



---

## 3. Semester - Projektrapport

### Spillemaskine

Projektgruppe 4

---

#### Gruppemedlemmer:

Tonny Nguyen  
Studienr.: 202010614

Mohamad Maher Alkayem  
Studienr.: 201903802

Rasmus Westergaard  
Studienr.: 202006482

Heja Hassan  
Studienr.: 201900316

Oscarboe Vesthy Damgaard Nielsen  
Studienr.: 201811388

Sezer Isik  
Studienr.: 20111338

Magnus Kappel Petersen  
Studienr.:

**Vejleder:**  
Søren Hansen

Ingeniørhøjskolen Aarhus Universitet  
4. april 2022

# 1 Resume

## 2 Abstract

# Indhold

<b>1</b>	<b>Resume</b>	<b>1</b>
<b>2</b>	<b>Abstract</b>	<b>2</b>
<b>3</b>	<b>Ordforklaring</b>	<b>7</b>
<b>4</b>	<b>Arbejdsfordeling</b>	<b>8</b>
<b>5</b>	<b>Forord</b>	<b>9</b>
<b>6</b>	<b>Projektformulering</b>	<b>10</b>
<b>7</b>	<b>Indledning</b>	<b>12</b>
<b>8</b>	<b>Metode og proces</b>	<b>13</b>
8.1	Proces . . . . .	13
8.2	SCRUM . . . . .	13
8.3	Udviklings modeller . . . . .	13
8.4	Rollefordeling . . . . .	13
<b>9</b>	<b>Kravspecifikationer</b>	<b>14</b>
9.1	Aktørbeskrivelse . . . . .	14
9.2	Usecase 1: Spil på maskine . . . . .	16
9.3	Usecase 2: Restock af maskinen . . . . .	16
9.4	Ikke-funktionelle krav . . . . .	17
9.4.1	Performance . . . . .	17
9.4.2	Useability . . . . .	17
9.4.3	Reliability . . . . .	17
9.4.4	Supportability . . . . .	17
9.4.5	Fysiske dimensioner . . . . .	17
9.4.6	Tekniske specifikationer . . . . .	17

<b>10 Systemarkitektur</b>	<b>18</b>
10.1 Overordnet BDD . . . . .	18
10.1.1 Blokbeskrivelse . . . . .	19
10.2 Overordnet IBD . . . . .	19
10.2.1 Signalbeskrivelse . . . . .	21
10.2.2 BDD Allocate diagram . . . . .	22
10.3 Sekvensdiagram for Use case 1 . . . . .	23
10.4 Sekvensdiagram for Use case 2 . . . . .	24
10.5 Applikationsmodel . . . . .	25
10.5.1 Domænemodel . . . . .	25
10.5.2 Klassediagram . . . . .	25
10.5.3 Sekvensdiagram . . . . .	26
<b>11 Accepttestspecifikationer</b>	<b>27</b>
11.1 Introduktion . . . . .	27
<b>12 Udbetalingsenhed</b>	<b>30</b>
12.1 Analyse . . . . .	31
12.2 Arkitektur . . . . .	32
12.2.1 BDD for udbetalingsenhed . . . . .	32
12.2.2 IBD for udbetalingsenhed . . . . .	33
12.2.3 System sekvensdiagram for udbetalingsenhed . . . . .	33
12.3 Design . . . . .	34
12.4 Implementeringen . . . . .	35
12.5 Modultest . . . . .	36
<b>13 Hjul og -styring</b>	<b>37</b>
13.1 Analyse . . . . .	38
13.2 Arkitektur . . . . .	39
13.2.1 BDD og blokbeskrivelse . . . . .	39
13.2.2 IBD og signalbeskrivelse . . . . .	41
13.2.3 Referencer . . . . .	41
13.3 Design . . . . .	43
13.4 Implementeringen . . . . .	44
13.5 Modultest . . . . .	45

<b>14 Indbetalingsssenhed</b>	<b>46</b>
14.1 Analyse . . . . .	47
14.2 Arkitektur . . . . .	48
14.2.1 Blok Definitions Diagram . . . . .	48
14.2.2 Blokbetegnelse . . . . .	49
14.2.3 Interne Blok Diagrammer . . . . .	49
14.2.4 Signaletegnelse . . . . .	49
14.3 Design . . . . .	53
14.4 Implementeringen . . . . .	54
14.5 Modultest . . . . .	55
<b>15 Spilstyringsenhed</b>	<b>56</b>
15.1 Analyse . . . . .	57
15.2 Arkitektur . . . . .	58
15.2.1 BDD . . . . .	58
15.2.2 Blokbetegnelse . . . . .	59
15.2.3 Hardware . . . . .	59
15.3 Design . . . . .	60
15.4 Implementeringen . . . . .	61
15.5 Modultest . . . . .	62
<b>16 Brugergrænseflade</b>	<b>63</b>
16.1 Analyse . . . . .	64
16.2 Arkitektur . . . . .	65
16.2.1 BDD . . . . .	65
16.2.2 Blokbetegnelse . . . . .	65
16.2.3 IBD . . . . .	66
16.3 Design . . . . .	67
16.4 Implementeringen . . . . .	68
16.5 Modultest . . . . .	69
<b>17 Integrationstest</b>	<b>70</b>
<b>18 Diskussion af resultater</b>	<b>71</b>

<b>19 Konklusion</b>	<b>72</b>
<b>20 Perspektivering</b>	<b>73</b>
20.1 Fremtiddigt arbejde . . . . .	73

### 3 Ordforklaring



## 4 Arbejdsfordeling

## 5 Forord


## 6 Projektformulering

### SpilleMaskine


Projektets øverste mål er at designe en spilleautomat til underholdning hvorpå man kan spille casino slots

Kort beskrivelse af projektet

Vores spilleautomat skal kunne tage imod credit, og kunne udlodde præmier ved spil. Spillet skal foregå som det klassiske casino slots som spilles på de kendte enarmede tyveknægte. Spillet skal som udgangspunkt have 3 “ruller”, der snurrer og stopper tilfældigt på et symbol. Hvis den stopper på 3 ens symboler, har man vundet. Der kan være forskellige symboler med forskellige størrelser af præmier. Dette program vil være softwarebaseret, og vil blive udført af et linux program eksekveret på en Raspberry Pi.

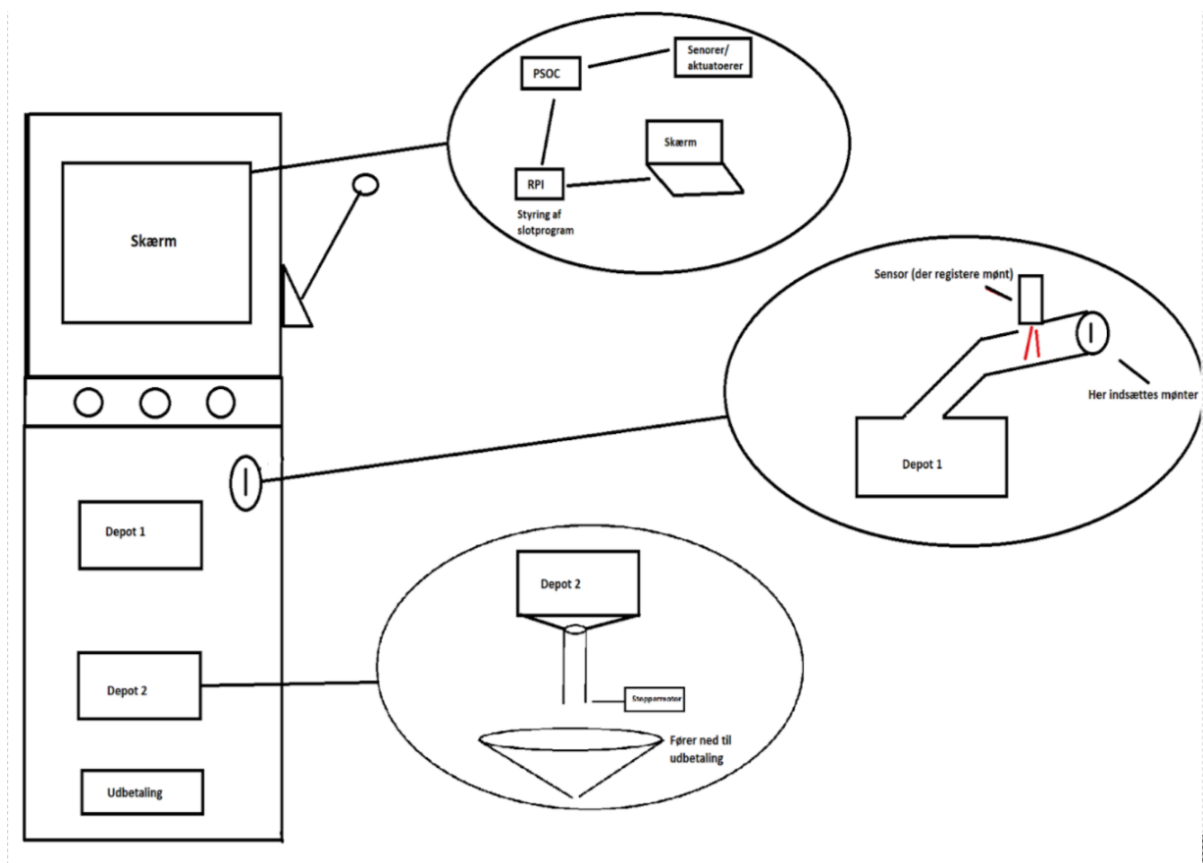
Vi vil bruge bevægelsessensorer til at aflæse om det er indsat mønter og om forskellige knapper er trykket. Vi vil også bruge en sensor til at se om et håndtag/knap er trukket/trykket på. Evt. kan vi tilføje en mønt sortering. 

Vi vil bruge aktuatorer til at udbetale mønter, og sørge for at dette sker med den korrekte mængde og type. Disse sensorer skal være forbundet til- og styret af en P-soc.

De 2 depotrum som ses på det rige billede fungerer som opbevaring af mønter. Begge depoter vil kunne tilgås via nogle låger. Dette kan selvfølgelig kun gøres af ejeren vha. en nøgle. Ejeren vil desuden skulle fylde mønter i udbetalings-depotrummet løbene, så maskinen har kapital til at udbetale præmier. 

Depot 1 er der hvor indsatte mønter ender op, og man kan på den måde også se hvad maskinen har indtjent indenfor en given periode.

Depot 2 er det lager af mønter som bruges til udbetaling hvor de vha. en steppermotor skubbes ned til udbetaling.



