre kontinuert, ved max hastighed, vinkelret mod en lodret flade, uden at blive beskadiget.
6. SumoBot **BØR** have batterilevetid til kontinuere kørsel ved max hastighed, i minimum 3 minutter.

7. SumoBot **BØR** kunne holde til 10cm frit drop på laminatgulv.
8. SumoBot **BØR** reagere respektivt på spillerens styring, inden for 1 sekund.

2.2.5 Styringsenhed

1. Styringsenheden **SKAL** kunne konvertere force på joystick til data som kan omsættes til bevægelse af en SumoBot'er.
2. Styringsenheden **SKAL** have en mikrofon pr. spiller, til at opfange lyd i frekvensområdet 300 Hz til 3.000×10^3 Hz.
3. Styringsenheden **BØR** have en SNR!¹ på over 60 dB.
4. Styringsenheden **BØR** have en input følsomhed på $(8 \pm 2) \text{ mV Pa}^{-1}$.
5. Styringsenheden **BØR** kunne konvertere lyd fra 100 Hz til 3.000×10^3 Hz til data som kan benyttes som en del af spillet.

¹SNR!

Tabel 2.1: Fully dressed beskrivelse af Use Case 1: "Spil Robo-Sumo-Battle med joystick".

Use Case 1	"Spil Robo-Sumo-Battle med joystick"
Mål:	At to spillere, styrer sine respektive SumoBot, indtil en spiller har vundet
Initiering:	Spillere trykker start på deres respektive start knapper.
Primær Aktør:	2X Spillere
Sekundær Aktør:	Ingen
Antal samtidig forekomster:	Ingen
Prækondition:	Begge SumoBot har fuld liv og står på deres respektive startfelter.
Postkonditioner:	<ul style="list-style-type: none"> • En spiller har vundet Robo-Sumo-Battle viser vinderen. • Robo-Sumo-Battle klar til nyt spil.

Hovedscenarie:

1. Spillere tilgår deres respektive joystick og trykker start.
2. Robo-Sumo-Battle viser nedtælling til start.
3. Spillere styrer deres respektive SumoBots retning og hastighed trådløst via joystick.
4. Den ene SumoBot aktivere den anden SumoBot attacksensor.
[Udvidelse 1: En SumoBot forlader spille banen]
5. Robo-Sumo-Battle viser at den angrebet SumoBot har mistet et liv.
6. spillere placere deres SumoBot på dens respektive startfelt og trykker start.
7. Robo-Sumo-Battle viser nedtælling og starter ny runde.
8. Der spilles til en SumoBot har mistet alle liv.
9. Robo-Sumo-Battle viser vinderen af spillet.
10. Spillere placere SumoBot på deres respektive startfelter.

Udvidelser / Undtagelser:

Udvidelse 1: En SumoBot forlader spillebanen

1. SumoBot der har forladt spillebanen mister alle liv.
2. Use case fortsætter fra punkt 9.

Tabel 2.2: Fully dressed beskrivelse af Use Case 2: "Spil Robo-Sumo-Battle med mikrofon".

Use Case 2	"Spil Robo-Sumo-Battle med mikrofon"
Mål:	At to spillere, styrer sine respektive SumoBot, indtil en spiller har vundet
Initiering:	Spillere trykker start på deres respektive start knapper.
Primær Aktør:	2X Spillere
Sekundær Aktør:	Ingen
Antal samtidig forekomster:	Ingen
Prækondition:	Begge SumoBot har fuld liv og står på deres respektive startfelter.
Postkonditioner:	<ul style="list-style-type: none"> • En spiller har vundet Robo-Sumo-Battle viser vinderen. • Robo-Sumo-Battle klar til nyt spil.

Hovedscenarie:

1. Spillere tilgår deres respektive mikrofoner og trykker start.
2. Robo-Sumo-Battle viser nedtælling til start.
3. Spillere styrer deres respektive SumoBots retning og hastighed trådløst via påvirke mikrofonen med lyd.
4. Den ene SumoBot aktivere den anden SumoBot attacksensor.
[Udvidelse 1: En SumoBot forlader spille banen]
5. Robo-Sumo-Battle viser at den angrebet SumoBot har mistet et liv.
6. spillere placere deres SumoBot på dens respektive startfelt og trykker start.
7. Robo-Sumo-Battle viser nedtælling og starter ny runde.
8. Der spilles til en SumoBot har mistet alle liv.
9. Robo-Sumo-Battle viser vinderen af spillet.
10. Spillere placere SumoBot på deres respektive startfelter.

Udvidelser / Undtagelser:

Udvidelse 1: En SumoBot forlader spillebanen

1. SumoBot der har forladt spillebanen mister alle liv.
2. Use case fortsætter fra punkt 9.

3. Proces og metode

3.1 Udviklingsværktøjer

Rapporten er sat i L^AT_EX med kpfonts.

Desuden er følgende software/hardwareløsninger brugt til at udarbejde Robo-Sumo-Battle-projektet.

3.1.1 Software

- Overleaf
- Mathworks MATLAB R2020a
- LTspice XVII
- National Instruments Multisim
- PlantUML
- Microsoft Visio
- GitHub

Målinger i projektet er foretaget med følgende udstyr, hvoraf noget har været tilgængeligt via. Aarhus Universitets elektroniklaboratorier

3.1.2 Måleudstyr

- Analog Discovery 2
- Udstyr udlånt af Aarhus Universitet
 - KEYSIGHT DSOX20002A Osiloskop
 - AIM TTi 1604 Multimeter
 - AIM TTi TG320 Funktionsgenerator
 - AIM TTi EL302RT Spændingsforsyning

4. Risk Assessment

Arbejdet med risikoanalyse er dels foregået på et overordnet plan hvor potentielle risici for projektet som helhed er identificeret, såvel som et mere blokspecifikt plan, hvor risici for specifikke blokke er identificeret. Disse er naturligvis identificeret med henblik på at minimere disse.

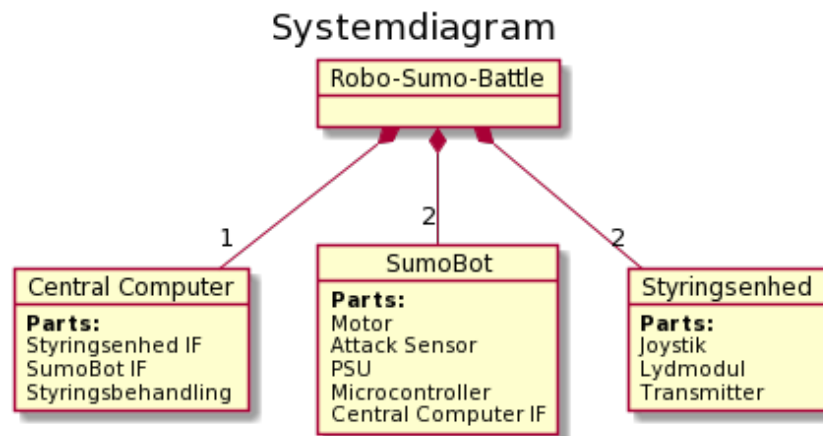
Den egentlige risikoanalyse som ses i [4.1](#) er udarbejdet i forlængelse af struktur-afsnittet hvor de første tanker om analyseafsnittet har været påbegyndt, således at risici er blevet identificeret tidligt for at kunne forebygge dem tidligst muligt, men ikke tidligere end der eksisterede et overblik over projektet. Det sagt, er identificering af yderligere potentielle risici naturligvist et fortløbende arbejde, ligesom risikominimeringen vil strække sig hele vejen igennem projektet.

Tabel 4.1: Risk Assessment					
Blok	Description	Probability	Consequence	Impact	Risk Mitigation Plan
Processor	Vi stiller krav til processorkraft som ikke kan køre på et embedded system (2xDFT med lille tidsinterval imellem, wifi, spilstyring mv.)	2	3	6	Krav til processorkraften overvejes nøje når der vælges processor. Hvis de simultane DFT analyser er en udfordring overvejes om dette skal foregå på sin egen processor.
SumoBot IF	Det lykkedes ikke at overføre data mellem SumoBot og Central Computer vha. Wifi	3	5	15	Vi kontakter vejleder og eller undervisere, der har kendskab til socketprogrammering
SumoBot IF	Wifi er for ustabil til kravet om konstant overføring af information	1	5	5	Vi kontakter vejleder og eller undervisere, der har kendskab til WiFi
Display	Viser ikke den ønsket information.	2	2	4	Spilinformation omlægges til at vises gennem LED'er
Styrings-enhed IF	Mikrofonens output er for udefinerbart til at kunne styre en SumoBot med	3	4	12	Mikrofonens output analyseres i blokken "styrings-enhed" hvor der er flere resourcer til at omdanne til et brugbart input
Overordnet	De individuelle delblokke kan ikke forbindes til et samlet system	3	3	9	Delgrupperne i projektgruppen har løbende kommunikation og tilpasser delblokkende med hinanden.
Overordnet	Omfanget af SumoBot er for stort til at kunne nås på den afsatte tid	3	3	9	Løbende evaluering og tilretning af ønsket til endeligt produkt
ADC	Det er ikke muligt at sende data pga for mange bitfejl	1	5	13	Vi bruger den frivillige øvelse i MSE på at få indsigt i ADC'er
Mikrofon	Blokfløjtsens frekvensspekter er for snæver til at kunne generere veldefinerede outputs	3	2	6	Der undersøges andre lydkilder
PSU	PSU støj og transienter forstyrrer styrings-enhed protokollen	3	5	7	Vejledere og undervisere med kendskab til PSU'er kontaktes. Alternativt bruges batteri eller købes/lånes velfungerende PSU'er
Overordnet	Omfanget af spillets realisering kræver håndværksmæssige kompetencer som vi ikke besidder	2	5	10	Der søges hjælp på værkstedet i Shannon eller hos bekendte
Mikrofon	Støj fra fysiske omstændigheder (menneskesnak, musik) gør analyse af mikrofon inputtet umuligt	3	5	15	Der søges om rådgivning i AUDIOLab. Der afsættes tid til at lave akustisk regulering til mikrofon inputtet.

