

Dokumentation

Semesterprojekt 3. Semester

Gruppe 10

Vejleder: Søren

Gruppemedlemmer:

Navn	Studienummer
Tonni Nybo Follmann	201504573
Stefan Nielsen	201508282
Mikkel Espersen	201507348
Halfdan Vanderbruggen Bjerre	20091153
Ahmad Sabah	201209619
Jacob Munkholm Hansen	201404796

Indhold

Indhold	i
1 Indledning	1
1.1 projektformulering	1
2 Kravspecifikation	2
2.1 Aktør-Kontekst	2
2.2 Use-Cases	3
2.3 Ikke-funktionelle krav	7
3 Accepttest Specifikation	9
3.1 Test af Usecase 1	9
3.2 Test af use case 2	11
3.3 Test af Usecase 3	13
3.4 Accepttest ikke-funktionelle krav	14
4 System Arkitektur	17
4.1 System Sekvens Diagrammer	17
5 Hardware Arkitektur	19
5.1 Block Definition Diagram	19
5.2 Internal Block Diagram	21
6 Software Arkitektur	23
6.1 CPU matrix	23

Kapitel 1

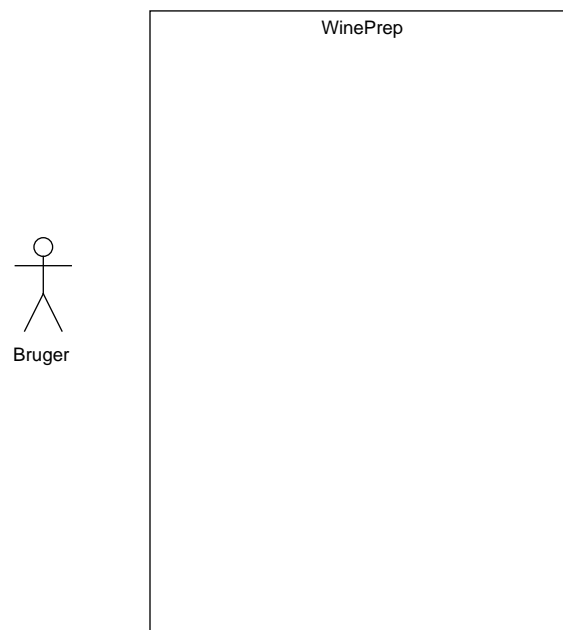
Indledning

1.1 projektformulering

Kapitel 2

Kravspekifikation

2.1 Aktør-Kontekst



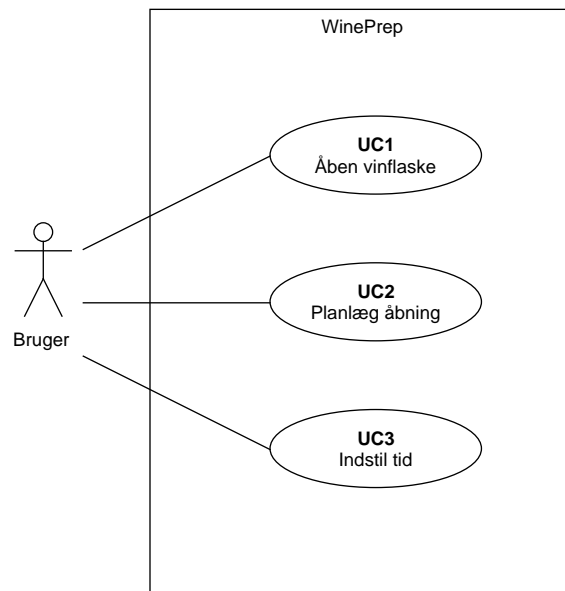
Figur 2.1: Aktør-kontekst Diagram

Aktør Beskrivelser

Bruger: Brugeren er systemets primære aktør. Brugeren er ham eller hende der betjener systemet, og har en opgave som ønskes løst af systemet.