Figur 1: Rigt billede

## 7 Indledning

## 8 Metode og proces

### 8.1 Proces

### 8.2 SCRUM

### 8.3 Udviklings modeller

### 8.4 Rollefordeling

## 9 Kravspecifikationer

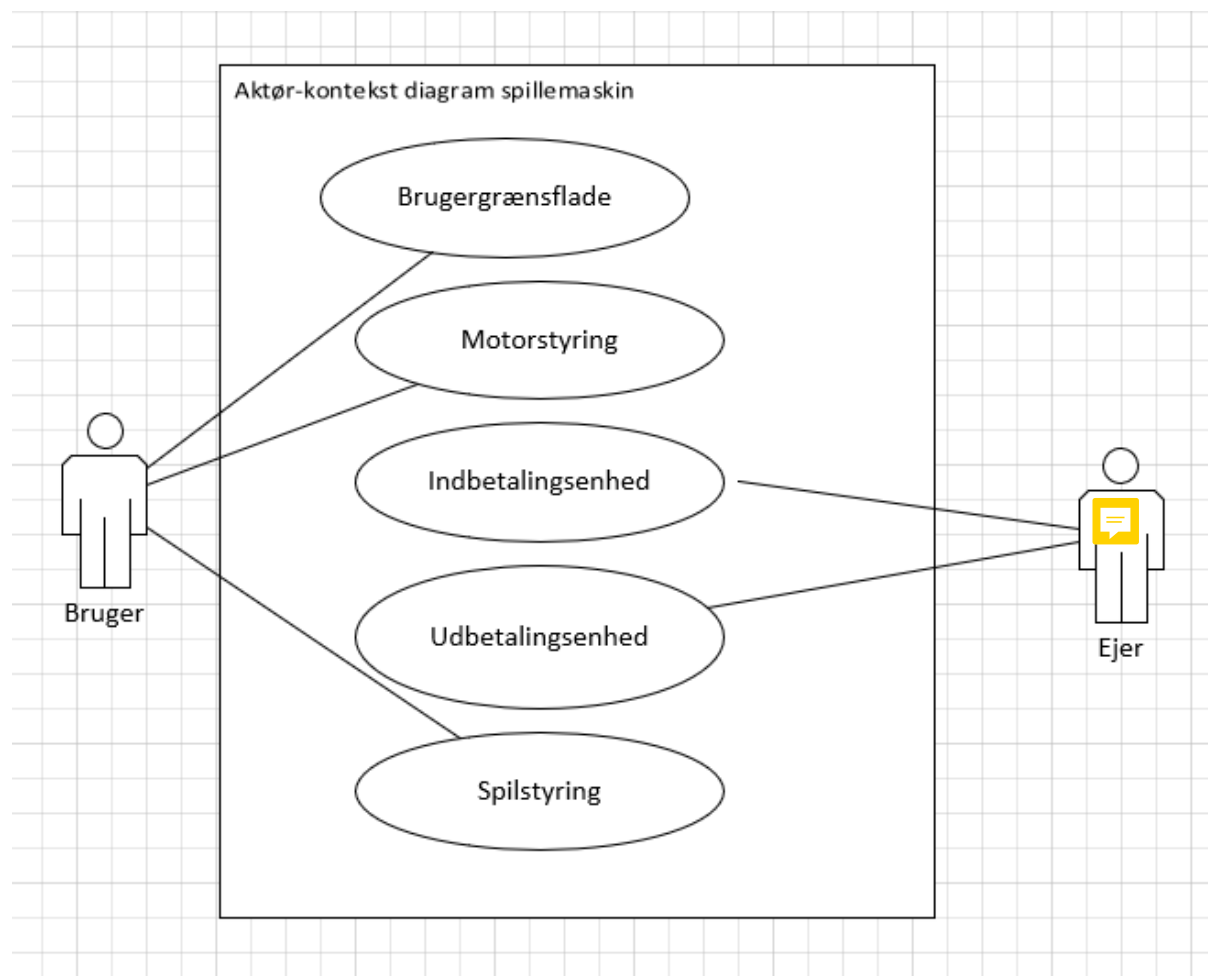
I dette afsnit vil de kravspecifikationer til systemet blive beskrevet. Da det var svært at finde datasheet på internettet, så prøvede vi at kontakte RoyalCasino, og spille myndigheder og høre om de kan hjælpe, men fik vi ikke stor hjælp. Derfor blev vi nødt til at selv definere systemet krav, og nogle af kraven er defineret i forhold til de regler, som findes på spil myndighedernes hjemmeside. Systemet er blevet delt til undermoduler, som ses nedenunder.



- Brugergrænseflade -Mohammad  
Man skal kunne stå op og spille.  
Man skal kunne aktivere spillet ved at trække i en arm eller trykke på knap
- Spilstyring –Rasmus og Oscar  
Tilbagebetalingsprocenten skal variere med max 0.5  
Spilstyringsenheden skal kunne evaluere om spillet er vundet, ved aktiveret spil.  
Spilstyringsenheden skal kunne få resultatet udtrykt på hjulene.  
Et spil (fra man trykker start, til man kan trykke start igen) skal minimum vare 2 sekunder jf samtale med spillemyndighederne/retsinformationens hjemmeside  
Desuden skal det vare max 6 sekunder, da der må være en grund til at de har måtte lave den regel, og derfor ønsker firmaerne at få tiden ned.  
Der skal være en udbetalingsprocent på mindst 74 % jf retsinformationens hjemmeside.
- Hjul og hjulstyring - Sezer  
Systemet skal have min. ét hjul og maks. tre hjul.  
Hjulet/hjulene skal bestå af 12-18 felter med symboler (jf. samtale med spillemyndighederne)  
Hjulet/hjulene skal være fysisk forbundet med en motor, og dreje synkront rundt med motorakslen.  
Hjulet/hjulene skal stoppe ved symbolet, som angives af spilstyringsenheden.  
Hjulene (hvis flere end 1) skal have samme fysiske dimensioner og samme antal felter med samme symboler i samme rækkefølge.  
Et spil (fra man trykker start, til man kan trykke start igen) bør minimum vare 2 sekunder (jf. <https://www.retsinformation.dk/eli/lta/2011/1302>).  
Et spil bør maksimum vare 6 sekunder.
- Udbetalingsenhed - Magnus (Tonny)  
Maskinen vil IKKE indeholde tilstrækkelige sikkerhedsforanstaltninger, så den fx ikke snydes af en mønt på en snor eller andet forsøg på snyd.  
Udbetalingsenheden skal kunne udbetale mønter ved vundet spil.
- Indbetalingsenhed Heja (Tonny)  
Indbetalingsenheden skal kunne registrere om der er kastet en mønt i den.  
Maskinen vil IKKE indeholde tilstrækkelige sikkerhedsforanstaltninger, så den fx ikke snydes af en mønt på en snor.

### 9.1 Aktørbeskrivelse

For at beskrive systemet er der blevet opstillet et Aktør-kontekst Diagram. Figur 2 vises den aktør der interagerer med systemet



Figur 2: Aktør-kontekst diagram



## 9.2 Usecase 1: Spil på maskine

Navn	Spil på maskine
Mål	Gennemfør et spil
Initiering	Aktør indkaster mønt.
Aktører	Spiller (primær)
Antal samtidige forekomster	0
Prækondition	Spillemaskine er klar
Postkondition	Et spil er udført
Hovedscenarie	<ol style="list-style-type: none"> <li>1. Aktør indkaster kredit/mønt.</li> <li>2. Spillemaskinens sensor registrerer mønten.</li> <li>3. Aktør trykker på knap/hiver i håndtag.</li> <li>4. Spillemaskinen (RaspberryPi) evaluerer om spilleren vinder vha. En random number generator.</li> <li>5. Spillemaskine gennemløber spil visuelt på hjul.</li> <li>6. Spillemaskine viser udfald på hjul. [Ext. 1: Vinder]</li> <li>7. Spil afsluttet og maskine er klar til nyt indkast.</li> </ol>
Udvidelse/ undtagelser	<p>[Extension 1:]</p> <ol style="list-style-type: none"> <li>6.1 Spillemaskines hjul lander på et vindende udfald</li> <li>6.2 Spillemaskinen udfører en udbetaling.</li> </ol>

Figur 3: Usecase 1: Spil på maskine

## 9.3 Usecase 2: Restock af maskinen

Navn	Restock af slot maskine
Mål	Genopfyldning af maskinen
Initiering	Ejer
Aktører	Ejer (primær)
Antal forekomster	En
Prækondition	At slot maskinen er ikke i brug
Postkondition	At slot maskinen har fyldt mønter i, samt er blev operationel
Hovedscenarie	<ol style="list-style-type: none"> <li>1) Åbning af døren</li> <li>2) Ejeren tager møntboksen ud</li> <li>3) Åbning af boksen</li> <li>4) Fyld mønter i</li> <li>5) Sæt boksen på maskinen igen</li> <li>6) Luk døren</li> <li>7) Tryk på start knappen i maskinen</li> </ol>
Udvidelse/ undtagelser	Ingen extension

Figur 4: Usecase 2: Restock af maskinen

## 9.4 Ikke-funktionelle krav

De ikke-funktionelle krav for systemet er beskrevet i dette afsnit.

### 9.4.1 Performance

1. Systemet skal være funktionelt uden en PC er tilsluttet
2. Spillet skal vare mindst 2 sekunder, inden det er muligt at starte næste spil.
3. Spillet bør være max 6 sekunder inden det er muligt at starte næste spil.
4. Der skal være en udbetalingsprocent på mindst 74%.

### 9.4.2 Useability

1. Man skal kunne stå op og spille.
2. Man skal kunne aktivere spillet ved at trække i en arm eller trykke på en knap.

### 9.4.3 Reliability

1. Spillet skal ramme en udbetalingsprocent, der afviger med maksimalt +/- 0.5% fra den definerede udbetalingsprocent efter 100.000 softwaretests.

### 9.4.4 Supportability

1. Systemet vil ikke kunne modtage sedler eller kort.

### 9.4.5 Fysiske dimensioner

1. Metalarmen skal have dimensionerne: 5 x 30 x 5 ( $\pm 1$ )
2. Spillemaskinens boks skal have dimensionerne: 150 x 30 x 50 ( $\pm 40$ )

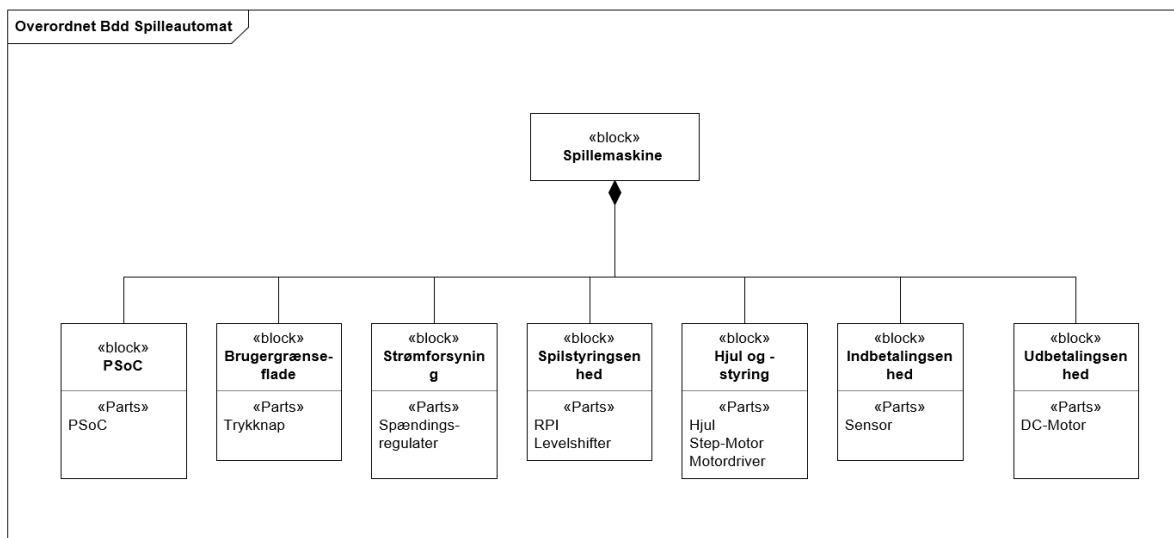
### 9.4.6 Tekniske specifikationer

1. Systemet skal have 2 møntbokse, en til udbetaling og en til pengeindsamling.
2. Systemet skal have mindst et hjul.
3. Systemet skal have en sensor og en steppermotor pr hjul.
4. Systemet bør have 3 knapper der giver mulighed for låsning af de individuelle hjul.
5. Systemet bør have en Metalarm der starter spillet.
6. Systemet bør have 3 hjul med 12 forskellige ikoner på.
7. Systemet kan være i stand til at udbetale varierede gevinster
8. Systemet kan tilkobles flere systemer over internettet, så man kan spille mod hinanden.

## 10 Systemarkitektur

### 10.1 Overordnet BDD

Herunder kan der ses det overordnede Blokdefinitionsdiagram. Der er under hvert modul listet hvilke dele dette indeholder. For mere dybdegående beskrivelser henvises der til arkitektur afsnittene for de individuelle moduler.



Figur 5: Overordnet BDD for Spillesmaskine

### 10.1.1 Blokbeskrivelse

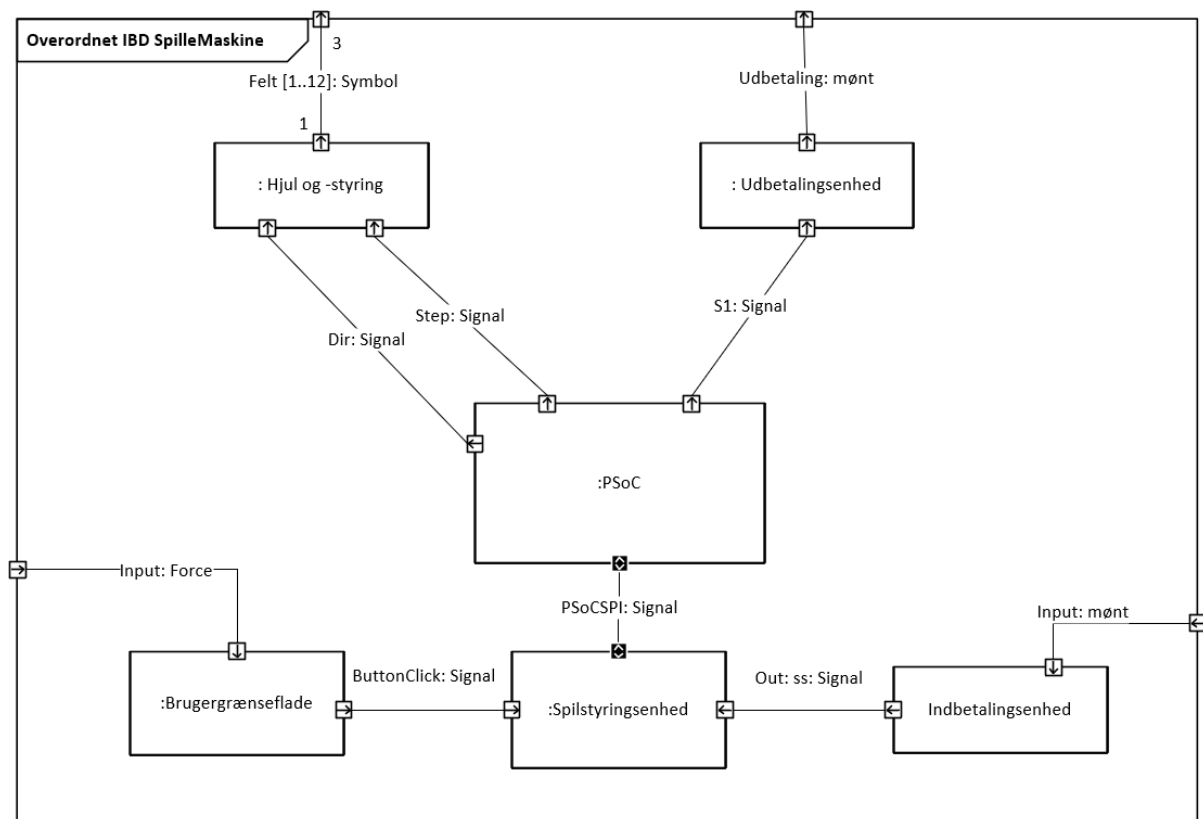
Herunder er Tabel 1, som kort beskriver hver blok:

Navn	Beskrivelse
PSoC	Består af Psoc'en. Dette modul skal fungere som slave til RPi'en der er en del af Spilstyringsenheden. Psoc'en skal styrer de motorer der er en del af hjul og -styring, og udbetalingsenhed modulet.
Brugrænseflade	Brugrænseflade indeholder de elementer som brugeren kan interagere med. Herunder bla. trykknappen til at igangsætte spil
Strømforsyning	Strømforsyningen er et modul hvor der er samlet strømforsyningen til hele systemet og der bliver der igennem reguleret spænding til hvad alle modulet hver især kræver
Spilstyringsenhed	Spilstyringsenheden er modulet som sørger for al logikken i spillets gennemførsel. Det indeholder en Raspberry Pi som kører med en integreret linuxplatform. samt en levelshifter. Rpi'ens formål er ved hvert spil at skabe et udfald. Derudover skal RPi'en udsende outputs afhængig af udfaldet. Levelshifterens rolle i modulet er at regulere spændingen mellem Rpi og PSoC'en sådan at kommunikationen der foregår som ønsket.
Hjul og -styring	Dette modul skal modtage signal fra spilstyringsenhed om et givent udfald og dermed fremvisse det på 3 roterende hjul. Dette gøres ved hjælp af 3 motorer som hvert styrer et hjul til motorerne hører der en driver til.
Indbetalingsenhed	Dette undersystem er en brugrænseflade hvor brugeren indkaster mønter, som derefter bliver registreret af en sensor. Indbetalingssystemet giver derefter besked til Spilstyringsenheden for at brugeren nu kan foretage et spil
Udbetalingsenhed	Her opbevares mønter som er indbetalt. Derudover skal Udbetalingsenheden frigive mønter såfremt modulet får et signal af Spilstyringsenhed hvor et udfald kræver udbetaling

Tabel 1: Overordnet blokbeskrivelse

## 10.2 Overordnet IBD

Det overordnede IBD viser det overordnede system og forbindelserne samt signaler imellem modulet. Der vises også input/output fra Systemet til omverdenen. Dybdgående forbindelse inbyrdes i modulet kan ses i IBD'erne for subsystemerne. Der er i det overordnede IBD for overblikkets skyld ikke inkluderet Strømforsyningsmodulet samt dennes forbindelse til de forskellige subsystemer (dette kan findes i IBD for Subsystemerne)



Figur 6: Overordnet IBD for Spillesmaskine

### 10.2.1 Signalbeskrivelse

Her ses Signalbeskrivelsen for Figur 6

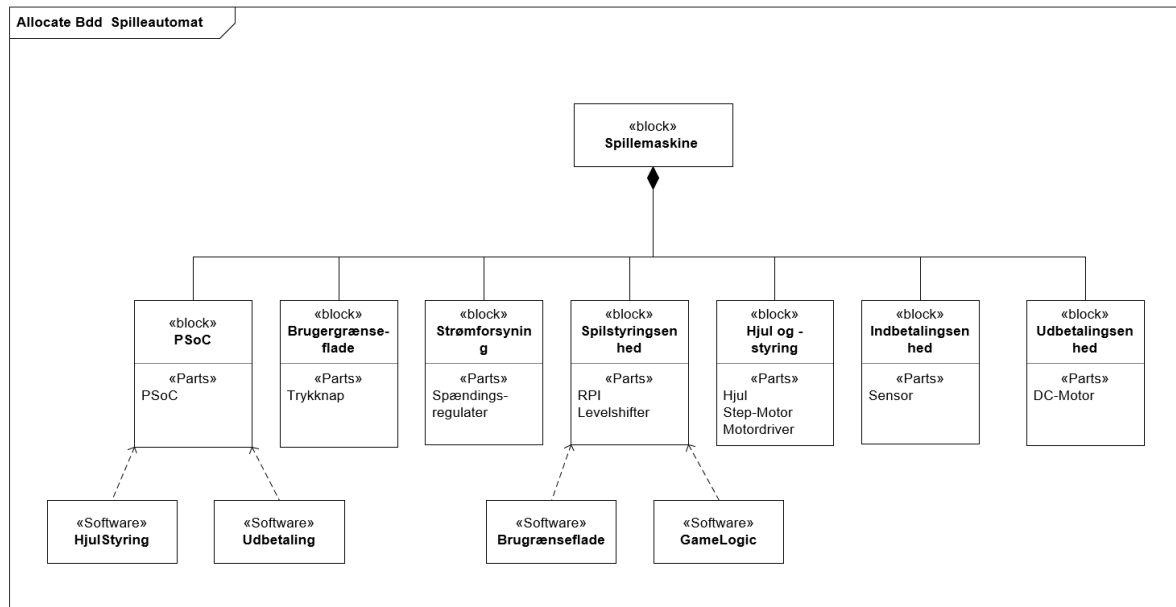
Signal-navn	Funktion	Område	Fysiske porte (source)	Fysiske porte (destination)	Kommentar
Input: Force	Input om ønsket spil	Mekanisk	Bruger	Trykknapp	
ButtomClick: Signal	Signal fra trykknapp til RPI	3.3 V	Trykknapp	RPI	
PSoCSPI: Signal	Sender Spi signaler imellem PsoC og Levelshifteren	5V	PsoC/ Levelshifter	PsoC/ Levelshifter	
Out: ss : Signal	Signal fra Sensor om indbetaling	0 - 5V	Sensor	Rpi	
Input: mønt	Indsat mønt	Mekanisk	Bruger	Sensor	
Dir: Signal	Signal til at vælge motorens omdrejningsretning	0/5V	PSoC	Motordriver	
Step: Signal	Pulssignal der skal få driveren til at køre motoren stepvist rundt	0/5V	PSoC	Motordriver	
Felt[1..12] : Symbol	Felt på hjulet, der indeholder et symbol, som vises til brugeren.	12 felter med symboler	Hjul	Bruger	
S1 : Signal	Sender signal til en h-bro som derefter åbner for strømmen til DC-motoren	0-5V	PSoC	DC-Motor	
Udbetaling: mønt	Mønt/ mønter udbetales	Mekanisk	Udbetalings-system	Bruger	

Tabel 2: Signalbeskrivelse af Overordnet IBD

### 10.2.2 BDD Allocate diagram

Her ses allocate diagrammet som beskriver allokeringen af software til hardware modulerne for det overordnede BDD. Det kan ses hvilke Applikationer der hører til de forskellige hardware-moduller

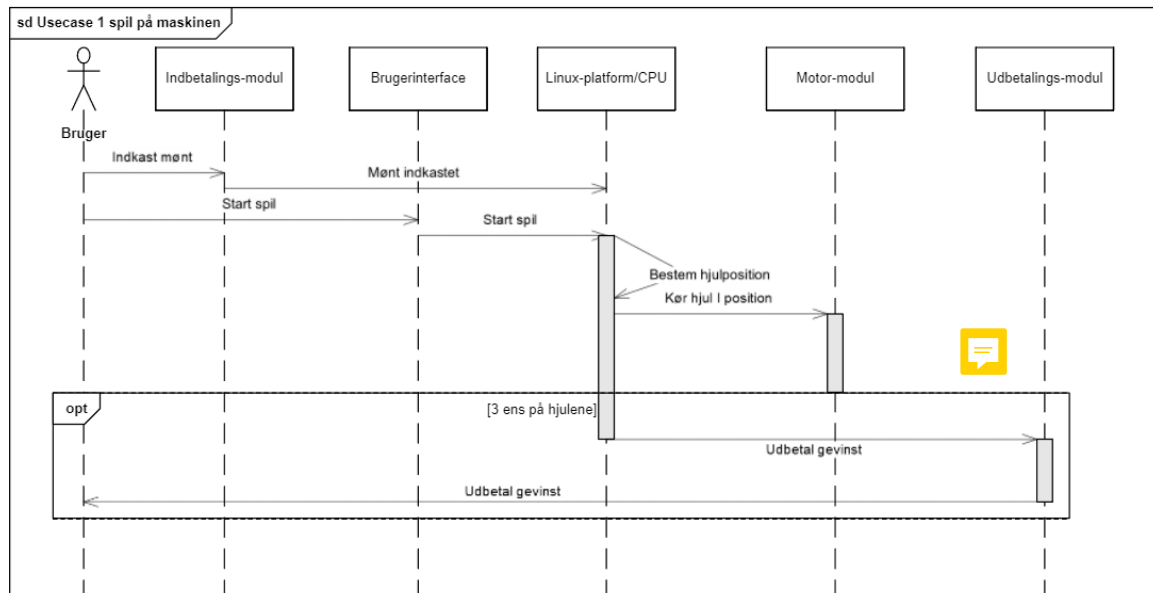
Disse applikationer er beskrevet yderligere i afsnittene til de tilhørende subsystemer



Figur 7: Overordnet BDD allocatediagram for Spillesmaskine

### 10.3 Sekvensdiagram for Use case 1

Sekvensdiagrammet beskriver kommunikationen mellem de forskellige moduler under en gennemgang af use case 1.

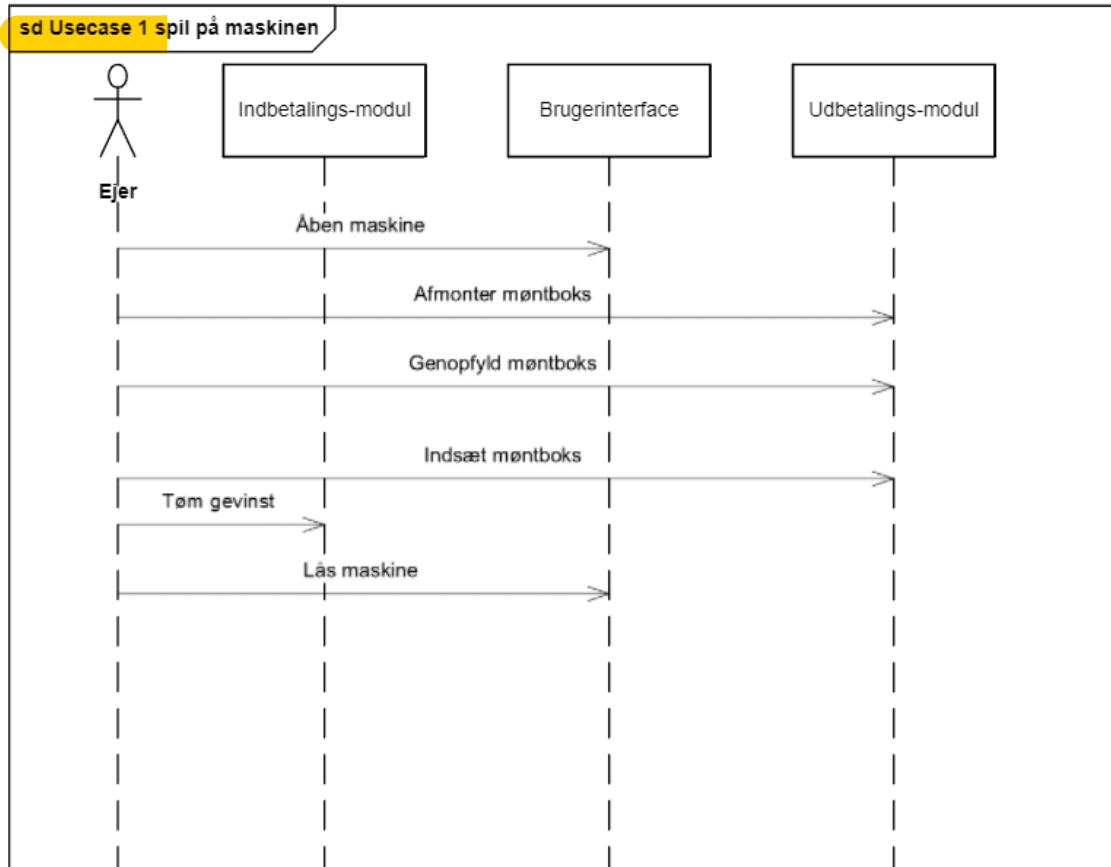


Figur 8: Sekvensdiagram for Use case 1



## 10.4 Sekvensdiagram for Use case 2

Sekvensdiagrammet beskriver kommunikationen mellem de forskellige moduler under en gennemgang af use case 2. Her er det dog primært Ejeren som interagerer med modulerne.