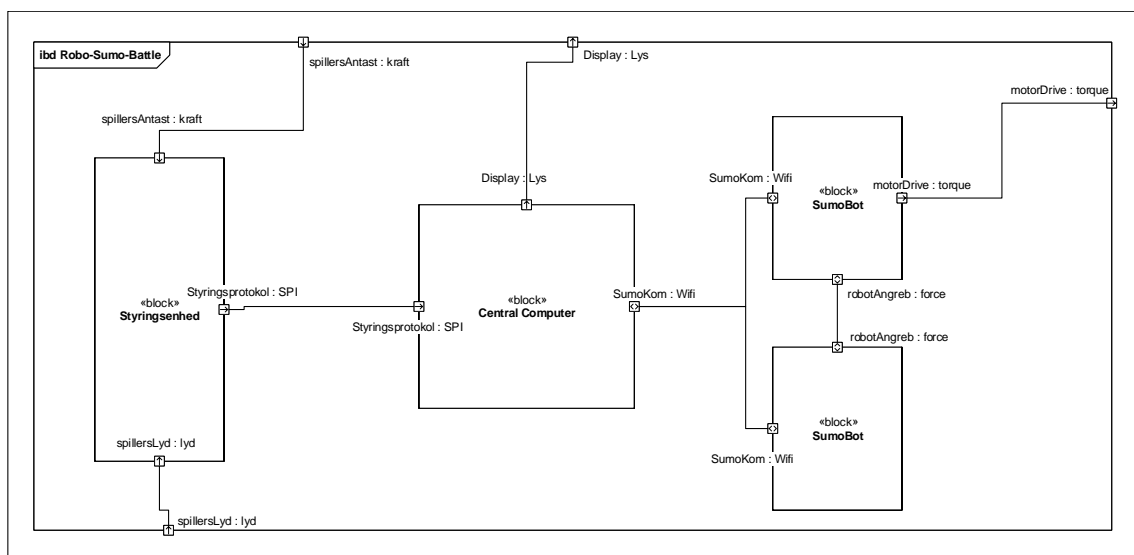
5. Strukturering

5.1 Robo-Sumo-Battle Strukturering

For Robo-Sumo-Battle er der udviklet struktur som illustreres ved figur 5.1, og figur 5.2. Delblokkene og interfaces imellem disse, som skitseret i IBDet figur 5.2 er yderligere beskrevet i blokbeskrivelsen som findes i tabel 5.1



Figur 5.1: Robo-Sumo-Battle BDD



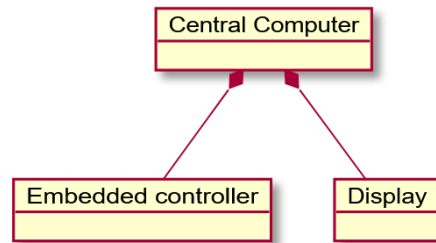
Figur 5.2: Robo-Sumo-Battle IBD

Tabel 5.1: Blokbeskrivelse for Robo-Sumo-Battle

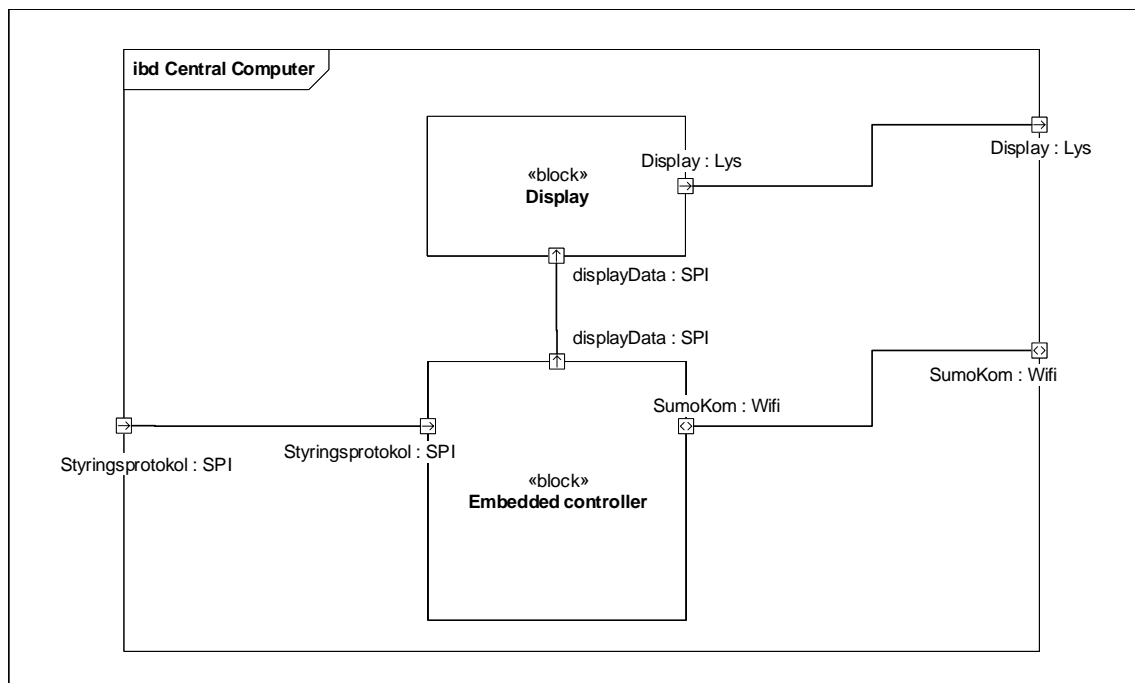
Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Styringsenhed	Omdanner fysisk lyd og joystick påvirkning til data Central Computeren kan læse.	In SpilleresAntast	Kraft	Spillerens fysiske manipulering af joysticket	• •
		In SpillersLyd	Lyd	Spillerens fysiske lyd fra stemme eller fløjte	• •
		Out Styringsprotokol	SPI	Digitalt samplet signal af det valgte styringsinput (lyd el-ler joystick) til Central Computer	• •
Central Computer	Videregiver information fra Styringsenhed til Sumobot, trådløst. Informere spillerene om spillets status. Liv, nedtælling etc.	In Styringsprotokol	SPI	Digitalt samplet signal af det valgte styringsinput (lyd el-ler joystick) fra styringsenheden	• •
		Out Display	Lys	Displayets løbende information til spillerne omkring spilstatus (re-sterende liv, resterende spilletid mv.)	• •
		Inout SumoKom	Wifi	Tovejskommunikation med SumoBot's hvor der sendesinformation omkring retning og hastighed og modtage information om et evt. angreb	• •
SumoBot	Omdanner styringsinfotmation fra Central Computeren, til bevægelse. Informere Centralcomputer om liv er mistet	Inout SumoKom	Wifi	Tovejskommunikation med Central Computer hvor dermodtages information omkring retning og hastighed ogsendes information om et evt. angreb	• •
		Inout RobotAngreb	Force		• •

5.2 Central Computer Strukturering

For Central Computeren er der udviklet struktur som illustreres ved figur 5.3, og figur 5.4. Delblokkene og interfaces imellem disse, som skitseret i IBDet figur 5.4 er yderligere beskrevet i blokbeskrivelsen som findes i tabel 5.1



Figur 5.3: Central Computer BDD



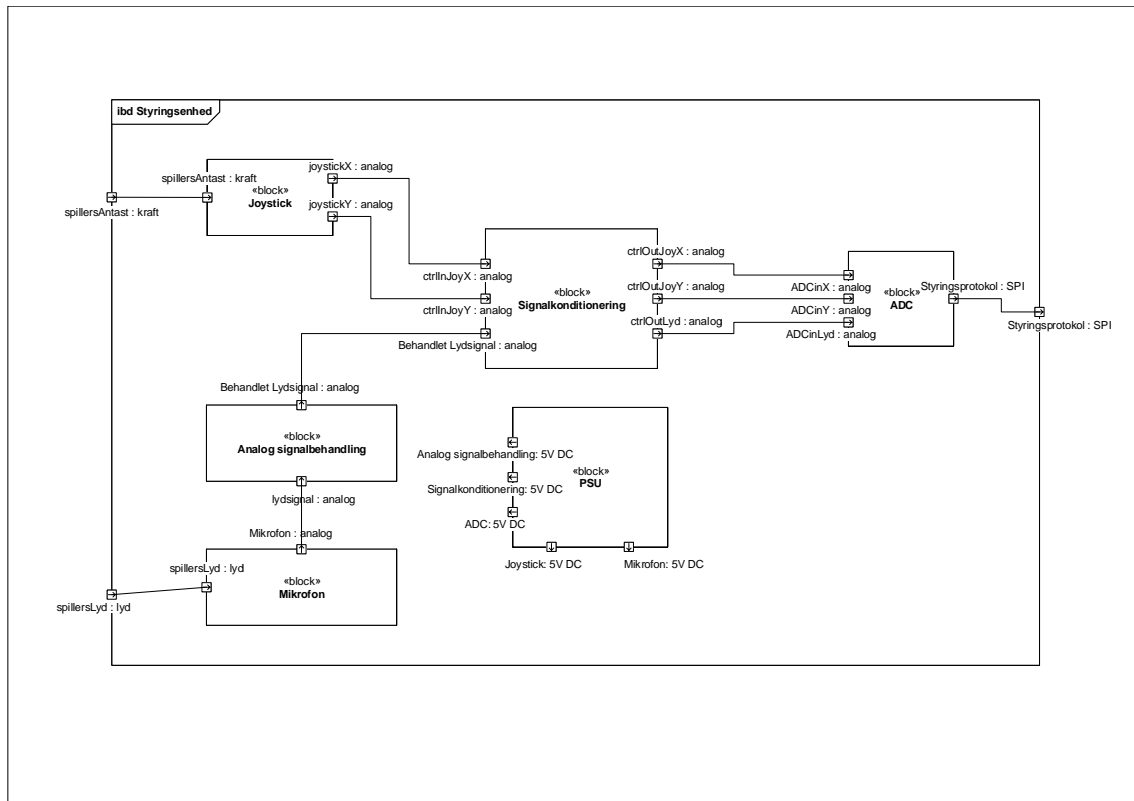
Figur 5.4: Central_Computer IBD

Tabel 5.2: Blokbeskrivelse for Central Computer

Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Embedded controller	Bearbejder og videregiver data fra Styringsenhed til Sumobot, trådløst. sender løbende information omkring spilstatus (re-sterende liv, resterende spilletid mv.) til Display blokken	In Styringsprotokol	SPI	Digitalt samplet signal af detvalgte styringsinput (lyd eller joystick) fra styringsen-hede	• •
		Out displayData	SPI	Løbende information omkring spilstatus (re-sterende liv, resterende spilletid mv.)	• •
		Inout SumoKom	Wifi	Tovejskommunikation med SumoBot's hvor der sendes information omkring retning og hastighed og modtage information om et evt. angreb	• •
Display	Viser løbende information omkring spilstatus (re-sterende liv, resterende spilletid mv.) til spillerne	Out Display	Lys	Displayets løbende information til spillerne omkring spilstatus (resterende liv, resterende spilletid mv.)	• •
		In displayData	SPI	Løbende information omkring spilstatus (re-sterendeliv, resterende spilletid mv.)	• •

5.3 Styringsenhed Strukturering

For Styringsenheden er der udviklet struktur som illustreres ved figur ??, og figur 5.5. Delblokkene og interfaces imellem disse, som skitseret i IBDet figur 5.5 er yderligere beskrevet i blokbeskrivelsen som findes i tabel 5.1



Figur 5.5: IBD for Styringsenhed

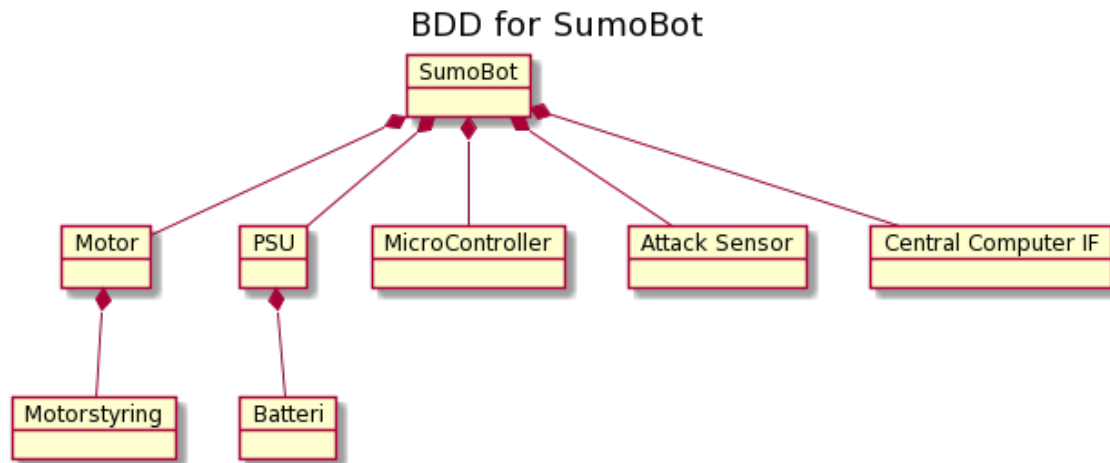
Tabel 5.3: Blokbeskrivelse for Styringsenhed [1/2]

Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Joystick	Styring af retning og hastighed på SumoBotsne ud fra påvirkning af joysticket	joystickX	Analog	Signal der afspejler x -koordinatet.	<ul style="list-style-type: none">• Udgangsimpedans: $0\ \Omega$ til $10\ \text{k}\Omega$• Spændingsområde: $0\ \text{V}$ til $5\ \text{V}$• Hvilespænding: $2.50\ \text{V}$
		joystickY	Analog	Signal der afspejler y -koordinatet.	<ul style="list-style-type: none">• Udgangsimpedans: $0\ \Omega$ til $10\ \text{k}\Omega$• Spændingsområde: $0\ \text{V}$ til $5\ \text{V}$• Hvilespænding: $2.50\ \text{V}$
Analog signal-behandling	Forstærker mikrofon inputtet i en grad at ADC kan konvertere de forskellige værdier. Gain: 250gg	Lydsignal	Analog	Ufiltreret og forstærket signal indeholdende lydinformation.	<ul style="list-style-type: none">• Indgangsimpedans: $20\ \text{k}\Omega$
		Behandlet lydsignal	Analog	Filtreret og forstærket signal indeholdende lydinformation.	<ul style="list-style-type: none">• Signalfrekvensområde: $300\ \text{Hz}$ til $3\ \text{kHz}$• Udgangsimpedans: ???
Mikrofon	Konvertering fra akustisk til elektrisk signal	Mikrofon	Analog		<ul style="list-style-type: none">• Udgangsimpedans: $2.2\ \text{k}\Omega$• Sensitivitet: $(8 \pm 2)\ \text{mV Pa}^{-1}$• SNR > 40dBV
		SpillersLyd	Akustisk		<ul style="list-style-type: none">• Dynamisk område: 70dB SPL

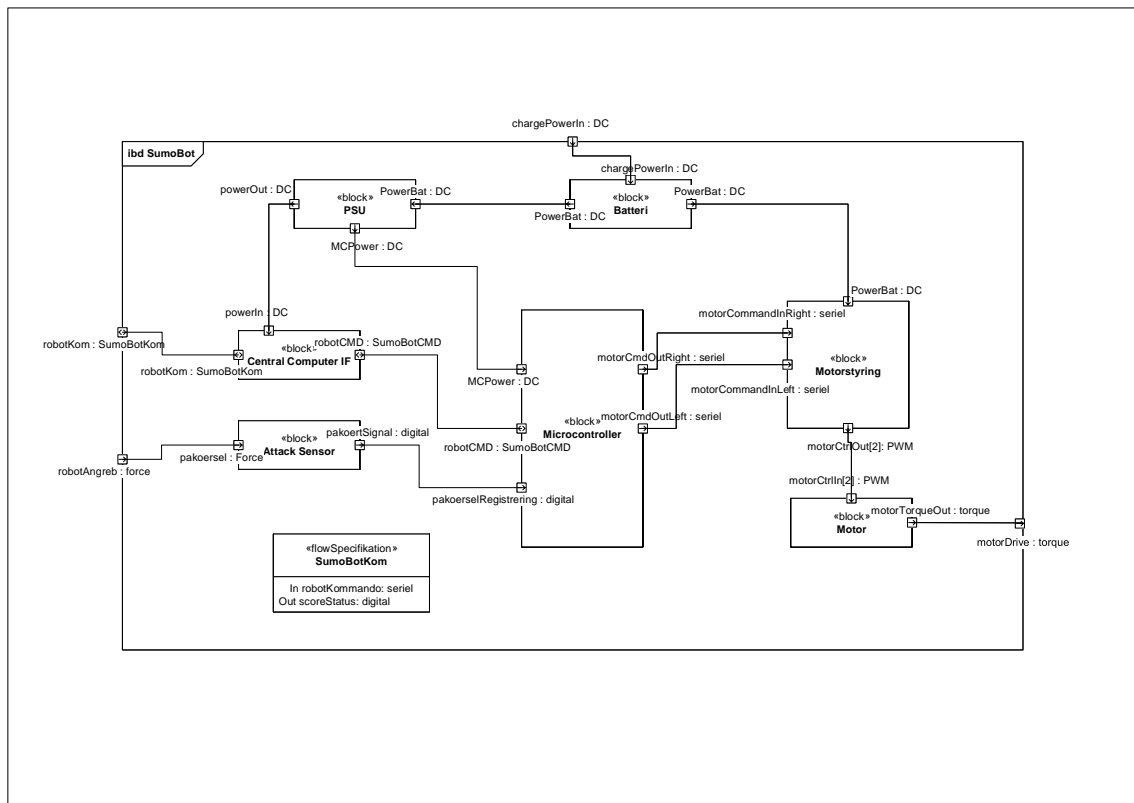
Tabel 5.4: Blokbeskrivelse for Styringsenhed [2/2]

Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Signal konditionering	Buffer stage der skal sikre maksimal spændingsoverførsel.	ctrlInJoyX	Analog		<ul style="list-style-type: none">Indgangsimpedans: 1 kΩ
		ctrlInJoyY	Analog		<ul style="list-style-type: none">Indgangsimpedans: 1 kΩ
		Behandlet lydsignal	Analog		<ul style="list-style-type: none">Indgangsimpedans: 10x udgangsim- pedans for Analog Signalbehandling blokken.
ADC	Analog til digital konvertering. Sample rate: 10 kHz. Bitopløsning: 10 bit SNR: 60 dB	ADCinLyd	Analog	Delta Sigma ADC	<ul style="list-style-type: none">Fuld skala input spændingsområde: 0 V til 5 V
		ADCinY	Analog	SAR Sequential ADC	<ul style="list-style-type: none">Fuld skala input spændingsområde: 0 V til 5 V
		ADCinX	Analog	SAR Sequential ADC	<ul style="list-style-type: none">Fuld skala input spændingsområde: 0 V til 5 V
		Styringsprotokol	SPI	Interface til Central Computer	<ul style="list-style-type: none">Overførselshastighed:

5.4 SumoBot Strukturering



Figur 5.6: SumoBot BDD



Figur 5.7: SumoBot IBD

Tabel 5.5: Blokbeskrivelse for Sumobot [1/2]

Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Central Computer IF	Mellemedet mellem Sumobot og Central Computeren. Sender og modtager data trådløst til og fra central computeren, og videregiver til Microcontroller.	robotKom	SumoBotKom	WiFi-signal fra Central Computer	•
		robotCMD	SumoBotCMD	Signal til Microkontroller	•
Attack Sensor	Registrere påkørsel mellem SumoBots. Kollisionsstatus sendes til Microcontroller.	pakoersel	Force	Fysisk påvirkning fra omverdenen	•
		pakoertSignal	Digital	Logisk I/O signal	• Spændingsområde: 0 V til 5 V
PSU	Forsyner de enkelte blokke med korrekt spændingsniveau. Kilden kommer fra batteri.	powerOut	DC	Forsyning til central Computer IF	• Spændingsforsyning 5V
		MCPower	DC	Forsyning til Microcontroller	• Spændingsforsyning 5V
		PowerBat	DC	Batteri forsyning (TBR)	• Spændingsniveau: 9V
Microcontroller	Styrer motorstyringen ud fra input fra Central Computer IF. Ligeledes har forbindelse med Attack Sensoren og sender data videre til Central Computeren gennem Central Computer IF.	robotCMD	SumoBotCMD	Signal fra Central Computer IF	• (TBD)
		pakoerselRegistrering	Digital	Logisk I/O signal fra Attack Sensor	• Spændingsområde: 0 V til 5
		motorCmdOutLeft	Signal	Signal til motorstyring om venstre motor bestående af 2 logiske signaler og 1 PWM signal	• logisk signal [2] med Spændingsområde: 0 V til 5 V • PWM signal med Spændingsområde: 0 V til 5 V
		motorCmdOutRight	Signal	Signal til motorstyring om højre motor bestående af 2 logiske signaler og 1 PWM signal	• logisk signal [2] (TBR) med Spændingsområde: 0 V til 5 V

Tabel 5.6: Blokbeskrivelse for Sumobot [2/2]

Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Batteri	Forsyner PSU samt motorerne gennem motorstyringen.	chargePowerIn	DC	Til batteri fra oplader	• Spændingsniveau: 9V
		PowerBat	DC	Fra batteri	• Spændingsniveau: 9V
Motor	Skaber fremdrift/bevægelse til Sumobot. Bliver kontrolleret gennem motorstyringen.	motorCtrlIn[2]	PWM	PWM fra motorstyring til motor	• PWM signal med Spændingsområde: 0 V til 9 V
		motorTorqueOut	torque	Energi fra motor til hjul	• Moment/energi fra motor til hjul (TBR)
Motorstyring	Styrer retning og hastighed af de 2 motorer ud fra kommandoer givet af Microcontrolleren.	motorCommandInRightSignal		Signal til motorstyring om venstre motor bestående af 2 logiske signaler og 1 PWM signal	• logisk signal [2] med Spændingsområde: 0 V til 5 V • PWM signal med Spændingsområde: 0 V til 5 V
		motorCommandInLeftSignal		Signal til motorstyring om højre motor bestående af 2 logiske signaler og 1 PWM signal	• logisk signal [2] med Spændingsområde: 0 V til 5 V • PWM signal med Spændingsområde: 0 V til 5 V
		motorCtrlOut[2]	PWM	PWM signal til motor	• PWM signal med Spændingsområde: 0 V til 9 V

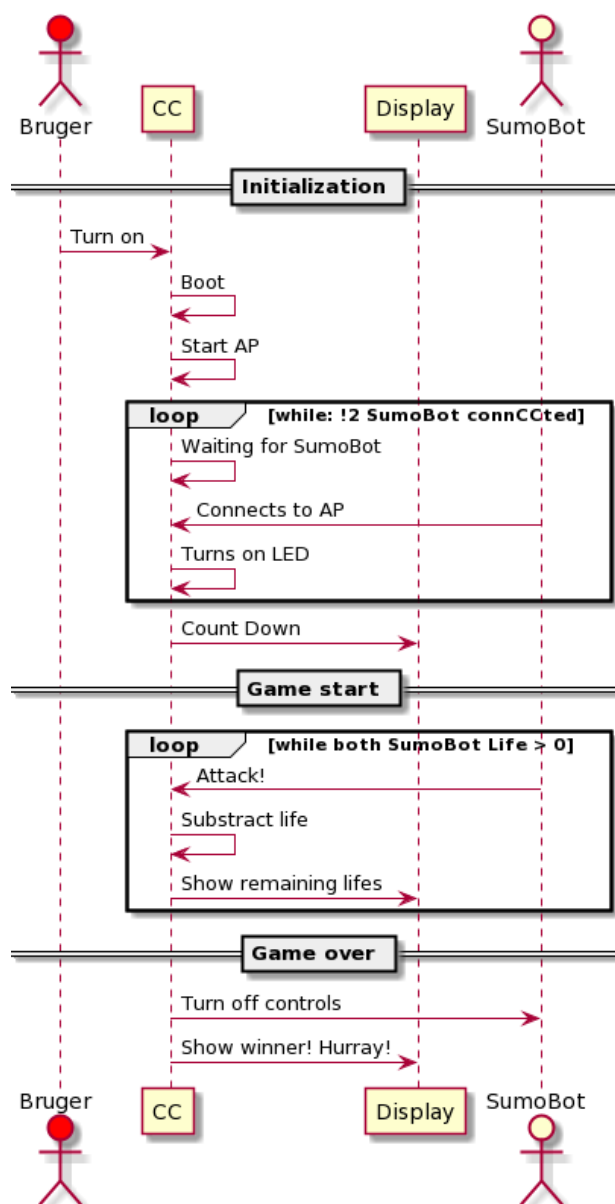
6. Analyse

6.1 Central Computer Analyse

Analyseafsnittet vil for de to blokke i central computer yderligere udpensle krav til selve blokke- ne, såvel som forbindelserne imellem dem. Da denne del af systemet er software præget, vil der især blive lagt vægt på hvilke krav der stilles til de teknologier som skal understøtte specifikke funktionaliteter.