2.2 Use-Cases

Use-case Diagram



Figur 2.2: Usecase Diagram

Use-case 1: Åbn Vin

Navn	UC 1: Åbn Vinflaske
Mål	At åbne vinflasken og dermed tillade brugeren adgang til vinen
Initiering	Bruger trykker <i>Åbn nu</i> på brugergrænsefladen
Aktører	Primær: Bruger
Antal Samtidige forekomster	1
Prækondition	Vinflasken er anbragt i maskinen og systemet er klar til brug. Desuden er vinflasken uåbnet og forseglingen er fjernet
Postkondition	Vinflasken er åbnet og proppen er fjernet
Hovedscenarie	<ol style="list-style-type: none"> 1. System detekterer vinflaskens type og position [Ext. 1: System registrerer ugyldig type af vinflaske] [Ext. 2: System kan ikke registrere en vinflaske] 2. System låser vinflasken i dens position 3. System fjerner prop fra vinflasken 4. System frigiver vinflasken 5. System meddeler brugeren om at vinflasken er åbnet og klar til brug. 6. System dispenserer prop.

Udvidelser/Undtag

- Ext.1 System registrerer ugyldig type af vinflaske
- [1.1] System meddeler brugeren om at typen af vinflaske er ugyldig.
 - [1.2] UC1 Afsluttes.
- Ext.2 System kan ikke registrere en vinflaske
- [2.1] System meddeler brugeren om at ingen vinflaske er registreret
 - [2.2] UC afsluttes

Use-case 2: Planlæg Åbning

Navn	UC 2: Planlæg Åbning
Mål	Vinen er drikkeklar til et forudbestemt tidspunkt

Initiering	Bruger trykker <i>Planlæg åbning</i> på brugergrænsefladen
Aktører	Primær: Bruger
Antal Samtidige forekomster	1
Prækondition	Vinflasken er anbragt i systemet og systemet er klar til brug. Desuden er vinflasken uåbnet og forseglingen er fjernet.
Postkondition	Vinflasken er drikkeklar til det valgte tidspunkt
Hovedscenarie	<ol style="list-style-type: none"> 1. Bruger vælger tidspunkt på systemet [Ext. 1: Bruger ønsker ikke at åbne vin] 2. Bruger bekræfter valgt tidspunkt [Ext. 2: Vinen kan ikke iltes korrekt til det valgte tidspunkt] 3. System detekterer vinflaskens type og position [Ext. 3: System registrerer ugyldig type af vinflaske] [Ext. 4: System kan ikke registrere en vinflaske] 4. System venter til iltningstidspunktet [Ext. 5: Bruger annullerer planlagt åbning af vin] 5. System detekterer vinflaskens type og position [Ext. 3: System registrerer ugyldig type af vinflaske] [Ext. 4: System kan ikke registrere en vinflaske] 6. System låser vinflasken i dens position 7. System fjerner prop fra vinflasken 8. System frigiver vinflasken 9. System dispenserer prop 10. System venter til, at vinen er drikkeklar 11. System meddeler brugeren om, at vinen er drikkeklar

Udvidelser/Undtag

Ext.1 Bruger ønsker ikke at åbne vin

[1.1a] Bruger trykker på *Tilbage*

[1.2b] UC afsluttes

Ext.2 Vinen kan ikke iltes korrekt til det valgte tidspunkt

[2.1] System beder bruger bekræfte valg af tidspunkt

[2.2a] Bruger trykker bekræft

[2.3a] UC fortsættes fra punkt 1 i UC 1

[2.2b] bruger trykker *Annuller*

[2.3b] UC afsluttes

Ext.3 System registrerer ugyldig type af vinflaske

[3.1] System meddeler brugeren om at typen af vinflaske er ugyldig.

[3.2] UC1 Afsluttes.

Ext.4 System kan ikke registrere en vinflaske

[4.1] System meddeler brugeren om at ingen vinflaske er registreret

Ext.5 Bruger annullerer planlagt åbning af vin

[5.1] Bruger trykker *STOP!*

[5.2] System beder bruger bekræfte valg

[5.3] Bruger trykker *Bekræft*

Use-case 3: Indstil tid

Navn	UC 3: Indstil tid
Mål	At indstille tiden på systemets indbyggede ur
Initiering	Bruger vælger indstillinger
Aktører	Primær: Bruger
Antal Samtidige forekomster	1
Prækondition	Bruger befinder sig i hovedmenuen
Postkondition	Tiden er indstillet korrekt
Hovedscenarie	<ol style="list-style-type: none"> 1. Bruger trykker på Indstillinger 2. Bruger trykker på dropdownmenu for timer. 3. Bruger vælger antal timer. 4. Bruger trykker på dropdownmenu for minutter. 5. Bruger vælger antal minutter. 6. System gør bekræftelsesknappen tilgængelig. 7. Bruger trykker på bekræftelsesknappen [Ext. 1:Bruger trykker på tilbageknap.] 8. System viser valgt klokkeslet på brugergrænsefladen

Udvidelser/Undtag

- Ext.1 :Bruger trykker på tilbageknap.
- [1.1] Bruger trykker på tilbageknappen.
 - [1.2] System kommer tilbage til hovedmenu.
 - [1.3]UC afsluttes.