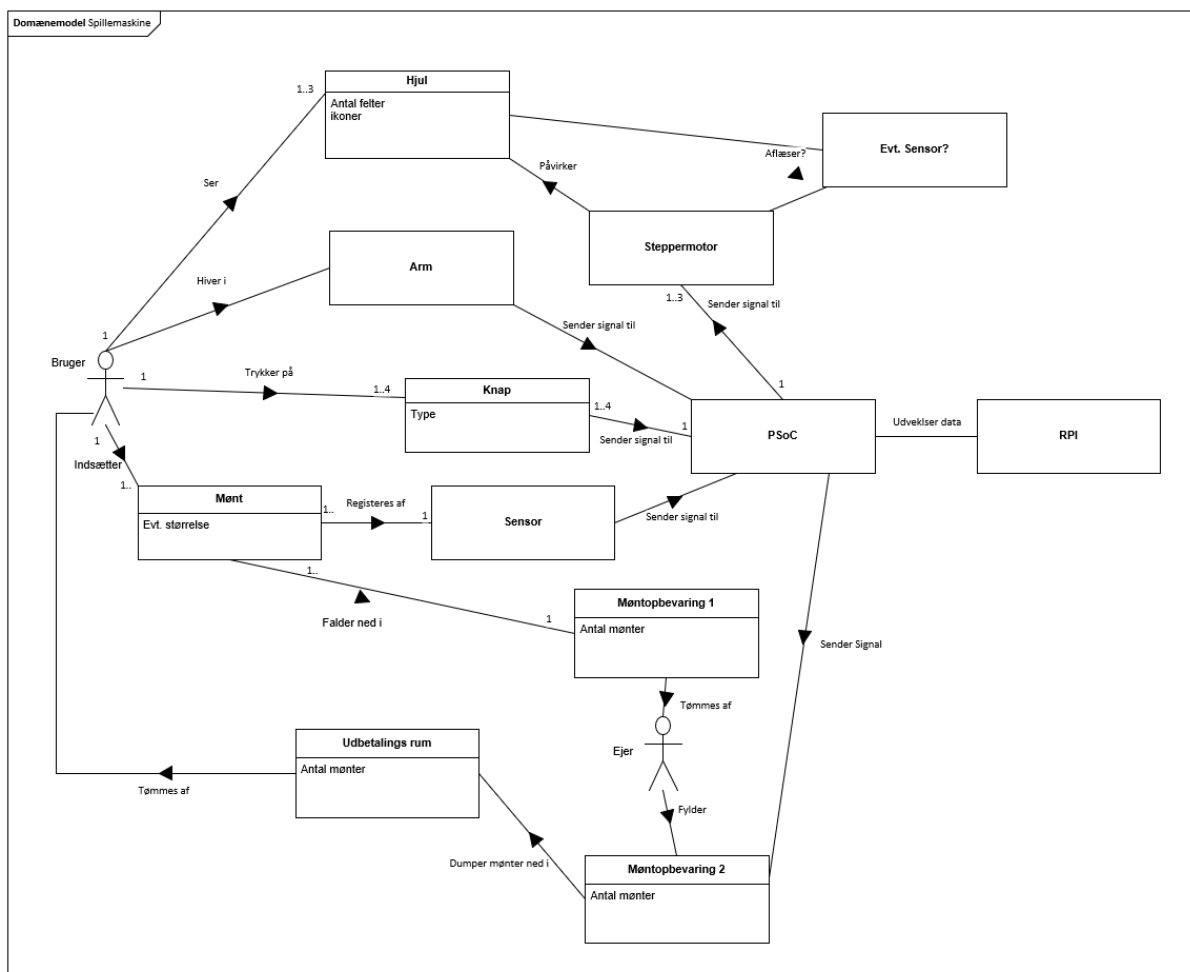
Figur 9: Sekvensdiagram for Use case 2

## 10.5 Applikationsmodel

I dette afsnit vil applikationsmodellen blive gennemgået. Først dannes et overblik vha. en domænemodel. Dernæst præsenteres de forskellige anvendte klasser i et klassediagram, og derefter vil deres funktion præsenteres i et sekvensdiagram.

### 10.5.1 Domænemodel

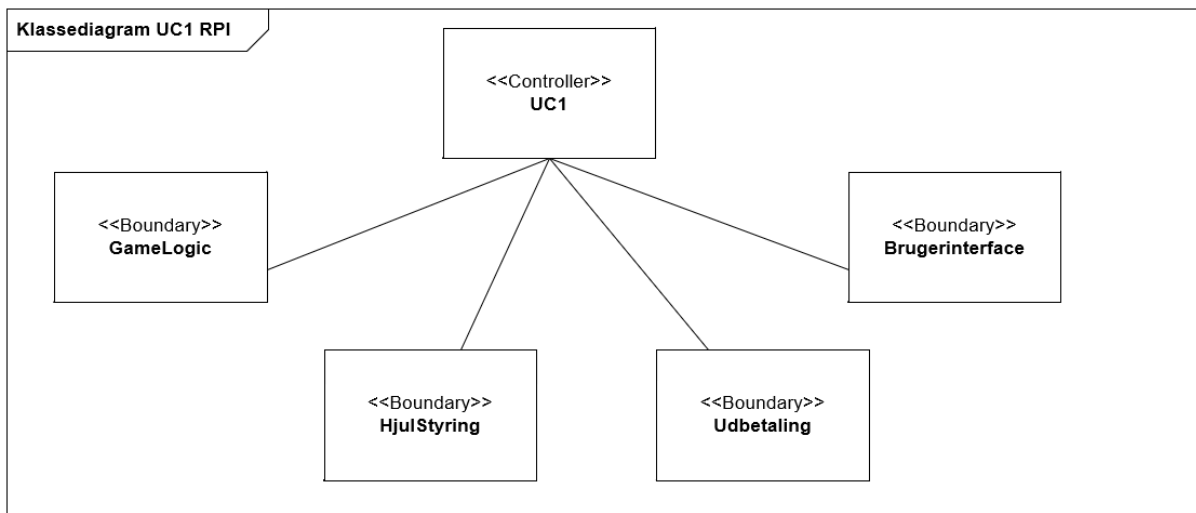
Domænemodellen beskriver systemet og forbindelserne imellem. Det er sat op i de klasser der skal bruges i systemet og der parametre som indgår i de klasser. De forbindelser hvor der ikke er beskrevet forhold skal der antages at det er 1 til 1.



Figur 10: Domænemodel for Spillesmaskine

### 10.5.2 Klassediagram

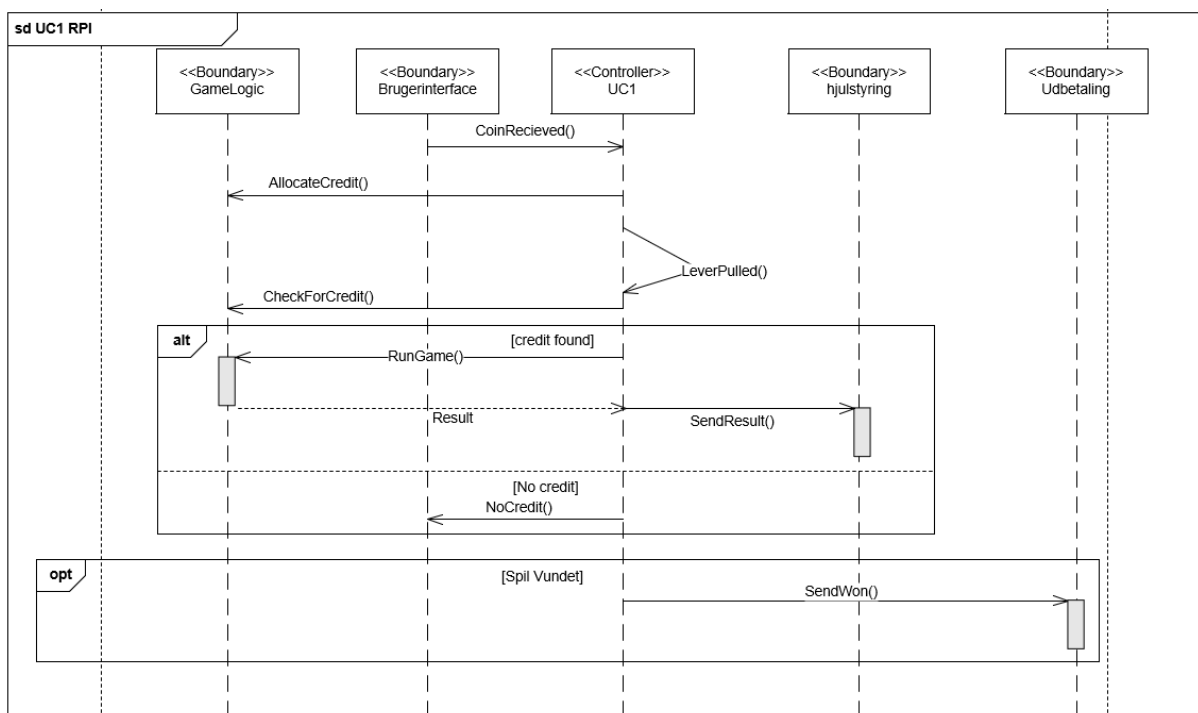
Her ses de forskellige boundary, og controlklasser, der skal implementeres på Raspberry Pi'en.



Figur 11: Klassediagram for software på RPI

### 10.5.3 Sekvensdiagram

Her ses et overblik over systemets funktion for usecase 1.



Figur 12: Sekvensdiagram for software på RPI

## 11 Accepttestspecifikationer

### 11.1 Introduktion

En accepttestspecifikation bruges til at definere, hvordan de enkelte ikke-funktionelle og funktionelle krav til et system skal testes i praksis. Her er en beskrivelse af præcist hvad der testes, hvordan det testes, og hvilket måleudstyr der er brugt til den tilhørende test. Sammenlign testresultaterne med de forventede resultater, der er blevet defineret før testen.

Krav Nr.	Krav	Test	Godkendt/ kommentar
<b>1</b>	<b>Spilstyringsenhed</b>		
<b>1.1</b>	Tilbagebetalingsprocenten skal variere med max 0.5% fra vores angivende tilbagebetalingsprocent (TBD)	Her laves et antal (TBD) software test (TBD) af algoritmen	
<b>1.2</b>	Spilstyringsenheden skal kunne evaluere om spillet er vundet, ved aktiveret spil.	Det generede udfald skal angives som vundet eller ej. Der kigges på et sat flag?	
<b>1.3</b>	Spilstyringsenheden skal kunne få resultatet udtrykt på hjulene.	Spilstyringsenhedens output til PSoC skal stemme overens med det generede udfald der kigges på et signal samt udfald?	
<b>1.4</b>	Et spil (fra man trykker start, til man kan trykke start igen) skal minimum være 2 sekunder jf samtale med spillemyndighederne/retsinformationens hjemmeside	Der tages tid på hvor lang tid det tager fra der trykkes på knappen, til der kan trykkes på knappen igen.	
<b>1.5</b>	Et spil skal maksimalt være 6 sekunder	Der ses på resultatet fra foregående test	
<b>1.6</b>	Der skal være en udbetalingsprocent på mindst 74 % jf retsinformationens hjemmeside.	Det tjekkes, at den funde tilbagebetalingsprocent i test 1.1 er over 74%.	
<b>2</b>	<b>Udbetalingsenhed</b>		
<b>2.1</b>	Udbetalingssystemet skal kunne udbetale mønter ved vundet spil.	Spillemaskinen spilles på indtil vundet spil. Der observeres om der falder mønt(er) ud.	
<b>3</b>	<b>Brugergrænseflade</b>		
<b>3.1</b>	Spillet skal kunne aktivere ved at trække i en arm eller trykke på knap.	hjulene begynder at dreje rundt .	

Tabel 3: Accepttest for usability krav

Krav Nr.	Krav	Test	Godkendt/kommentar
<b>4</b>	<b>Indbetalingsenhed</b>		
<b>4.1</b>	Sensoren skal detektere at der er kommet en mønt ned igennem møntrøret og sende et signal til Rpi	Der bliver smidt en mønt ned igennem møntrøret indtil at sensoren har registreret en mønt.	