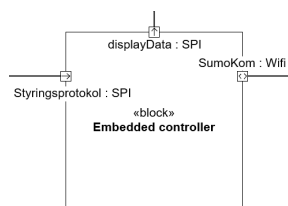
Den centrale computers opgave er at fungere som det digitale bindeled mellem styringen- hederne og SumoBots, således at den fra styringsenhederne modtager information som proces- seseres til retning og hastighed, som ultimativt kan blive videreformidlet til de to SumoBot's. Samtidig holder Central Computer styr på spillets status, dvs. antal liv, resterende spilletid mv.

Central Computers funktionalitet er overordnet skitseret i figur 6.1



Figur 6.1: Central Computer System Sekvensdiagram

6.1.1 Embedded Controller



Figur 6.2: Embedded Controller I/O

Tabel 6.1: Blokbeskrivelse for Central Computer: Embedded Controller

Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Embedded controller	Bearbejder og videregiver data fra Styringsenhed til Sumobot, trådløst. sender løbende information omkring spilstatus (resterende liv, resterende spilletid mv.) til Display blokken	In Styrings-protokol	SPI	Digitalt samlet signal af detvalgte styringsinput (lyd eller joystick) fra styringsenheden	•
		Out displayData	SPI	Løbende information omkring spilstatus (resterende liv, resterende spilletid mv.)	•
		Inout SumoKom	Wifi	Tovejskommunikation med SumoBot's hvor der sendes information omkring retning og hastighed og modtage information om et evt. angreb	•

Embedded Controller ud- og indgange ses i figur 6.2

Da afsnittet omfatter til den ene blok Embedded Controller, er dette afsnit yderligere inddelt ift. dens opgaver og interfaces. Kravene for de forskellige underafsnit skal ses som en samlet mængde krav til valg af Embedded Controller.

Interface mod styringsenheden

Interfacet til Styringsenheden modtager signalet Styringsprotokol som overføres jf. SPI-protokollen. Signalet er ikke processeret på dette tidspunkt og vil blot være en bitstrøm af samplede amplituder fra styringsenhedens lyd- eller controllerinput. Disse skal senere processeres til at blive en retning og en hastighed for en SumoBot. Controllerinputtet analyseres udelukkende ift. amplitude mens lydinputtet analyseres ift. amplituder indenfor specifikke frekvensspektre, som ultimativt vil give en retning og en hastighed.

Krav

- Skal have indgange til minimum 2 SPI-forbindelser. En til hver Styringsenhed.
- SPI-indgangen skal kunne modtage data i en hastighed som stemmer overens med samplingsfrekvensen på >10kHz og bitopløsningen på 10 bit fastsat i afsnittet 6.2 (TBR). Det vil sige med denne sammenhæng:

$$F_s \cdot \text{bitopløsning} < \text{hastighed} \left[\frac{\text{kbit}}{\text{s}} \right]$$

$$\text{hastighed} > 100.000 \left[\frac{\text{bit}}{\text{s}} \right]$$

Controllerbehandling

Lydbehandling Lyden som opfanges af mikrofonerne og sendes digitaliseret til Central Computer fra Styringsenhederne. Disse skal herefter processeres først vha. frekvensanalyse (diskret Fourier-transformation, herefter DFT). Den algoritme som danner baggrund for DFT kræver meget processorkraft, som, særligt når behandlingen skal foretage så mange på hinanden følgende gange med kort tidsinterval. DFT kommer ikke til at stå alene, da der efterfølgende vil være endnu en algoritme som omregner frekvensspektret til en retning og hastighed.

Krav

- Processoren skal kunne arbejde med en høj clock-hastighed og gerne ved mange bits, således at det, samtidig med de resterende ting som foregår på processoren, er muligt at have to DFT som med kort tidsinterval foregår samtidig.

Joystikbehandling Behandlingen af inputtet fra joystikket er meget mere simpelt, da styringen igennem dette baserer sig på amplituden på signalet, som modtages fra styringsenheden digitaliseret og som blot vha. en simpel algoritme omregnes til retning og hastighed.

Spilstyring

Spilstyringen kan se som en state-machine som styrer resterende tid og liv for de to SumoBots. Spilstyringen skal programmeres i et sprog som egner sig hertil, hvortil der stilles krav om at det er overskueligt og gerne med et højt abstraktionsniveau, således at det er let at programmere og senere udbygge hvis der på et tidspunkt ønskes flere spilstilstande. Det skal igennem sproget være let at manipulere GPIO's, således at der spillets status let kan formidles til displayet.

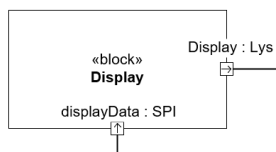
Interface mod SumoBots (Trådløs kommunikation)

Interfacet med SumoBots er en trådløs forbindelse som hostes af Central Computer og som SumoBots under opstart kobler sig på. Informationen som formidles til SumoBots bliver efter en simpel protokol som sendes med korte tidsintervaller ($\approx 50\text{ms}$).

Krav

- Embedded Controller skal kunne lave et wifi-access point
- Wifi-forbindelsen skal række min. 4 meter
- Wifi-forbindelsen skal være stabil

6.1.2 Display



Figur 6.3: Display I/O

Displayet viser løbende information omkring spilstatus (resterende liv, resterende spilletid mv.) til spillerne, se figur 6.3

Tabel 6.2: Blokbetegnelse for Central Computer: Display

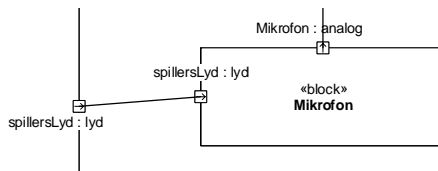
Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Display	Viser løbende information omkring spilstatus (resterende liv, resterende spilletid mv.) til spillerne	Out Display	Lys	Displayets løbende information til spillerne omkring spilstatus (resterende liv, resterende spilletid mv.)	• •
		In display-Data	SPI	Løbende information omkring spilstatus (resterende liv, resterende spilletid mv.)	• •

6.2 Styringsenhed Analyse

6.2.1 Analyse af lydgivere

Det antages at frekvensområdet ligger i området 300 Hz til 3.000×10^3 Hz[2]. At designe et vel-fungerende filter i dette frekvensområde burde være muligt uden de store armbevægelser.

6.2.2 Mikrofon



Figur 6.4: Udsnit af mikrofonblokken

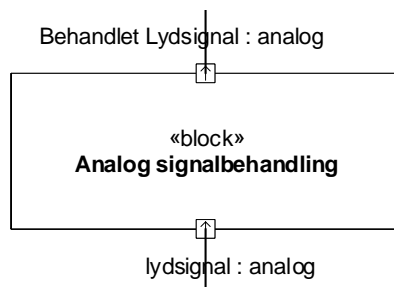
Tabel 6.3: Blokbeskrivelse for Styringsenhed: Mikrofon

Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Mikrofon	Konvertering fra akustisk til elektrisk signal	Mikrofon	Analog		<ul style="list-style-type: none"> • Udgangsimpedans: 2.2 kΩ • Sensitivitet: $(8 \pm 2) \text{ mV Pa}^{-1}$ • SNR > 40dBV
		SpillersLyd	Akustisk		<ul style="list-style-type: none"> • Dynamisk område: 70dB SPL

Der er blevet foretaget et valg om at benytte en elektretmikrofon (**TBR**)(link til mikrofon) som allerede var til rådighed. Mest kritisk for mikrofonen er SNR[3]. Hvis denne ikke viser sig tilstrækkelig høj til at kunne differentiere baggrundsstøj fra det reelle input vil der skulle laves akustisk regulering.

Outputtet fra mikrofonen behandles og konditioneres, hvorfor kravene til Analog signalbehandling og Signalkonditionering blokkene defineres ud fra mikrofonens output.

6.2.3 Analog signalbehandling



Figur 6.5: Udsnit af blokken til analog signalbehandling

Denne blok skal filtrere frekvenser udenfor pasbåndet fra samt forstærke signalet fra mikrofonen op, så det kan registreres af en ADC.

Det er nødvendigt at have et filter der uden problemer kan lukke frekvenser i området 300 Hz til 3.000×10^3 Hz igennem, uden at det opstår bemærkelsesværdi forvrængning eller faseproblemer, derudover bør

Tabel 6.4: Blokbeskrivelse for Styringsenhed: Analog signalbehandling

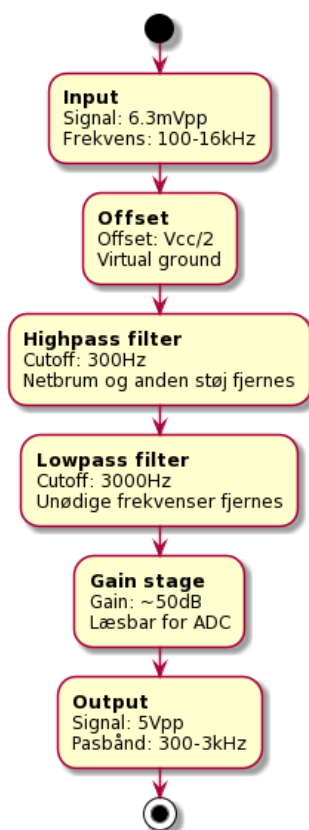
Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Analog signal-behandling	Forstærker mikrofon inputtet i en grad at ADC kan konvertere de forskellige værdier. Gain: 250gg	Lydsignal	Analog	Ufiltreret og forstærket signal indeholdende lydinformation.	<ul style="list-style-type: none"> Indgangsimpedans: 20 kΩ
		Behandlet lydsignal	Analog	Filtreret og forstærket signal indeholdende lydinformation.	<ul style="list-style-type: none"> Signalfrekvensområde: 300 Hz til 3.00 kHz Udgangsimpedans: ???

uønskede frekvenser bortfiltreres i en grad af de ikke opfanges af ADC'en. Med et pasbånd beliggende i området 300 Hz til 3.000×10^3 Hz og en dertilhørende forstærkning på ≈ 300 , vil et passende frekvensrespons være et båndpasfilter der kan beskrives :

$$T(s) = T_{HP} \cdot T_{LP} \cdot T_{Gain} \quad (6.1)$$

hvor følgende gør sig gældende for de forskellige led

$$\begin{aligned} T_{HP1} &= f_{3db} = 300 \text{ Hz} \quad 20 \text{ dB/dec} \\ T_{LP1} &= f_{3db} = 3.000 \times 10^3 \text{ Hz} \quad 40 \text{ dB/dec} \\ T_{Gain} &= \text{Gain} \approx 300 \end{aligned} \quad (6.2)$$



Figur 6.6: Flow der viser funktionaliteten af filteret

Der ses bort fra -3 dB dæmpningen ved de to knækfrekvenser, da denne på nuværende tidspunkt ikke vurderes at have en større betydning — dette korrigeres hvis det volder problemer ifb. ADC-konverteringen.