2.3 Ikke-funktionelle krav**Brugervenlighed**

1. De virtuelle knapper på systemets grafiske brugergrænseflade skal have et areal på min 2.5 x 2.5 cm.
2. Systemet skal give brugeren besked om vinens status via tekst på touch skærmen.

Ydelse

1. Når systemet tændes, skal det kunne starte op, og være klar til modtage brugerinput på max. 2 minutter.
2. Den grafiske brugergrænsefladen skal have en reaktionstid på max 1 sek fra brugerinput via touchskærmen til opdatering af det grafiske layout.
3. Systemet skal kunne starte motorer til fastlåsning af vinen indenfor max 5 sekunder efter brugerinput "Åben nu" på brugergrænsefladen, og kunne færdiggøre åbningen af vinen indenfor max 1 minut efter brugerinput.
4. Når brugeren vælger "Planlæg åbning", skal systemet kunne åbne vinflaksen med en afvigelse på max 1 minut fra det indstillede åbningstidspunkt. Her skal åbning af vinen ligeledes kunne færdiggøres af systemet på max 1 minut.

Vedligeholdelse

1. Koden til systemet skal skrives i programmerings sprogene c og c++.
2. Systemet skal betjenes via et embedded system hvorpå en Linux platform er installeret.
3. Motor- og sensorstyring skal foregå via en PSoC.

Kapitel 3

Accepttest Specifikation

3.1 Test af Usecase 1

Tabel 3.1: Accepttestspecifikation UC1 Hovedscenarie

Use case under test		U1: Åbn Vinflaske		
Scenarie		Hovedscenarie		
Prækondition		En gyldig type vinflaske er korrekt anbragt i maskinen og systemet er klar til brug. Desuden er vinflasken uåbnet og forseglingen er fjernet		
Step	Handling	Forventet observation/resultat	Faktisk observation/resultat	Vurdering (OK/FAIL)
1	Tryk på Åbn nu på brugergrænsefladen	Vinflasken åbnes af systemet, og bruger meddeles via brugergrænsefladen om, at vinen er åbnet		

Tabel 3.2: Accepttestspecifikation UC1 Ext. 1: System registrerer ugyldig type af vinflaske

Use case under test				
U1: Åbn Vinflaske				
Scenarie				
Ext. 1: System registrerer ugyldig type af vinflaske				
Prækondition				
En gyldig type vinflaske er korrekt anbragt i maskinen og systemet er klar til brug				
Step	Handling	Forventet observation/resultat	Faktisk observation/resultat	Vurdering (OK/FAIL)
1	Tryk på Åbn nu på brugergrænsefladen	System meddeler bruger via brugergrænsefladen at vinflasken er af ugyldig type, og bruger bedes fjerne genstand fra systemet		

Tabel 3.3: Accepttestspecifikation UC1 Ext. 2: System kan ikke registrere vinflaske

Use case under test				
U1: Åbn Vinflaske				
Scenarie				
Ext. 2: System kan ikke registrere vinflaske				
Prækondition				
Systemet er klar til brug				
Step	Handling	Forventet observation/resultat	Faktisk observation/resultat	Vurdering (OK/FAIL)
1	Tryk på Åbn nu på brugergrænsefladen	System meddeler bruger via brugergrænsefladen at ingen vinflaske kan registreres		

3.2 Test af use case 2

Tabel 3.4: Accepttestspecifikation UC2 Hovedscenarie

Use case under test UC2: Planlæg Åbning				
Scenarie		Hovedscenarie		
Prækondition		En gyldig type vinflaske er korrekt anbragt i maskinen og systemet er klar til brug. Desuden er vinflasken uåbnet og forseglingen er fjernet		
Step	Handling	Forventet observation/resultat	Faktisk observation/resultat	Vurdering (OK/FAIL)
1	Tryk på <i>Planlæg åbning</i> på brugergrænsefladen	Undermenuen <i>Planlæg åbning</i> vises på brugergrænsefladen		
2	Indstil på brugergrænsefladen klokkeslættet 4 timer og 30 minutter frem	Den indstillede tid vises til det valgte klokkeslæt		
3	Tryk på <i>Bekræft</i>	Hovedmenuen vises og det valgte klokkeslæt vises i <i>Aktuel info</i> på brugergrænsefladen		
4	Vent 4 timer og 30 minutter	System meddeler bruger via brugergrænsefladen om at vinen er drikkeklar		