Tabel 4: Accepttest for usability krav

Krav Nr.	Krav	Test	Godkendt/kommentar
<b>5</b>	<b>Hjul og -styring</b>		
<b>5.1</b>	Systemet skal have min. ét hjul og maks. tre hjul	Se om systemet har min. ét hjul og maks. tre hjul	
<b>5.2</b>	Hjulet/hjulene skal bestå af min. 12 felter og maks. 18 felter med ét symbol i hvert felt	Se om hjulet/hjulene består af min. 12 felter og maks. 18 felter med ét symbol i hvert felt	
<b>5.3</b>	Hjulene (hvis flere end 1) skal have samme fysiske dimensioner og samme antal felter med samme symboler i samme rækkefølge	Se om hjulene (hvis flere end 1) har samme fysiske dimensioner ( $\pm 5\text{mm}$ ) og samme antal felter med samme symboler i samme rækkefølge	

Tabel 5: Accepttest for usability krav - Hjul og -styring

## 12 Udbetalingsenhed

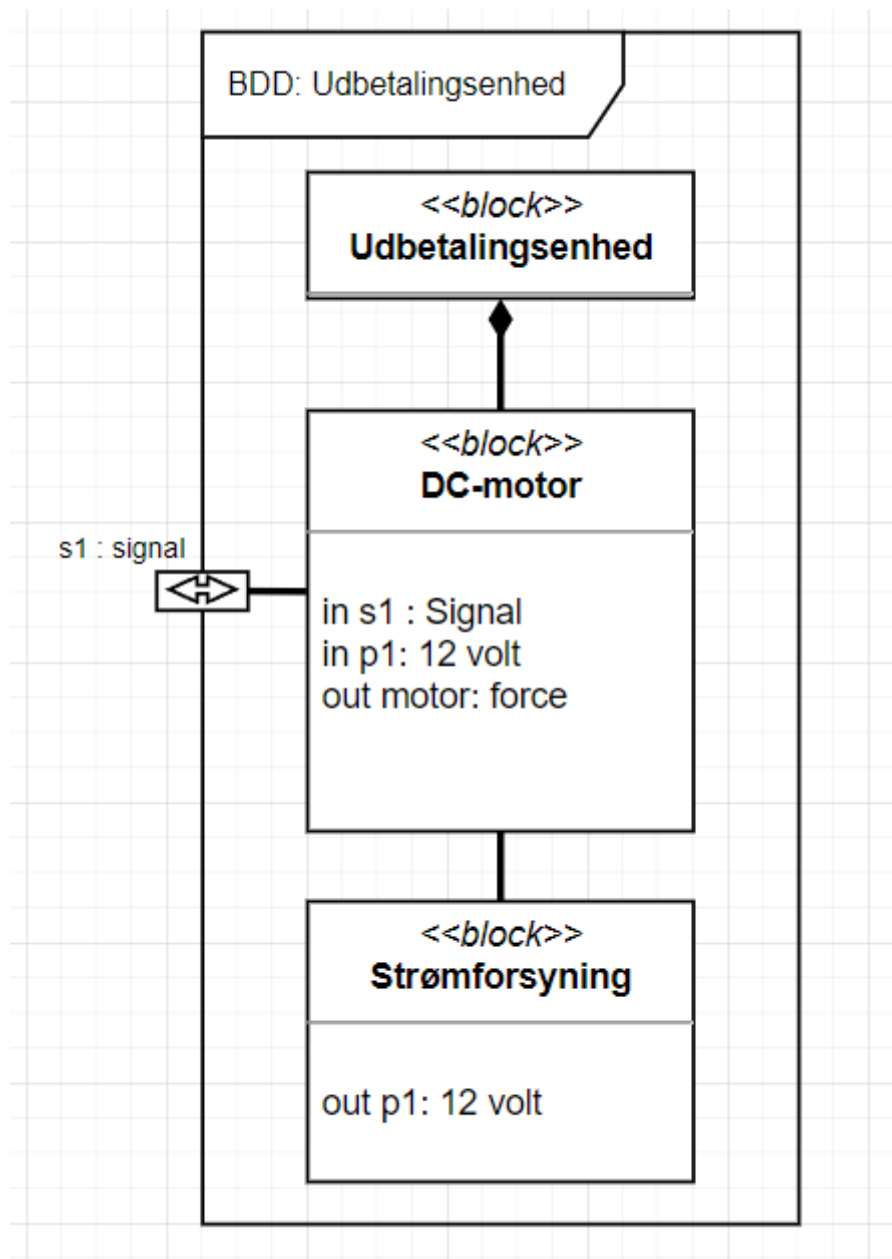
## 12.1 Analyse

Hændelse	Sandsynlighed	Konsekvens	Risiko
Møntsystemets motor kan rotere i en præcision på 5 mm i forhold til start og slut positionen	2	2	4
Møntsystem kan udbetale et bestemt antal mønter	1	2	2



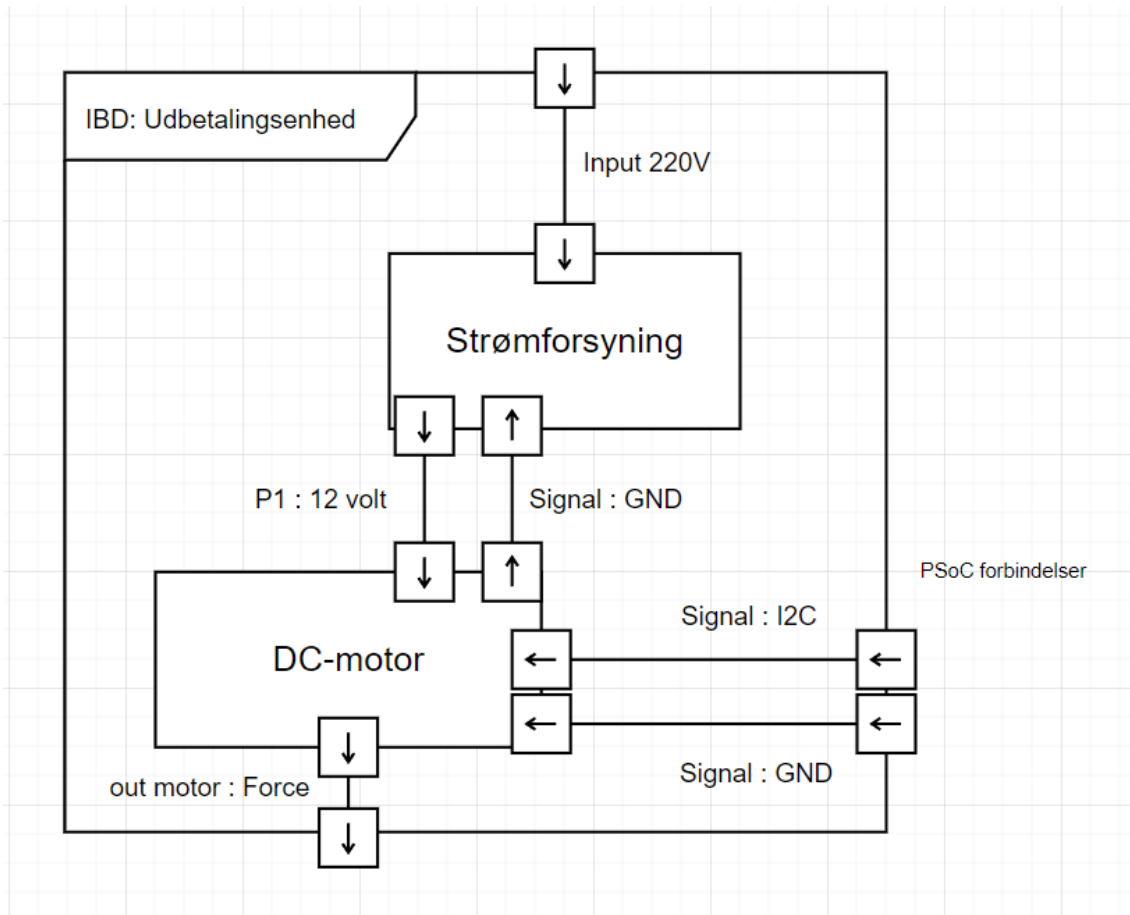
## 12.2 Arkitektur

### 12.2.1 BDD for udbetalingsenhed



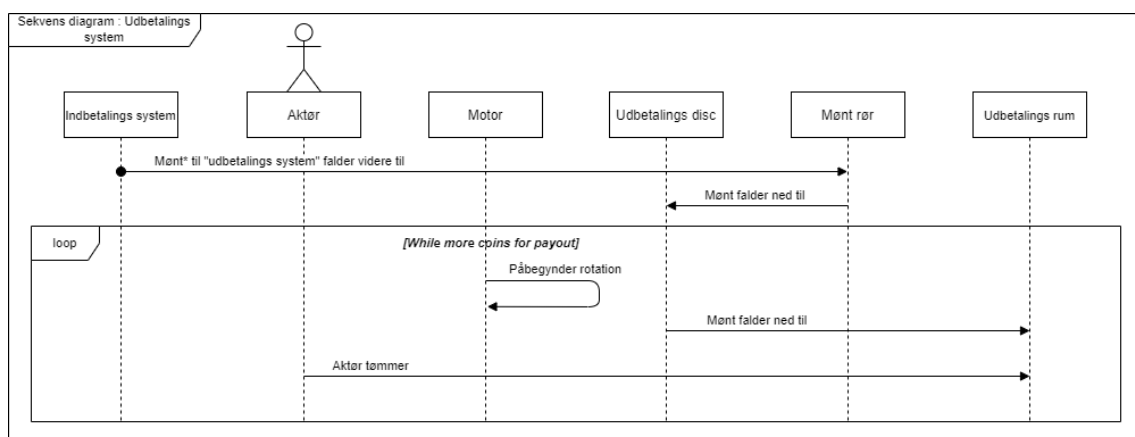
Figur 13: BDD udbetalingsenhed

### 12.2.2 IBD for udbetalingsenhed



Figur 14: IBD udbetalingsenhed

### 12.2.3 System sekvensdiagram for udbetalingsenhed



Figur 15: Sekvensdiagram for udbetalingsenhed

## 12.3 Design

## 12.4 Implementeringen

## 12.5 Modultest

## 13 Hjul og -styring

<i>Beskrivelse</i>	<i>Sandsynlighed 1-5</i>	<i>Konsekvens 1-5</i>	<i>Indvirkning 1-25</i>
Motoren stopper ikke ved den ønskede position (felt)	1	5	5
Motoren drejer rundt i mindre end 2 eller mere end 6 sek.	1	2	2
Det bliver nødvendigt at kalibrere motorpositionen i løbet af dagen	3	3	9
Motoren stopper ikke ved midten af det ønskede felt	3	2	6