Ved at lade lavpasledet dæmpe med 40 dB/dec ved 3 kHz, bør aliaseringsproblemer kunne undgås når signal sendes videre til ADC'en.

For at kunne aflæse signalet korrekt på ADC'en er signalet nødt til at fluktuere omkring et fast punkt, hvor der her vælges $\frac{V_{CC}}{2}$, hvorved et med der svinger m. 0 til VCC. Ændring af amplitude samt evt. ekstra *offsetbiasing* bør finde sted i signalkonditioneringen.

For at fastsætte en endelig gain er en reel test af mikrofonen nødvendig, dog antages det at

$$\text{Mikrofon spændingsoutput} \approx 8 \text{ mV (TBR)} \quad (6.3)$$

Her af

$$\frac{\text{Maksimal spænding}}{\text{Mikrofon output}} = \text{Gain} \quad (6.4)$$

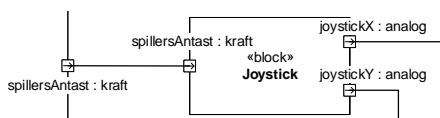
Kan niveauet fastsættes ud fra en antagelse om en outputspænding fra mikrofonen på $(8 \pm 2) \text{ mV}$

$$\begin{aligned} \text{Gain} &\approx 300 \\ 300 \cdot 8 \text{ mV} &= 2.4 \text{ V} \end{aligned} \quad (6.5)$$

Ved nærmere analyse af den valgte mikrofon egenskaber, kan gainniveauet justeres efter behov — dette bør kunne udføres uden nogle vanskeligheder i et senere forløb.

Outputimpedansen for signalbehandlingsmodul har ikke den store betydning, men bør holdes i omegnen af 1 kΩ til 10 kΩ for ikke at skabe problemer med videre interfacing af moduler.

6.2.4 Joystick



Figur 6.7: Udsnit af joystickblokken

Tabel 6.5: Blokbeskrivelse for Styringsenhed: Joystick

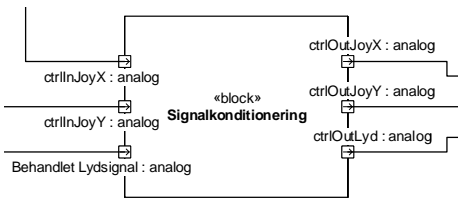
Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Joystick	Styring af retning og hastighed på SumoBotsne ud fra påvirkning af joysticket	joystickX	Analog	Signal der afspejler x-koordinatet.	<ul style="list-style-type: none"> Udgangsimpedans: 0 Ω til 10 kΩ Spændingsområde: 0 V til 5 V Hvilespænding: 2.50 V
		joystickY	Analog	Signal der afspejler y-koordinatet.	<ul style="list-style-type: none"> Udgangsimpedans: 0 Ω til 10 kΩ Spændingsområde: 0 V til 5 V Hvilespænding: 2.50 V

Denne enhed har til formål at udsende en spænding, der kan bearbejdes som styringsinput via en ADC. (TBD)

6.2.5 Signalkonditionering

6.2.6 ADC

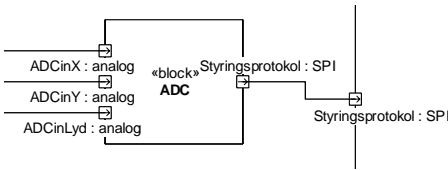
Pr. styringsenhed er der én ADC-blok som konverterer inputs fra hhv. joystickX, joystickY og mikrofon input til bitstrømme. Disse bitstrømme processeres i en microprocessor hvor der kører en realtids-FFT algoritme for at kunne generere en veldefineret styringsprotokol for SumoBotsne.



Figur 6.8: Udsnit af blokken til signalkonditionering

Tabel 6.6: Blokbeskrivelse for Styringsenhed: Signal konditionering

Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Signal konditionering	Buffer stage der skal sikre maksimal spændingsoverførsel.	ctrlInJoyX	Analog		• Indgangsimpedans: 1 kΩ
		ctrlInJoyY	Analog		• Indgangsimpedans: 1 kΩ
		Behandlet lydsignal	Analog		• Indgangsimpedans: 10x udgangsimpedans for Analog Signalbehandling blokken.



Figur 6.9: Udsnit af ADC-blokken

For at gøre dette muligt er der udarbejdet følgende krav til ADC modulet:

- Samplingrate: 10 kHz
- Bitopløsning: 10 bit
- Fuld-skala input range: 5V
- 1x Delta Sigma Converter
- SNR: 60 dB
- Indgange: 3
- Høj indgangsimpedans

Fuld skala input range på ADC’ens indgang skal stemme overens med inputsignalets spændingsområde. Derfor fastsættes at fuld-skala input range = 5V.

(TBR)Databladet for elektret mikrofonen som der anvendes foreskriver at SNR er mindst 60 dB. Hvis dette bruges som udgangspunkt til ADC, kan man ud fra denne antagelse om SNR og fuld skala input bestemme en mindste bitopløsning[4], som ADC’en skal outputte med:

$$60 = 6.02 * n + 1.76$$

$$n = 9.67$$

Tabel 6.7: Blokbeskrivelse for Styringsenhed: ADC

Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
ADC	Analog til digital konvertering. Sample rate: 10 kHz. Bitopløsning: 10 bit SNR: 60 dB	ADCinLyd	Analog	Delta Sigma ADC	<ul style="list-style-type: none"> Fuld skala input spændingsområde: 0 V til 5 V Indgangsimpedans: høj
		ADCinY	Analog	SAR Sequential ADC	<ul style="list-style-type: none"> Fuld skala input spændingsområde: 0 V til 5 V Indgangsimpedans: høj
		ADCinX	Analog	SAR Sequential ADC	<ul style="list-style-type: none"> Fuld skala input spændingsområde: 0 V til 5 V Indgangsimpedans: høj
		Styringsprotokol I2C		Interface til Central Computer	<ul style="list-style-type: none"> Overførselshastighed:

Tabel 6.8: Vurdering af ADC

	Delta Sigma	3 indgange	10 bit opløsning	5V fuld skala input range	10 kHz sample rate
Arduino Mega2560		x		x	x
ADC Pi Zero		x	x		
PSoC 5LP	x	x	x	x	x

Der skal altså anvendes en ADC med en effektiv bitopløsning på 10 bit. Det vil sige, der skal i designet af ADC'en tages hensyn til LSB fejl.

Til lydinputtet fra mikrofonforstærkeren bestemmes det, at der skal anvendes en Delta Sigma ADC. Dette er bestemt med følgende grundlag:

- Forhåndsforsikring for, at ADC'en kan yde den tilstrækkelige bitopløsning som inputs til en FFT algoritme.
- Delta Sigma ADC opererer kontinuerligt(**TBR**reference til Texas Instruments) (integrerer over tid) og er derfor bedst til realtids-applikationer.

For at opfylde kravene til blokken ADC er der gjort overvejelser, som sammen med en vurdering ud fra kravene er opstillet i tabel 6.8. Da PSoC 5LP opfylder alle krav, vælges denne.

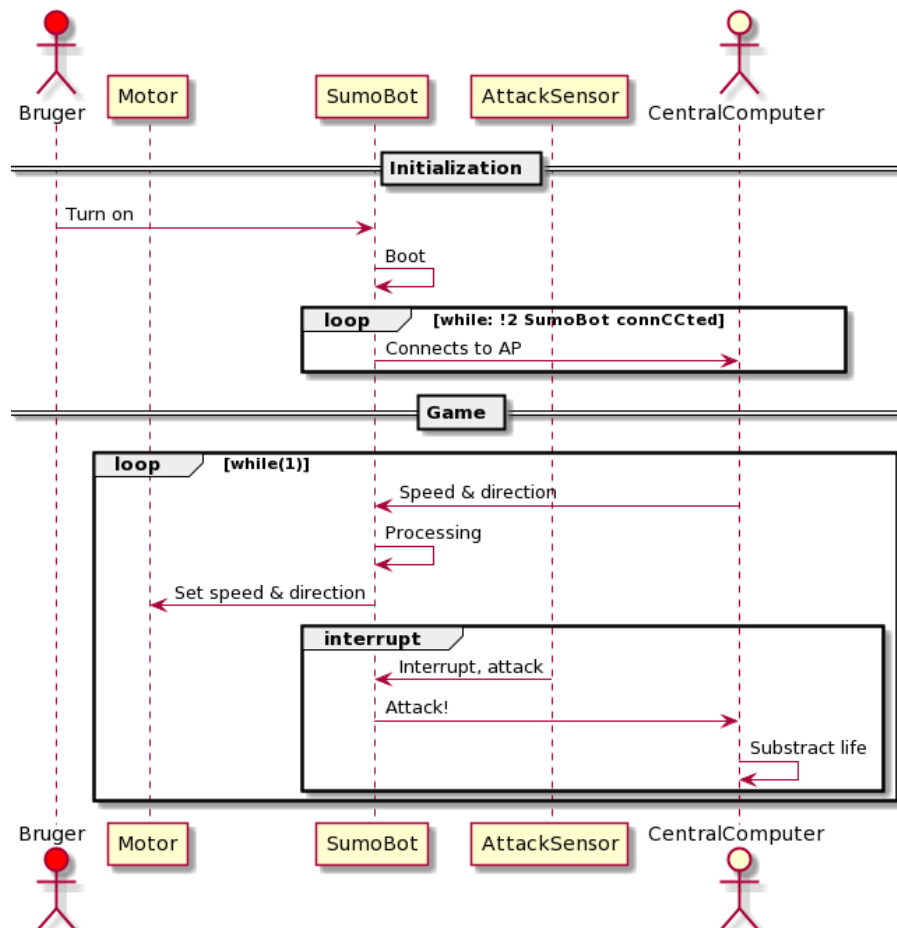
6.2.7 Digital signal processing

For ikke selv at skulle designe ADC'er analogt tages udgangspunkt i en platform hvor der kan køres noget firmware med en FFT-algoritme til lydsignalet fra mikrofonen. Her er til PSoC 5LP udarbejdet et bibliotek til FFT(**TBR**), som der i første omgang tages udgangspunkt i. Denne kan lave en 64 punkters FFT hvilket vurderes tilstrækkeligt i denne applikation med en konservativ signalbåndbredde. Derved kan signalprocesseringen også foregå i selvsamme blok.