Tabel 3.5: Accepttestspecifikation UC2 Ext. 1: Bruger ønsker ikke at åbne vin

Use case under test U2: Planlæg Åbning				
Scenarie		Ext. 1: Bruger ønsker ikke at åbne vin		
Prækondition		Bruger befinder sig i undermenuen <i>Planlæg åbning</i>		
Step	Handling	Forventet observation/resultat	Faktisk observation/resultat	Vurdering (OK/FAIL)
1	Tryk <i>Tilbage</i> på brugergrænsefladen	Hovedmenuen vises		

Tabel 3.6: Accepttestspecifikation UC2 Ext. 2: Vinen kan ikke iltes korrekt til det valgte tidspunkt

Use case under test U2: Planlæg Åbning				
Scenarie		Ext. 2: Vinen kan ikke iltes korrekt til det valgte tidspunkt		
Prækondition		Bruger befinder sig i undermenuen <i>Planlæg åbning</i> . Bruger har valgt klokkeslæt til 10 minutter fra nuværende tidspunkt		
Step	Handling	Forventet observation/resultat	Faktisk observation/resultat	Vurdering (OK/FAIL)
1	Tryk <i>Bekræft</i> på brugergrænsefladen	System meddeler bruger om at den indstillede tid ikke er tilstrækkelig til at vinen er drikkeklar, og beder bruger genbekræfte valget.		
2	Tryk <i>Annuller</i>	Undermenuen <i>Planlæg åbning</i> vises		

Accepttests for Ext.3 og 4 er ikke medtaget her fordi de testes under accepttest for UC1: Åbn Vinflaske.

Tabel 3.7: Accepttestspecifikation UC2 Ext. 5: Bruger annullerer planlagt åbning af vin

Use case under test U2: Planlæg Åbning				
Scenarie		Ext. 5: Bruger annullerer planlagt åbning af vin		
Prækondition		Bruger har valgt klokkeslæt til 4 timer og 30 minutter fra nuværende tidspunkt og har bekræftet sig valg. Desuden er en gyldig type vinflaske korrekt placeret i maskinen		
Step	Handling	Forventet observation/resultat	Faktisk observation/resultat	Vurdering (OK/FAIL)
1	Tryk <i>STOP!</i> på brugergrænsefladen	System beder bruger via brugergrænsefladen bekræfte valg		
2	Tryk <i>Bekræft</i>	Hovedmenuen vises og <i>Aktuel info</i> er tom		

3.3 Test af Usecase 3

Tabel 3.8: Accepttestspecifikation UC3 Hovedscenarie

Use case under test U3: Indstil tid				
Scenarie Hovedscenarie				
Prækondition Bruger befinder sig i hovedmenu.				
Step	Handling	Forventet observation/resultat	Faktisk observation/resultat	Vurdering (OK/FAIL)
1	Tryk på knappen <i>Indstillinger</i> på brugergrænsefladen	Brugergrænsefladen skifter menu til indstillingsmenuen		
2	Tryk på tekstboksen med timeantal	En dropdownmenu med antal timer fra 0 til 23 dukker op.		
3	Tryk på timeantallet 23	Dropdownmenuen forsvinder og det valgte timeantal står nu i tekstboksen, og bekræftelsesknappen bliver synlig.		
4	Ændre timer til 12	Bekræftelsesknappen dukker op		
5	Ændre minutter til 30	Bekræftelsesknappen dukker op		
6	Tryk på bekræftelsesknappen	Tiden er nu skiftet til 12:30.		