Tabel 6: Risikoanalyse for Hjul og -styring

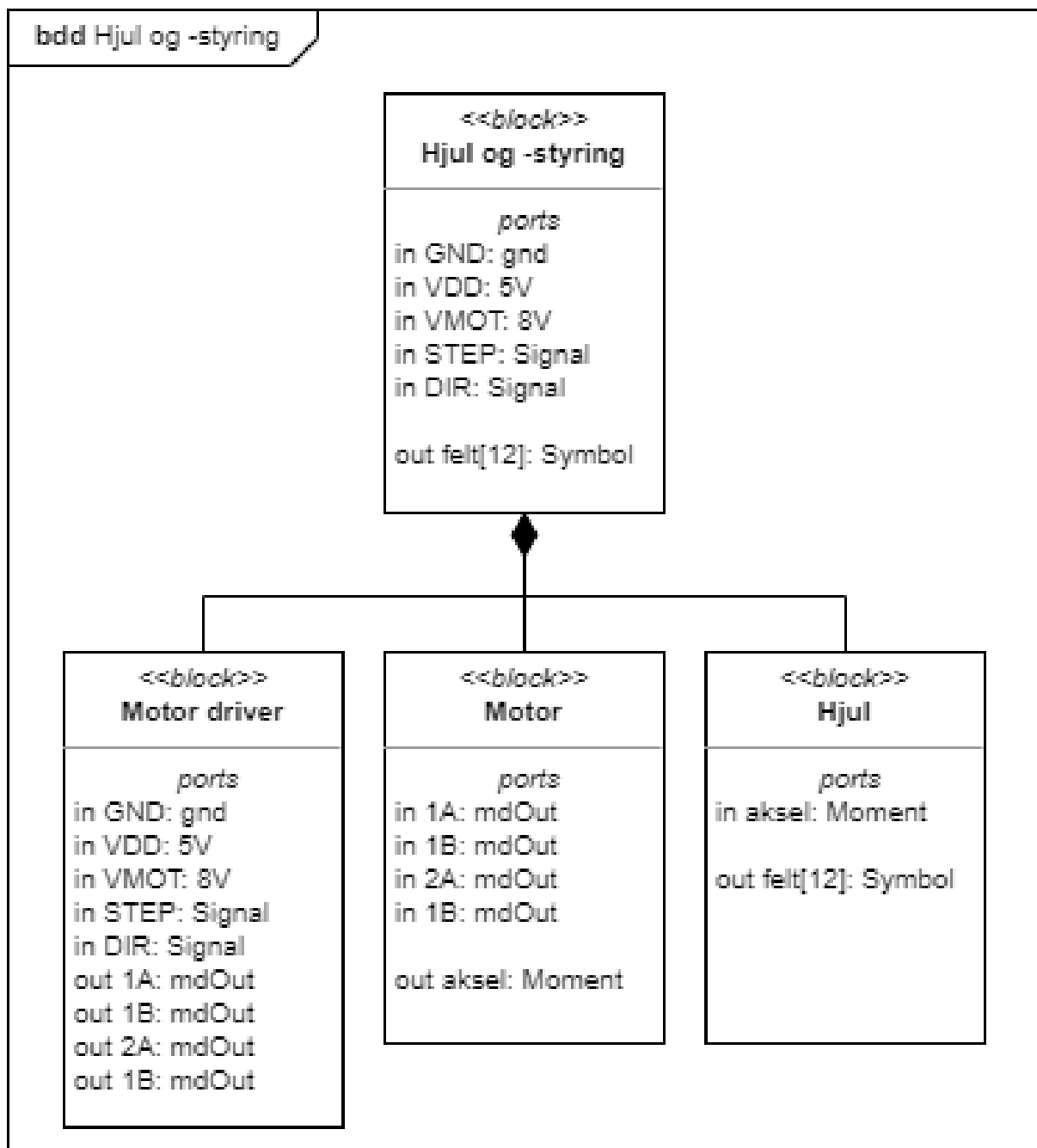
### 13.1 Analyse

Nedenunder ses en risikoanalyse af hjulstyringen bestående af Motor og Motor driver (HW). Tallene er skrevet ud fra skøn og til dels subjektivt, og afspejler derfor ikke nødvendigvis, hvordan risikoen vil kunne fortolkes af andre personer. Det er f.eks. muligt, at en anden person vil kunne se en større konsekvens af, at det kan være nødvendigt at kalibrere motorpositionen i løbet af en dag.

## 13.2 Arkitektur

Nedenunder findes hhv. BDD, blokbeskrivelse, IBD og signalbeskrivelse for blokken, der er kaldt Hjul og -styring. Hvis man kigger på BDD, vil man kunne gennemskue, at der med hjulstyring menes Motor driver og Motor, der sammen skal styre hjulet.

### 13.2.1 BDD og blokbeskrivelse



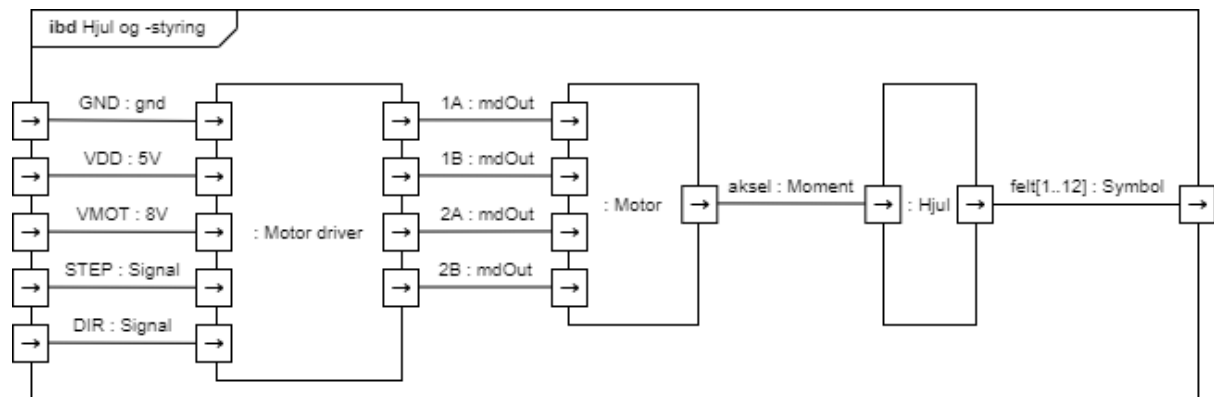
Figur 16: BDD for Hjul og -styring



Bloknavn	Funktionsbeskrivelse	Port	Signaltype	Kommentar
Hjul og -styring	Modul bestående af Hjul, Motor driver og Motor	GND	gnd	Spændingsreference
		VDD	5V	Spændingsforsyning til driver
		VMOT	8V	Spændingsforsyning til forsyning af Motor
		STEP	Signal	Logisk signal fra PSoC
		DIR	Signal	Logisk signal fra PSoC
		felt	Symbol	Feltet på hjulet, der indeholder et symbol, som vises til brugeren. Der findes i alt 12 felter.
Motor driver	Styrer motoren ud fra nogle udefrakommende signaler	GND	gnd	Spændingsreference
		VDD	5V	Spændingsforsyning til driver
		VMOT	8V	Spændingsforsyning til forsyning af Motor
		STEP	Signal	Logisk signal fra PSoC
		DIR	Signal	Logisk signal fra PSoC
		1A	mdOut	Spænding til motorspole
		1B	mdOut	Spænding til motorspole
		2A	mdOut	Spænding til motorspole
		2B	mdOut	Spænding til motorspole
Motor	Drejer hjulet	1A	mdOut	Spænding til motorspole
		1B	mdOut	Spænding til motorspole
		2A	mdOut	Spænding til motorspole
		2B	mdOut	Spænding til motorspole
		aksel	Moment	Omdrejningsmoment fra motorakslen, der skal få hjulet til at dreje rundt
Hjul		aksel	Moment	Omdrejningsmoment fra motorakslen, der skal få hjulet til at dreje rundt
		felt[12]	Symbol	Felt på hjulet, der indeholder et symbol, som vises til brugeren. Der findes i alt 12 felter.

Tabel 7: Blokbekrivelse for Hjul og -styring

### 13.2.2 IBD og signalbeskrivelse



Figur 17: IBD for Hjul og -styring

### 13.2.3 Referencer

"A4988 - DMOS Microstepping Driver with Translator And Overcurrent Protection". Allegro MicroSystems, LLC. <https://www.pololu.com/file/0J450/A4988.pdf>

"A4988 Stepper Motor Driver Carrier". Pololu Corporation. <https://www.pololu.com/product/1182>

"Wantai Mini Stepper Product Specifications". 42BYGHW811. HAOYU STAR Electronics. <http://www.haoyuelectronics.com/Attachment/42BYGHW811/42BYGHW811.pdf>

Signalnavn	Funktion	Område	Port 1 (source)	Port 2 (destination)
GND : gnd	Spændingsreference	0V	Power supply	Motor driver
VDD : 5V	Forsyningsspænding til driver	5V (max. 8mA)	Power supply	Motor driver
VMOT : 8V	Forsyningsspænding til motor	8V (1A)	Power supply	Motor driver
STEP : Signal	Pulssignal der skal få driveren til at køre motoren stepvist rundt	0/5V	PSoC	Motor driver
DIR : Signal	Signal til at vælge motorens omdrejningsretning	0/5V	PSoC	Motor driver
1A : mdOut	Spænding til motorspole, der skal styre motoren	8V (1A)	Motor driver	Motor
1B : mdOut	Spænding til motorspole, der skal styre motoren	8V (1A)	Motor driver	Motor
2A : mdOut	Spænding til motorspole, der skal styre motoren	8V (1A)	Motor driver	Motor
2B : mdOut	Spænding til motorspole, der skal styre motoren	8V (1A)	Motor driver	Motor
aksel : Moment	Moment fra motorakslen, der skal få hjulet til at dreje rundt eller får den til at stå stille	Holding torque: 4800 g-cm  Detent torque: 280 g-cm  Rotor Inertia: 68 g-cm <sup>2</sup>	Motor	Hjul
felt[1..12] : Symbol	Felt på hjulet, der indeholder et symbol, som vises til brugeren.	12 felter med symboler	Hjul	Bruger

Tabel 8: Signalbeskrivelse for Hjul og -styring

### 13.3 Design

## 13.4 Implementeringen

## 13.5 Modultest

## 14 Indbetalingsssenhed

## 14.1 Analyse

Vores risikoanalyse ses herunder, hvor vi ved flere hændelser vil gå i gang med den hændelse, som har den største risiko:

Hændelse	Sandsynlighed	Konsekvens	Risiko
Indbetalingssystemet kan registrere en mønt	2	2	4

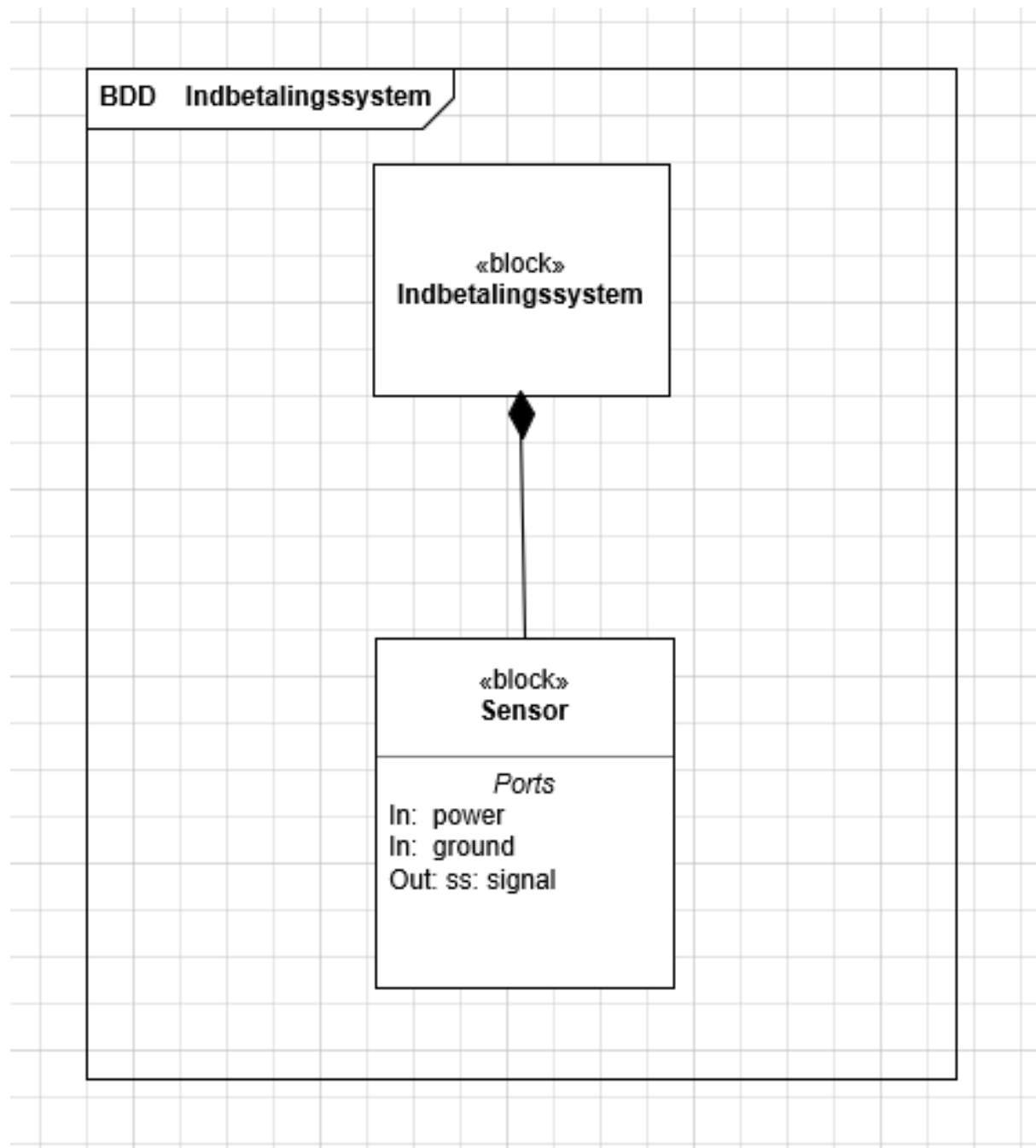


## 14.2 Arkitektur

Der vil i denne del gennemgå hardware og software arkitekturen for indbetalingsenhed. Dertil vil der blive beskrevet alle diagrammer for både hardware og software dele.

### 14.2.1 Blok Definitions Diagram

På Figur 18 ses de et overordnet blok definitions diagram over Indbetalingsenhed.