6.3 Sumobot Analyse

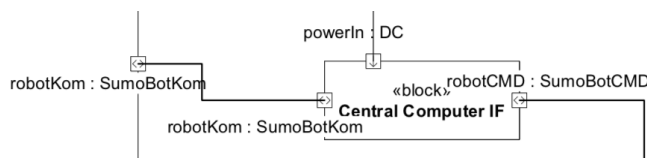
I dette analyseafsnit vil der blive beskrevet ansvaret for de enkelte delblokke for SumoBot. Foruden en beskrivelse af blokkene, vil forbindelserne mellem dem såvel blive udfoldet. Afsluttende vil der blive stillet en række krav for selve blokkene, der bliver grundlag for valg af komponenter senere i projektet.

SumoBots funktionalitet er overordnet skitseret i figur 6.10



Figur 6.10: SumoBot System Sekvensdiagram

Central Computer IF



Figur 6.11: Central Computer IF IBD

Dette er kommunikationsledet mellem SumoBot og Central Computeren. Det er denne blok der sørger for at fortælle SumoBotten omkring retning og fart, der bliver sendt digitaliseret fra Central Computeren. Blokken skal ligeledes sørge for at sende information tilbage til Central Computeren når en attack sensor er blevet aktiveret.

Krav

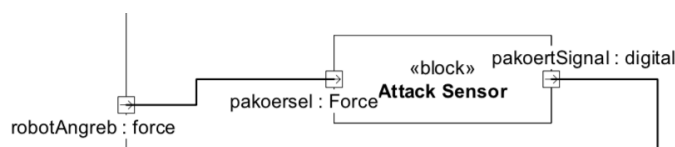
- Skal kunne kommunikere over WiFi.

Tabel 6.9: Blokbeskrivelse for Central Computer IF

Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Central Computer IF	Mellemløst mellem SumoBot og Central Computeren.	robotKom	SumoBotKom	WiFi-signal fra Central Computer	•
	Sender og modtager data trådløst til og fra central computeren, og videregiver til Microcontroller.	robotCMD	SumoBotCMD	Signal til Microcontroller	•

- Skal kunne køre 5V logik.
- Skal fylde mindre en "7x4x4"cm.
- Skal minimum have 4 In/Out porte til kommunikation med Microcontroller.

Attack Sensor



Figur 6.12: Attack Sensor IBD

Tabel 6.10: Blokbeskrivelse for Attack Sensor

Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Attack Sensor	Registrere påkørsel mellem SumoBots. Kollisionsstatus sendes til Microcontroller.	pakoersel	Force	Fysisk påvirkning fra omverdenen	•
		pakoertSignal	Digital	Logisk I/O signal	• Spændingsområde: 0 V til 5 V

Attack Sensor er den del af SumoBotten der registrerer at den er blevet påkørt af en anden SumoBot. 2 switches monteres på bagsiden af hver SumoBot og registrere kollision.

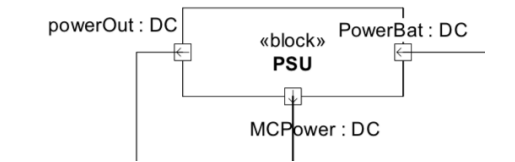
Krav

- Skal kunne holde til påkørsel fra alle tænkelige retninger.
- Skal kunne håndtere 5V med mellem 0 og 20 mA.
- Skal kunne holde til minimum 1000 tryk.

PSU

Delblokken PSU er strømforsyningen, der skal levere et stabilt 5V DC signal til de interne komponenter. Det er vigtigt at den kan levere et stabilt signal så komponenterne ikke slukker eller genstarter pga. for høj eller lav spænding. Nedenstående krav er specificeret ved 25 grader celsius.

Krav



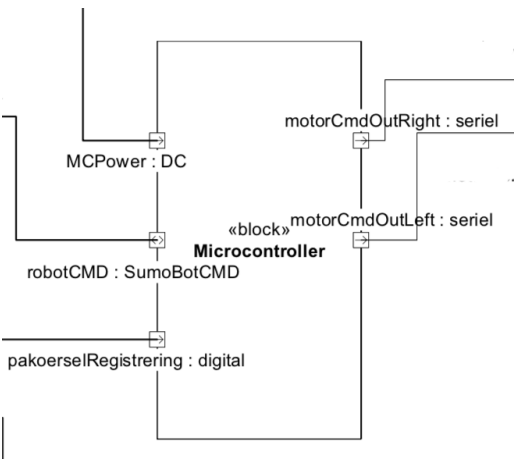
Figur 6.13: PSU IBD

Tabel 6.11: Blokbeskrivelse for PSU

Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
PSU	Forsyner de enkelte blokke med korrekt spændingsniveau. Kilden kommer fra batteri.	powerOut	DC	Forsyning til central Computer IF	• Spændingsforsyning 5V
		MCPower	DC	Forsyning til Microcontroller	• Spændingsforsyning 5V
		PowerBat	DC	Batteri forsyning (TBR)	• Spændingsniveau: 9V

- Skal kunne levere en stabil 5V DC forsyning med maks afvigelse på $4.85 < 5.00 < 5.15$ V.
- Skal kunne lervare minimum 500 mA.
- Skal kunne køre stabilt med et 7 til 12V DC input.
- Må ikke fylde mere end 4x4x4 cm
- Må maks have en output noice voltage på 120 mikroVolt

Microcontroller



Figur 6.14: Microcontroller IBD

Microkontrolleren er den del af SumoBotten der styrer motorstyringen ud fra indput fra Central Computer IF. Det er også denne der er forbundet med Attack Sensoren og sender dataen videre til Central Computer IF. Af den grund skal mircocontrolleren være hurtig nok til at bearbejde dataen og modtage/sende det videre, samt styre motorstyring.

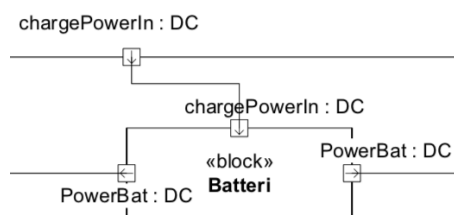
Krav

Tabel 6.12: Blokbeskrivelse for Microcontroller]

Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Microcontroller	Bærer motorstyringen ud fra input fra Central Computer IF. Ligeledes har forbindelse med Attack Sensoren og sender data videre til Central Computeren gennem Central Computer IF.	robotCMD	SumoBotCMD	Signal fra Central Computer IF	• (TBD)
		pakoerselRegister	Digital	Logisk I/O signal fra Attack Sensor	• Spændingsområde: 0 V til 5 V
		motorCmdOutSignal	Signal	Signal til motorstyring om venstre motor bestående af 2 logiske signaler og 1 PWM signal	• logisk signal [2] med Spændingsområde: 0 V til 5 V • PWM signal med Spændingsområde: 0 V til 5 V
		motorCmdOutSignal	Signal	Signal til motorstyring om højre motor bestående af 2 logiske signaler og 1 PWM signal	• logisk signal [2] (TBR) med Spændingsområde: 0 V til 5 V • PWM signal med Spændingsområde: 0 V til 5 V

- Skal kunne kommunikere serielt.
- Skal have minimum 10 In/Out ben (TBR).
- Skal kunne køre 5V DC.
- Må Maks være 12x4x4 cm.

Batteri



Figur 6.15: Batteri IBD

I og med at Sumobot er trådløs, er det essentielt at have et batteri monteret, hvilket er det blokken *batteri* står for. Men da robotten har en fysisk begrænsning, må batteriet ikke være for stort. Derudover skal batteriet levere et hvis spændingsniveau, som kan forsyne de forskellige delmoduler.

Krav

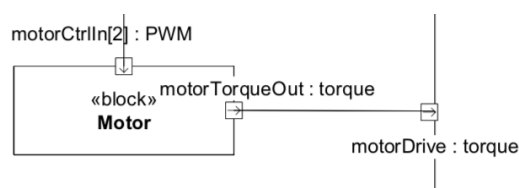
- Må maks være 10x10x5 cm. (TBR)
- Skal kunne levere 3A.
- skal have minimum 7000 mAh. (TBR)

Tabel 6.13: Blokbeskrivelse for Batteri]

Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Batteri	Forsyner PSU samt motorerne gennem motorstyringen.	chargePowerIn	DC	Til batteri fra oplader	• Spændingsniveau: 9V
		PowerBat	DC	Fra batteri	• Spændingsniveau: 9V

- Skal være et lithium batteri.
- 9 V spænding(**TBR**)

Motor



Figur 6.16: Motor IBD

Tabel 6.14: Blokbeskrivelse for Motor]

Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Motor	Skaber fremdrift/bewægelse til Sumobot. Bliver kontrolleret gennem motorstyringen.	motorCtrlIn[2]	PWM	PWM fra motorstyring til motor	• PWM signal med Spændingsområde: 0 V til 9 V
		motorTorqueOut	torque	Energi fra motor til hjul	• Moment/energi fra motor til hjul (TBR)

Blokken *motor* repræsenterer selve motoren. Dens opgave er at flytte SumoBot rundt på spillepladen. I valg af motor sætter de fysiske dimensioner (20x20x20) af SumoBot atter en begrænsning. Et andet aspekt der er værd at tage i betragtning er momentet i valget af motor. Et andet aspekt i valg af motor motor, er at hver bil består af 2 motorer. Der skal altså i alt bruges 4 styk til projektet. (**TBD**)

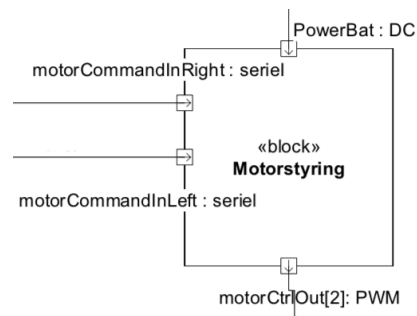
Krav

- Skal være en DC motor.
 - Må Maks være 8x6x6 cm.
 - Skal kunne afgive en moment på 1 NM fra 0 RPM.(**TBR**)
- (**TBR**)

Motorstyring

Motorstyringen tager sig af styring af de 2 fastmonterede motorer på SumoBotten. Der skal være mulighed for at køre de 2 motorer begge retninger med forskellige hastigheder uafhængige af hinanden. Den skal ligeledes kunne holde til det "load" motorene påtrykkes.