Tabel 3.9: Accepttestspecifikation UC3 Ext. 1: Bruger trykker på tilbageknap

Use case under test U3: Indstil tid				
Scenarie Ext. 1: Bruger trykker på tilbageknap				
Prækondition Bruger er inde i indstillingsmenuen				
Step	Handling	Forventet observation/resultat	Faktisk observation/resultat	Vurdering (OK/FAIL)
1	Tryk på tilbageknap	Brugergrænseflade er tilbage i hovedmenu		

3.4 Accepttest ikke-funktionelle krav

Tabel 3.10: Accepttest af ikke-funktionelle krav

Hvad testes	Hvordan udføres testen	Forventet resultat	Observation/- resultat	Vurdering (OK/FAIL)
GUIs virtuelle knapper skal have areal på min. 2.5 x 2.5 cm.	Knapper måles med lineal.	Knapperne har et areal på min. 2.5 x 2.5 cm.		
Systemet skal give besked om vinen status via tekst på touch skærmen.	Efter åbning af vinen, udføres en visuel test på touch skærmen.	Der står en tekstbesked på touch skærmen om at vinen er åbnet og klar til brug.		
GUI skal have en reaktionstid på MAX. 1 sekund.	Der trykkes på GUIs virtuelle knapper og tages tid med stopur.	Der går max 1. sekund fra tryk til opdatering af det grafiske layout.		
Systemet skal starte op og være klar til at modtage brugerinput på MAX. 2 minutter.	Systemet tændes og der tages tid med stopur.	Systemet starter op og er klar til at modtage brugerinput på MAX. 2 minutter.		

Hvad skal testes	Hvordan udføres testen	Forventet resultat	Observation/- resultat	Vurdering (OK/FAIL)
Systemet skal kunne starte motorer til fastlåsning af vinen indenfor max 5 sekunder efter brugerinput "Åben nu".	Der ind-sættes en vinflaske og trykkes på knappen "åben nu", hvorefter der tages tid med stopur.	Motorer til fastlåsning af vinen starter inden for MAX. 5 sekunder.		
Systemet skal kunne færdiggøre åbningen af vinen indenfor max 1 minut efter brugerinput.	Der ind-sættes en vinflaske i systemet og trykkes på knappen "Åbn nu", hvorefter der tages tid med stop ur.	Vinen åbnes på MAX. 1 minut.		
Systemet skal kunne åbne vin-flaksen med en afvigelse på max 1 minut fra det indstillede åbnings-tidspunkt.	"Planlæg åb-ning" vælges, og der ind-tastes et åbnings-tidspunkt. Der tages tid med stopur fra åbnings-tidspunkt til åbning af vinen på-begyndes.	Der går MAX. 1 minut fra åbnings-tidspunkt til åbning af vinen på-begyndes.		

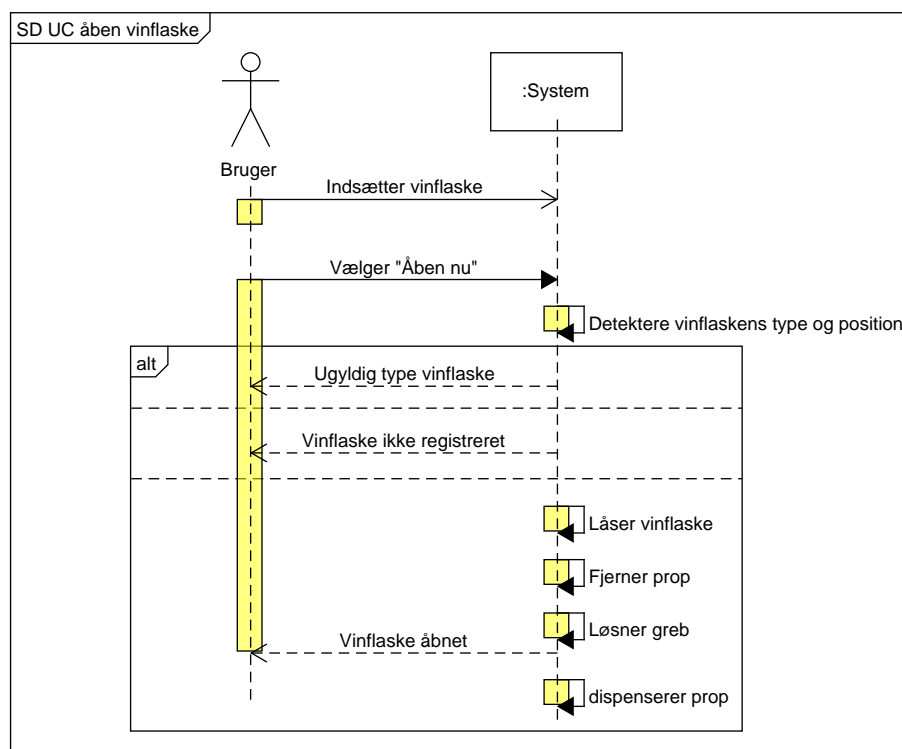
Hvad skal testes	Hvordan udføres testen	Forventet resultat	Observation/-resultat	Vurdering (OK/FAIL)
Koden til systemet skal skrives i programmerings sprogene c og c++.	Der kigges på koden til systemet.	Koden er skrevet i c/c++.		
Systemet skal betjenes via et embedded system hvorpå en Linux platform er installeret.	Visuel test af OS på embedded system.	Der er installeret en Linux platform på embedded system.		
Motor- og sensorstyring skal foregå via en PSoC.	Visuel test af systemet.	En PSoC er benyttet til Motor- og sensorstyring.		