Figur 18: BDD for Indbetalingsenhed

### 14.2.2 Blokbeskrivelse

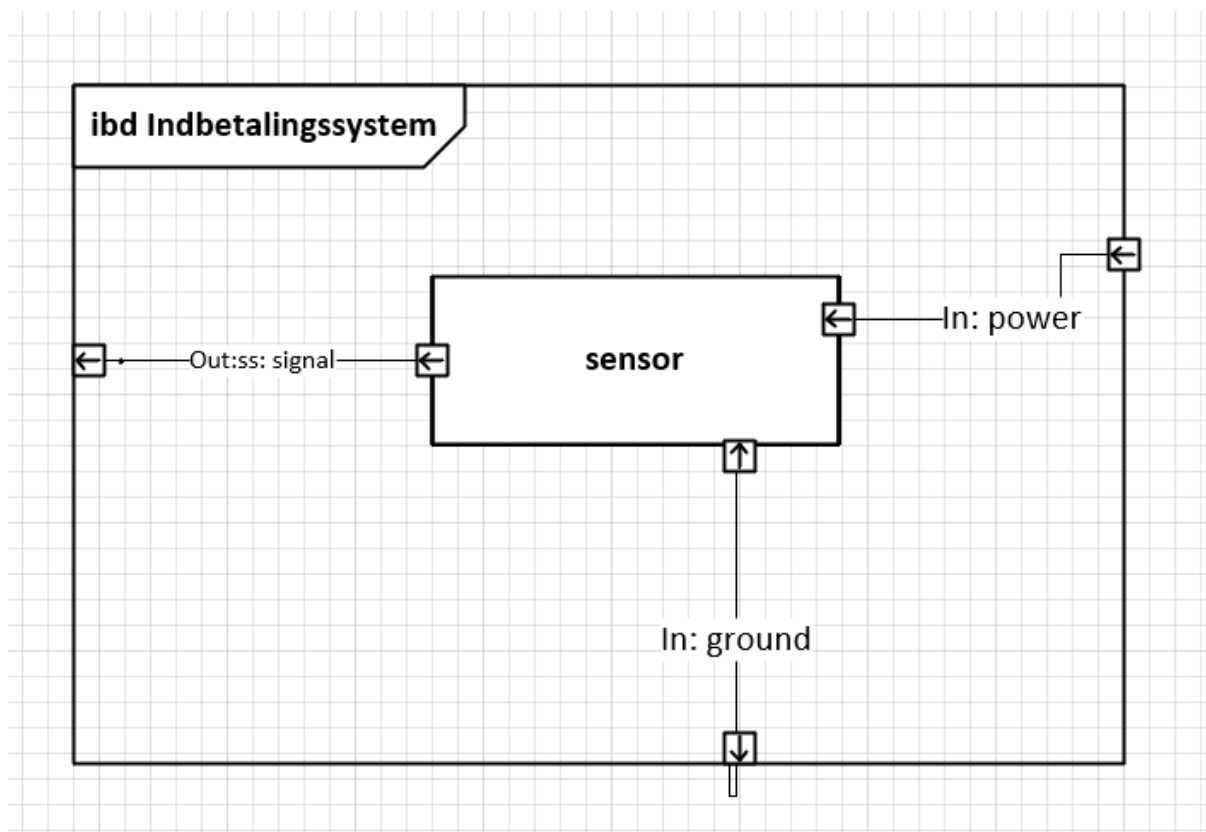
På Tabel 9 kan blokbeskriverne for Indbetalingsenhedssystemet ses.

Blok-Navn	Funktionsbeskrivelse	Signaler	Kommentar
Sensor	Detekter en mønt og sendes signalet til Rpi	in: power	
		outss signal	
		in: ground	

Tabel 9: Blokbeskrivelse af Indbetalingsenhed

### 14.2.3 Interne Blok Diagrammer

På Figur 19 ses de et overordnet Intern Blok Diagram over Indbetalingsenhed.



Figur 19: IBD for Indbetalingsenhed

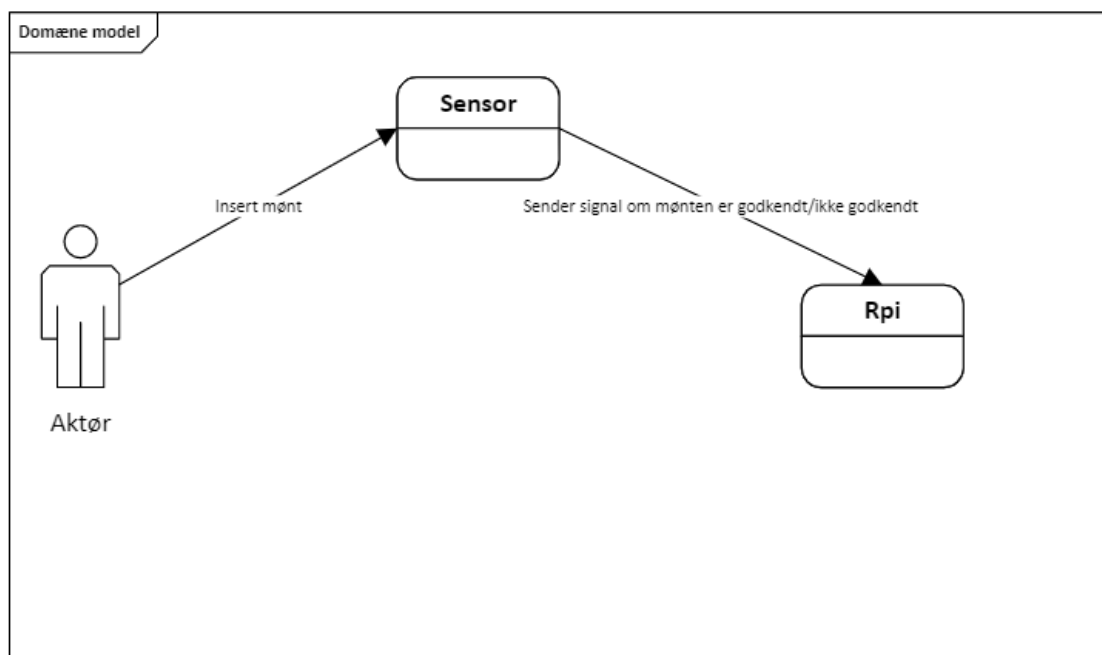
### 14.2.4 Signalbeskrivelse

På Tabel 10 kan signalbeskriverne for Indbetalingsenhedssystemet ses.

Signal-navn	Funktion	Område	Fysiske porte (source)	Fysiske porte (destination)	Kommentar
In: power	Forsyning til sensor	2.7 V - 3.6V	Ekstern strømforsyning	Sensor	
In: ground	Forsyning til sensor	0 - 5 V	Ekstern strømforsyning	Sensor	
Out : ss signal	Signal fra sensor til Rpi	0- 5	Sensor	Rpi	

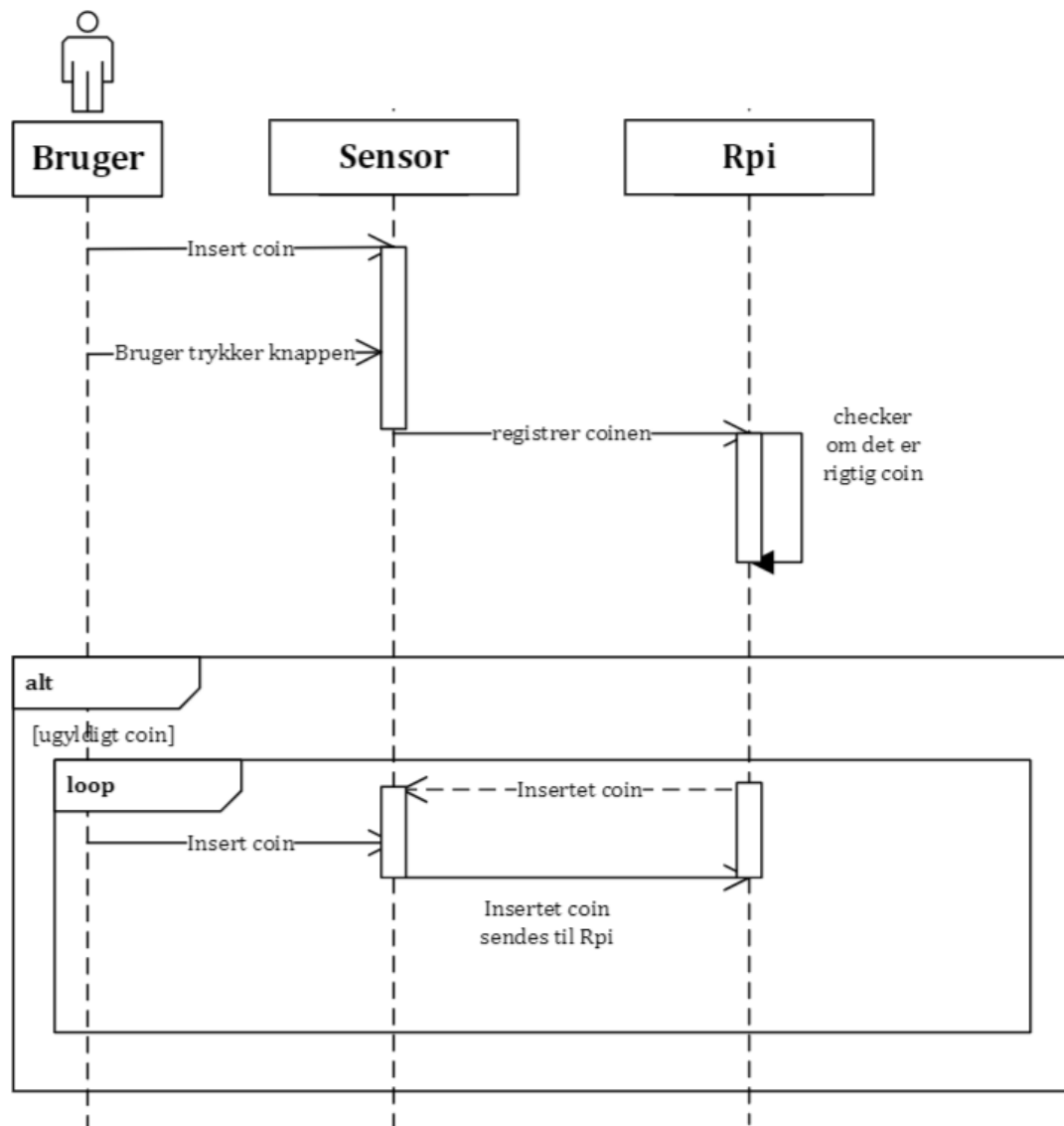
Tabel 10: Signalbeskrivelse af Indbetalingsenhed

For at få en overblik over Indbetalingsenhedssystemet er domænemodellen på Figur 20 blevet lavet. Her kan forholdene mellem brugeren og subsystemen ses.



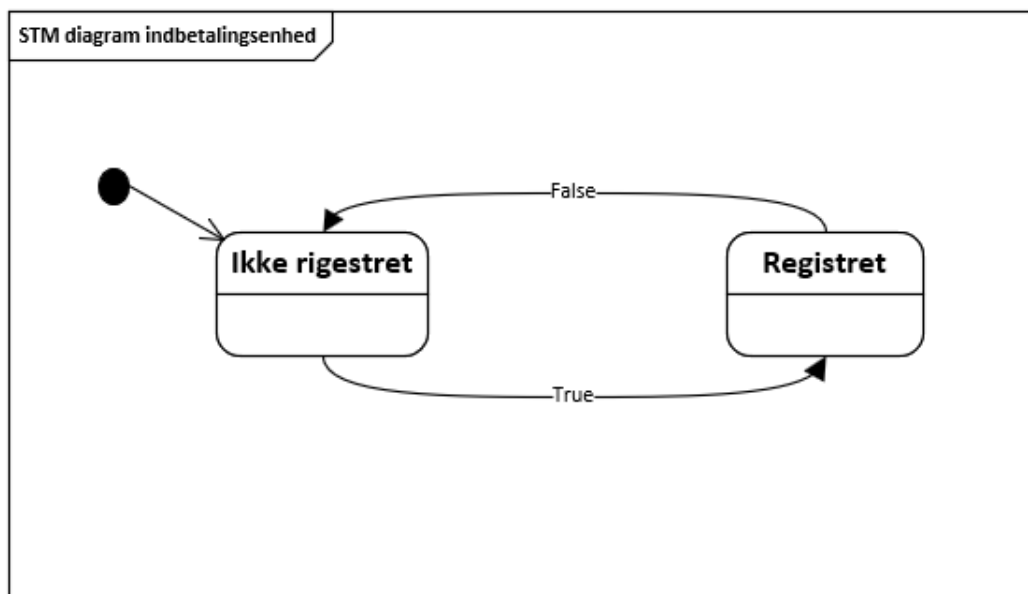
Figur 20: Domænemodel for Indbetalingsenhed

På Figur 21 kan sekvensdiagram for Indbetalingsenhedssystemet ses.