Krav



Figur 6.17: Attack Sensor IBD

Tabel 6.15: Blokbeskrivelse for Motorstyring

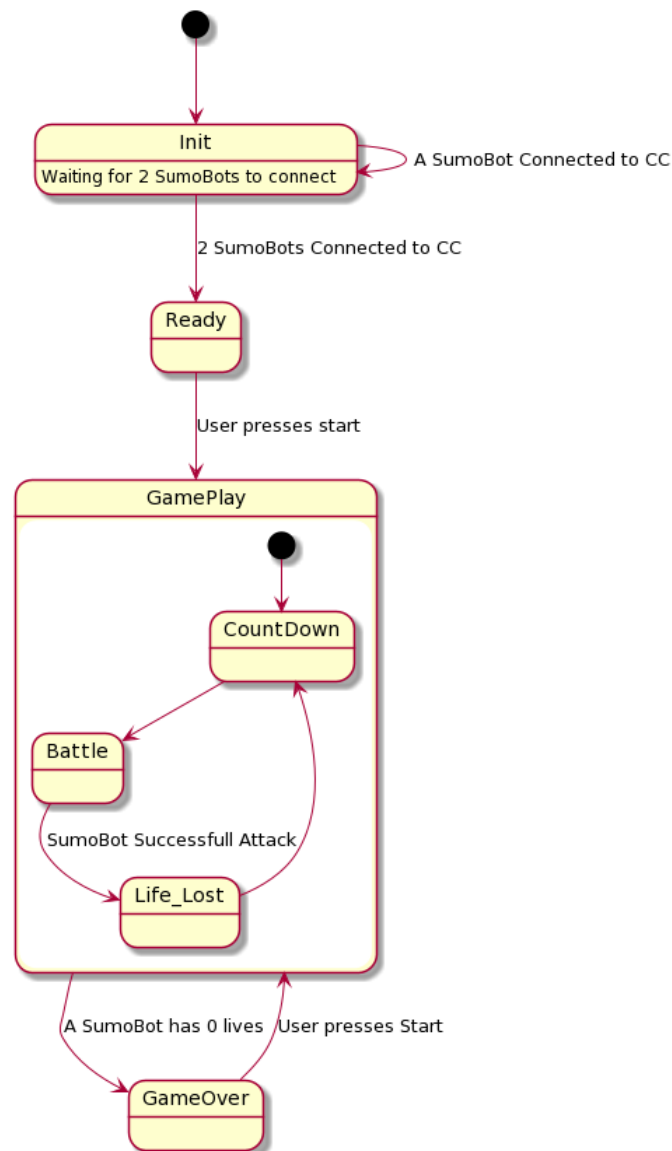
Bloknavn	Funktion	Portnavn	Type	Beskrivelse	Værdier
Motorstyring	Styrer retning og hastighed af de 2 motorer ud fra kommandoer givet af Microcontrolleren.	motorCommandInRight	Signal	Signal til motorstyring om venstre motor bestående af 2 logiske signaler og 1 PWM signal	<ul style="list-style-type: none">logisk signal [2] med Spændingsområde: 0 V til 5 VPWM signal med Spændingsområde: 0 V til 5 V
		motorCommandInLeft	Signal	Signal til motorstyring om højre motor bestående af 2 logiske signaler og 1 PWM signal	<ul style="list-style-type: none">logisk signal [2] med Spændingsområde: 0 V til 5 VPWM signal med Spændingsområde: 0 V til 5 V
		motorCtrlOut[2]	PWM	PWM signal til motor	<ul style="list-style-type: none">PWM signal med Spændingsområde: 0 V til 9 V

- Skal kunne anvende 5V logik.
- skal kunne styre retning og hastighed på 2 motore uafhængigt af hinanden.
- Skal kunne styre hastighed ved hjælp af PWM.
- Skal kunne håndtere minimum 2A load kontinuerligt ved minimum 12V DC.
- Skal kunne klare 12V DC indgang / udgang.
- Må ikke være større end 5x5x5 cm

7. Design

7.1 Central Computer Design

7.1.1 Overordnet



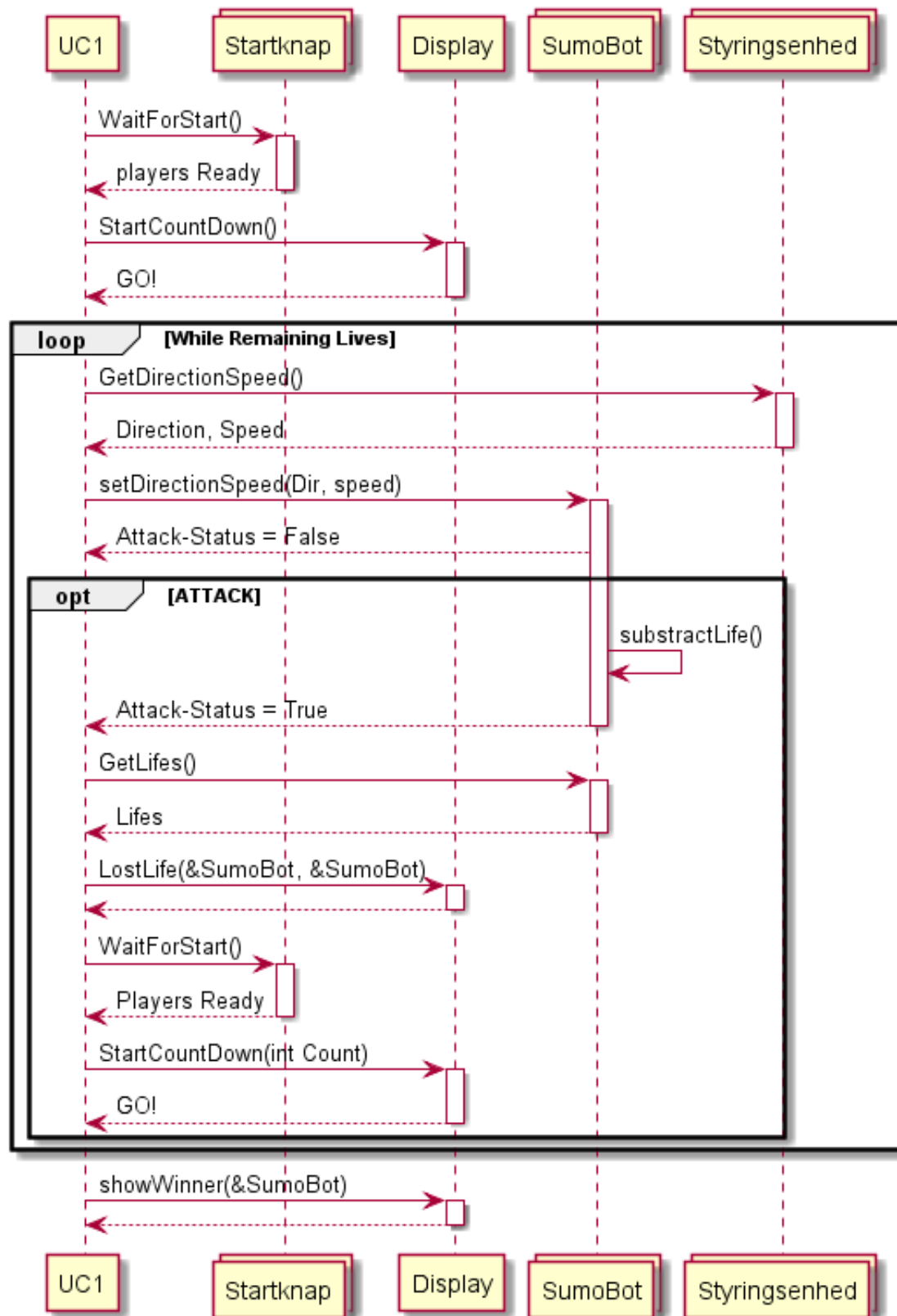
Figur 7.1: Statemachine over Central Computer

7.1.2 Initialisering

Init sekvensen handler meget om hvordan de fysiske enheder taler sammen, derfor er den med aktører.

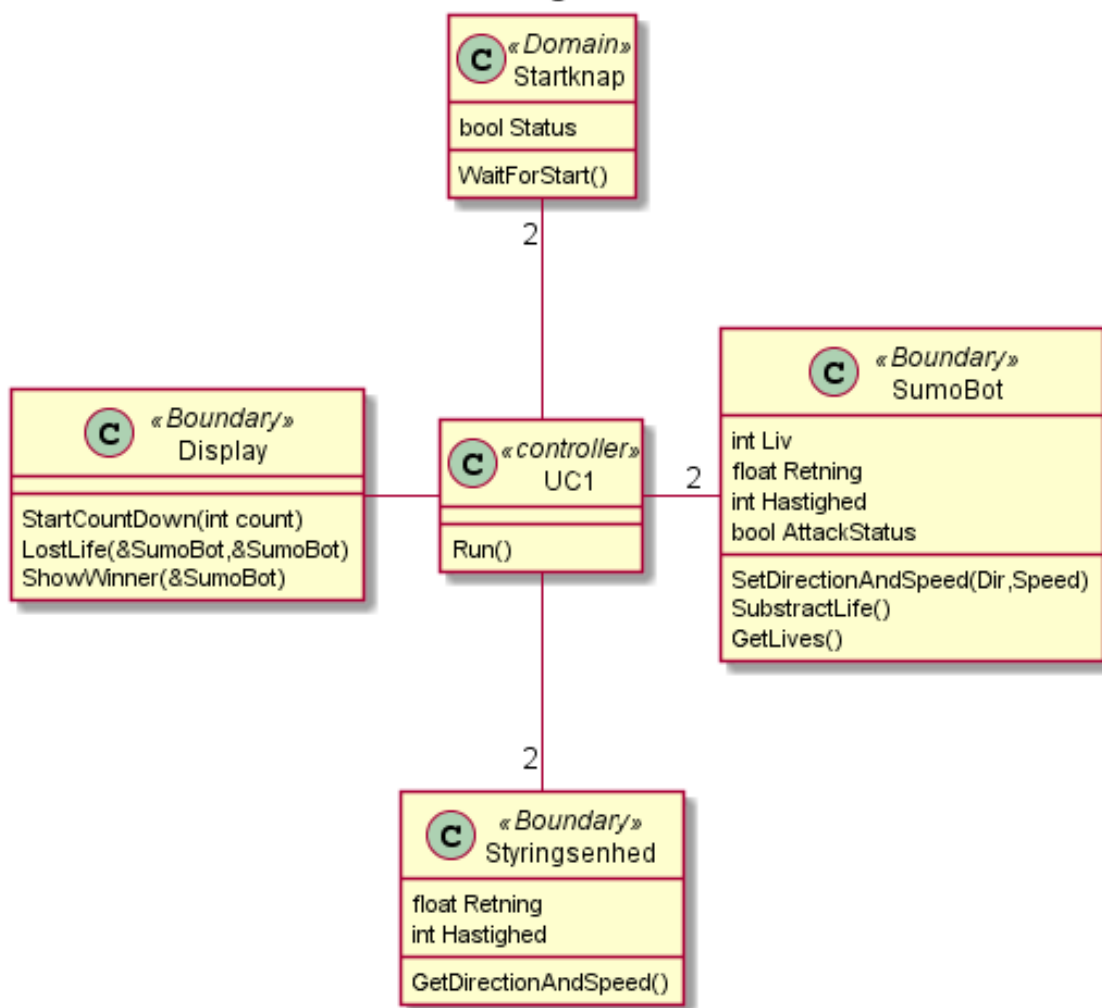
startCountDown(int count)

Description Funktionen varetager nedtællingen efter begge spillere har meldt klar
Parameters count: Antallet af sekunder som tælles ned
Returnvalue Ingen