Kapitel 4

System Arkitektur

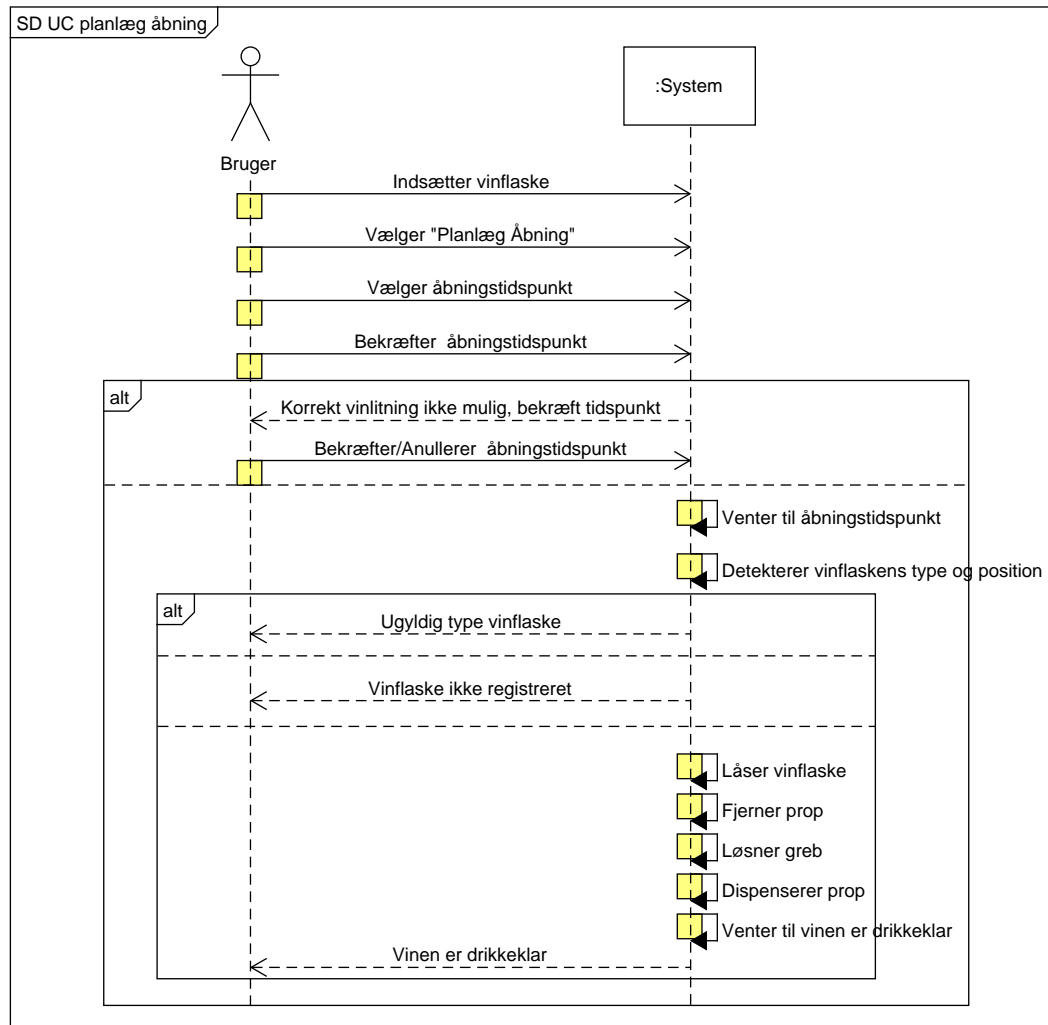
4.1 System Sekvens Diagrammer

System Sekvens Diagram for use-case 1



Figur 4.1: System Sekvens diagram for UC 1

System Sekvens Diagram for use-case 2

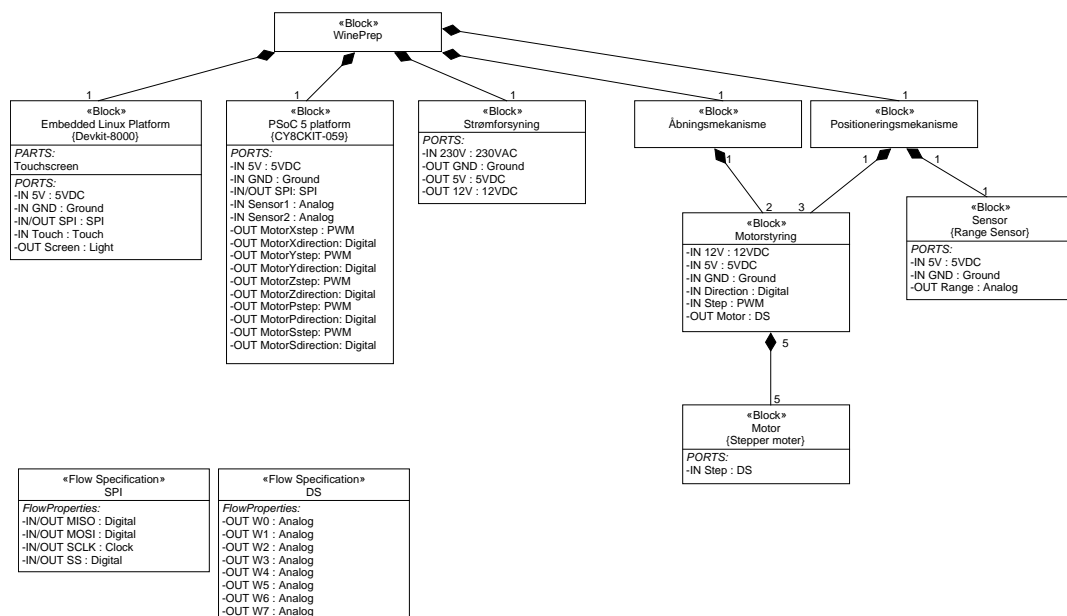


Figur 4.2: System Sekvens diagram for UC 2

Kapitel 5

Hardware Arkitektur

5.1 Block Definition Diagram



Figur 5.1: BDD for WinePrep

Blok beskrivelser

Her følger beskrivelser af de enkelte blokke på vores BDD, se side 19 Figur 5.1.

WinePrep blokken er det samlede system der består af underblokkende Embedded Linux Platform, PSoC 5 Platform, Åbningsmekanisme, Positioneringsmekanisme samt strømforsyning.

Embedded Linux Platform Dette er den blok der håndtere brugerens interaktion med systemet. Blokken består af et Devkit800 med touchskærm. Som styresystem på platformen anvendes der Linux distributionen Ångström. Her fra anvendes der QT til at lave den grafiske brugerflade der vises på touchskærmen til brugeren af systemet. Samtidig kommunikere Embedded Linux Platformen med vores PSoC 5 Platform via SPI standarden.