Figur 21: Sekvensdiagram for Indbetalingsenhed

På Figur 22 kan der ses et stm omhandlende hvordan de to tilstande, **Registret** og **Ikke Registret** hænger sammen.



Figur 22: State Machine Diagram for Indbetalingsenhed

## 14.3 Design

## 14.4 Implementeringen

## 14.5 Modultest



## 15 Spilstyringsenhed

## 15.1 Analyse

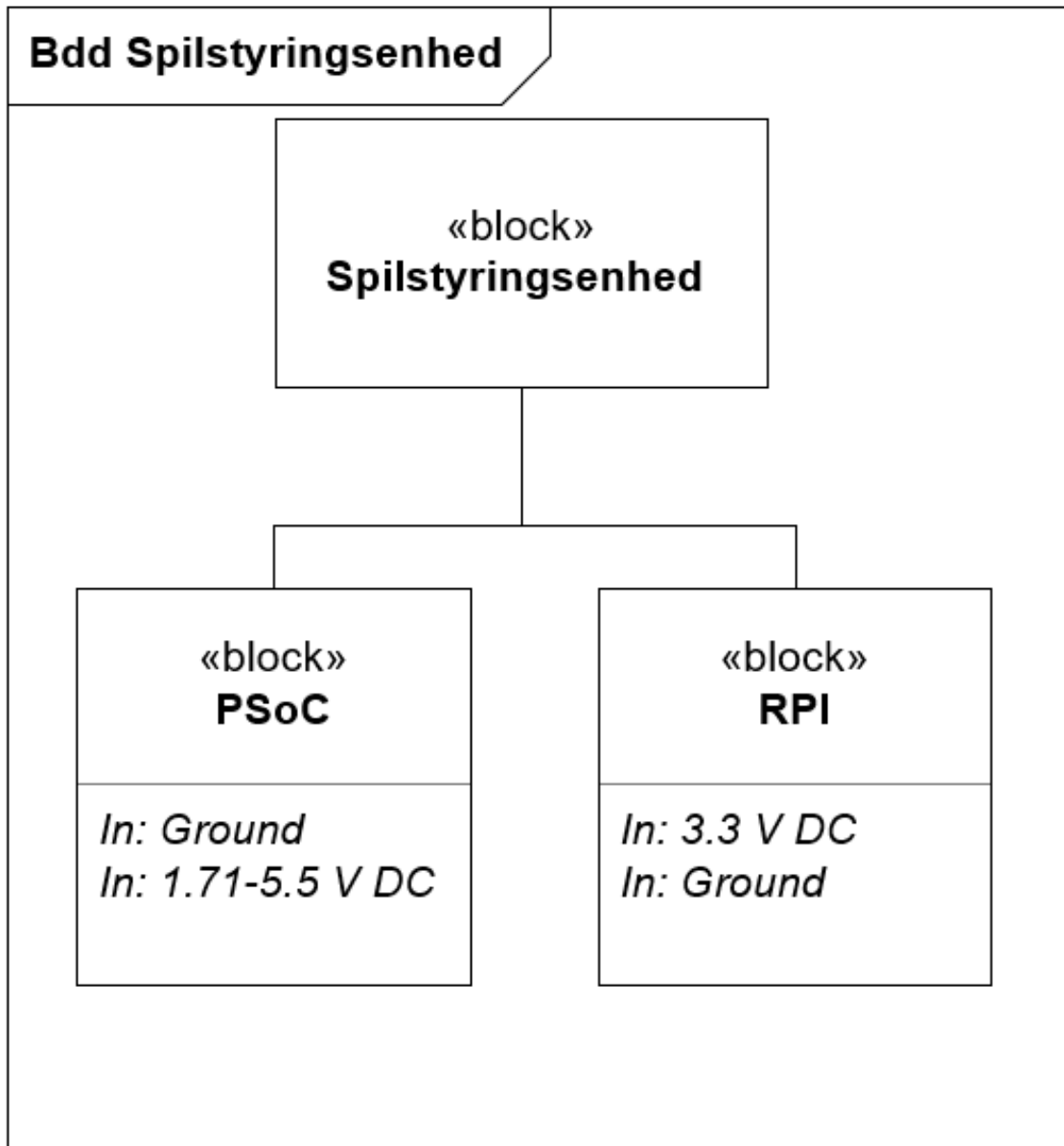
Her ses et overblik over de forskellige opgaver ifbm. spilstyringen, og den vurderede risiko/konsekvens for fejl heri.

Opgaver	Sandsynlighed for fejl	konsekvens	risiko faktor
At identificerer om spillet er vundet eller tabt	1	5	5
At identificere hvilke pladser hjulene skal lande på, og videresende informationen.	4	4	16
Modtage information om hvorvidt en mønt er indkastet	3	3	9
Modtage information om at et spil startes, og aktiverer spil derudfra.	3	3	9

Tabel 11: risikoanalysen for spilstyringen

## 15.2 Arkitektur

### 15.2.1 BDD



Figur 23: BDD for Spilstyringsenhed

### 15.2.2 Blokbeskrivelse

Bloknavn	Funktionsbeskrivelse
Raspberry Pi	Styrer spillet ved at give outputs til hjul og betaling, modtager inputs fra PSoC, brugergrænseflade og indbetalingsenhed. Raspberry Pi agere som master til PSoC'en, Den basale spilstyring kører på en integreret linuxplatform på RPI'en
PSoC	Får input fra sensorer i indbetaling og sender output motorstyringsenheden. Agerer som Slave til RPI

### 15.2.3 Hardware

#### RPI

Der bruges i spilstyringenhed en RPi som er controlleren for hele Spillesmakschine systemet. Der er ikke ændret noget hardware mæssigt så der henvises her til Datasheetet for en Raspberry pi zero W (bilag). Dog er tilføjet et shield som er det som bruges i faget HAL. Der henvises her til layout og datasheet på dette (bilag).

#### Levelshifter

levelshifteren er et undermodul i Som skal spændingsregulere imellem PSoC og RPi. Dette er en nødvendighed da PSoC arbejder med højere spændinger end RPi'en kan klarer

Levelshifteren skal derfor ned og -opjusterer signaler sådan at et højt signal der sendes fra Rpi også registreres som højt på PSoC.

## 15.3 Design

## 15.4 Implementeringen

## 15.5 Modultest

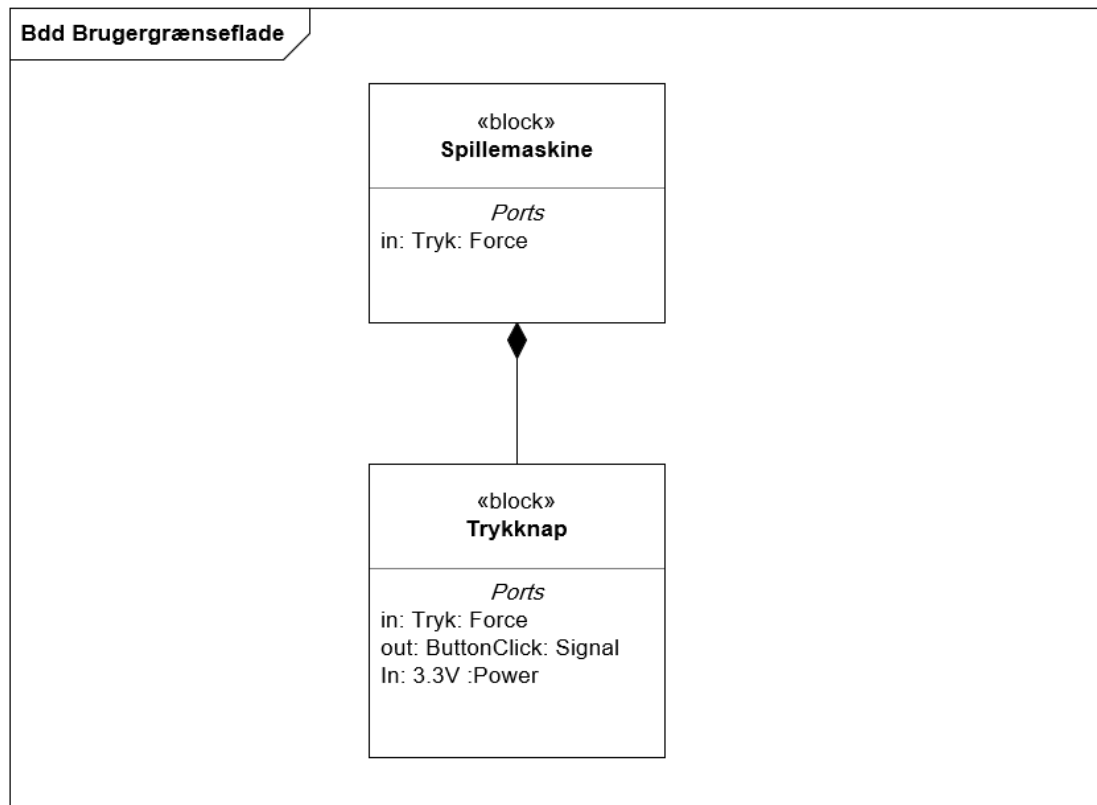
## 16 Brugergrænseflade



## 16.1 Analyse

## 16.2 Arkitektur

### 16.2.1 BDD



Figur 24: BDD for Brugergrænseflade

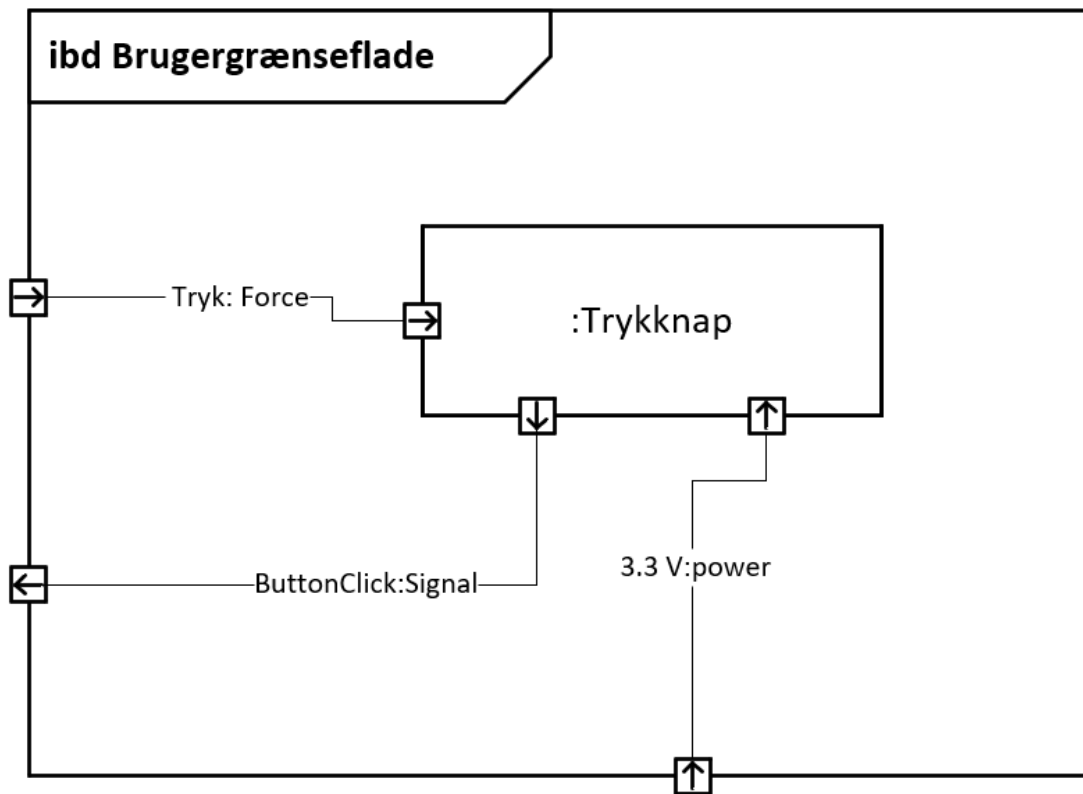
### 16.2.2 Blokbeskrivelse

På Tabel 9 kan blokbeskriverne for Brugergrænseflade ses.

Blok-Navn	Funktionsbeskrivelse	Signaler-Signaltipe	Kommentar
Trykknab	Knap til at påbegynde spillemaskine	in: Tryk / Force	Fysisk tryk på knap
		in: DC /3,3 V	Strømforsyning
		out: ButtonClick / Signal	Signal fra trykknab til RPI

Tabel 12: Blokbeskrivelse af Brugergrænseflade

### 16.2.3 IBD



Figur 25: IBD for Brugergrænseflade

## 16.3 Design

## 16.4 Implementeringen

## 16.5 Modultest

## 17 Integrationstest

## 18 Diskussion af resultater



## 19 Konklusion

## 20 Perspektivering

### 20.1 Fremtiddigt arbejde