Figur 7.2: Sekvensdiagram over UC1+2

CC Class diagram for UC1



Figur 7.3: Klassesdiagram

7.2 Styrsenhed Design

7.2.1 ADC

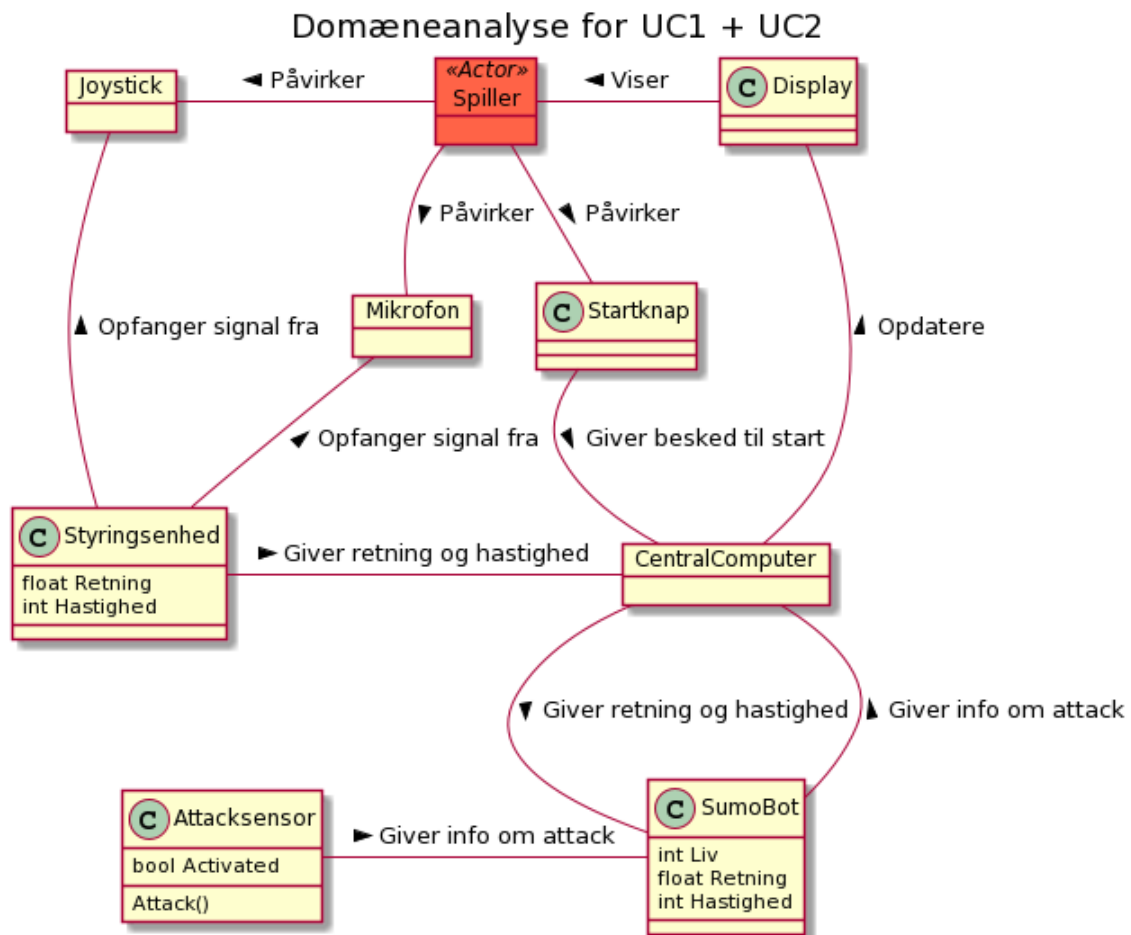
Da signalprocesseringen også kan foregå på PSoC 5LP er denne platform et oplagt valg til også at designe styrsenhedens ADC'er i. Herved kan analog til digital konvertering og processering integreres i ét modul. Der ses i figur(TBR)?? et flowdiagram over den tiltænkte virkemåde for ADC og DSP i styrsenheden.

Til designet af denne enhed er der som inspiration brugt et bredt udvalg af applikationsnoter fra Cypress Semiconductor som har udviklet PSoC 5LP. Den pågældende er af familien CY8C58xx. Opsætningen og implementeringen af ADC og DSP blokken er gjort i PSoC Creator 4.3.

Delta Sigma ADC komponenten på PSoC kan enten konfigureres ved firmware eller ved et grafisk interface i PSoC Creator 4.3. Her er det grafiske interface brugt, og i det følgende vil designovervejelserne dokumenteres.

Analog interfaces

For at få bedst mulig interface til mikrofonforstærker og filter er der på baggrund af applikationsnoten AN58304, s. 3 (TBR)REFERENCE ... valgt port [P0:7] som indgang til ADC'en, da denne er en af de foreskrevne analoge porte med bedst performance.



Figur 7.4: Domænemodel

Input range

For at få så meget headroom og derved opløsning som muligt, konfigureres ADC'en til (i princip) rail to rail operation. Men ved denne konfiguration i PSoC aktiveres en buffer på indgangen for at give mulighed for at forstærke indgangssignalet. Dette fravælges (bufferen bypasses), for at opnå størst muligt headroom og derved opløsning, se Figur 7.6. Hertil opnås den højst mulige indgangsimpedans, da anvendelsen af denne buffer er på bekostning af indgangsimpedansen. (TBR)REFERENCE TIL DELTA SIGMA ADC, s. 12. Hvis der herved opstår problemer med overstyring af ADC'en, kan der konfigureres til rail-to-rail med buffer gain på 1.

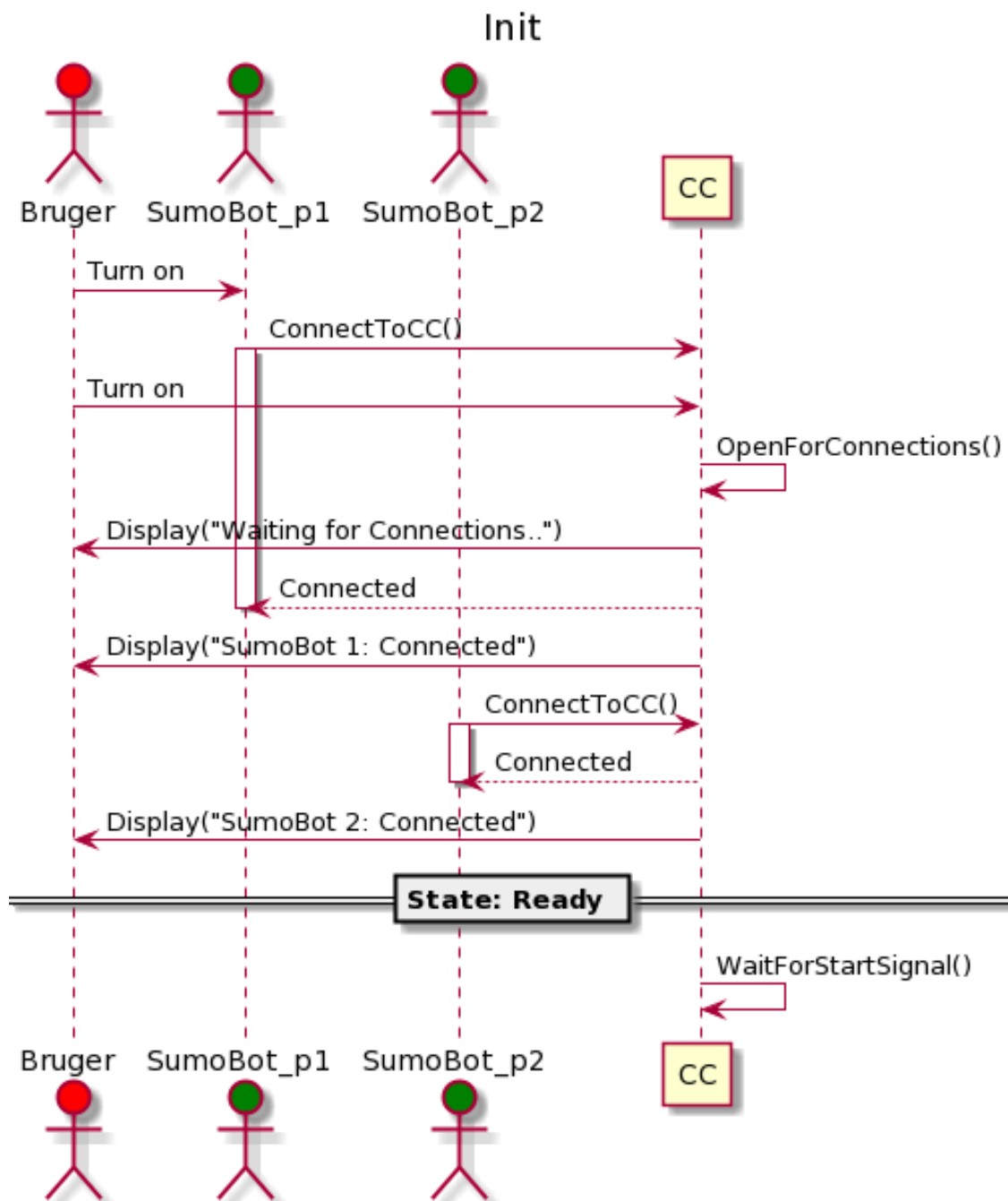
Input mode

Referencespændinger

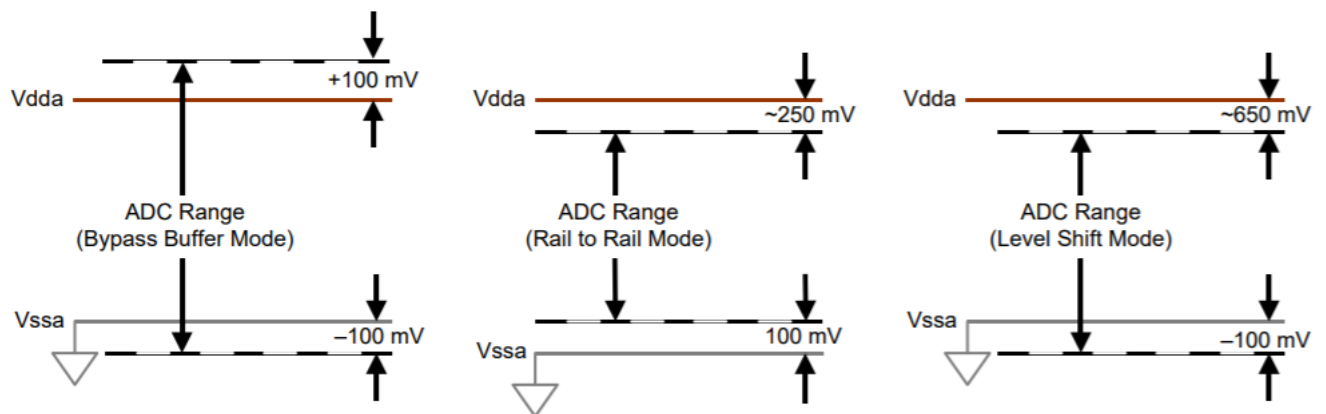
PSoC 5LP har interne referencespændinger med en tolerance på $\pm 0.1\%$. (TBR)Reference til datablad for PSoC, side 94.

Interrupt, polling, DMA

7.3 SumoBot Design



Figur 7.5: Sekvensdiagram over initialisering



Figur 7.6: Forskellige input modes for Delta Sigma ADC

8. Implementering

Litteratur

- [1] "IEEE 802.11," in *Encyclopedia of Wireless Networks*. Springer International Publishing, 2020, pp. 583–583.
- [2] F. M. Mims, *Formulas, Tables and Basic Circuits*, 1st ed. Radio Shack Electronics, 1984.
- [3] J. Lewis, "Understanding Microphone Sensitivity," vol. 46, no. 2, p. 3, 2012. [Online]. Available: <https://www.analog.com/en/analog-dialogue/articles/understanding-microphone-sensitivity.html>
- [4] W. Kester, "MT-001 TUTORIAL Taking the Mystery out of the Infamous Formula, "SNR = $6.02N + 1.76\text{dB}$,"and Why You Should Care," Tech. Rep.

A. Revisionshistorik

Revision	Dato	Forfatter	Beskrivelse
1.0	11/11/2020	RK	Revisionshistorik tilføjet