PSoC 5 Platform PSoC 5 baseret platform der står for styring af Motor og Sensor blokkene, samt kommunikere med blokken Embedded Linux Platform over SPI.

Strømforsyning Strømforsyning skal kunne modtage 230V fra dansk stik-kontakt, og forsyne systemet med de nødvendige spændinger.

Positioneringsmekanisme Denne blok indeholder alt hvad vi bruger til at bevæge på vores sensorer når vi scanner flasken, og til at flytte på vores åbningsmekanismer i forhold til flaskens placering. Blokken består dermed af en motorstyrings blok samt en motor blok for hver af de 3 akser.

Åbningsmekanisme Åbningsmekanismen består af de to motorer som anvendes til at skrue proptrækker-skruen i vinflaskens prop, samt til at trække proppen ud af vinflasken, samt to motorstyrings blokke til disse motorer.

Motorstyring Motorstyrings blokken består af en CY8CKIT-059, som anvendes til at styre én motor når der kommer signal fra PSOC5 platforms blokken om dette.

Sensor1 Afstandssensorer til detektering af vinflaskens placering samt størrelse, så åbningsmekanismen ud fra dette kan positioneres korrekt ved hjælp af motorer på X, Y , Z akserne.

Sensor2 Sensor til at detekterer når en akse kommer til et yderpunkt.

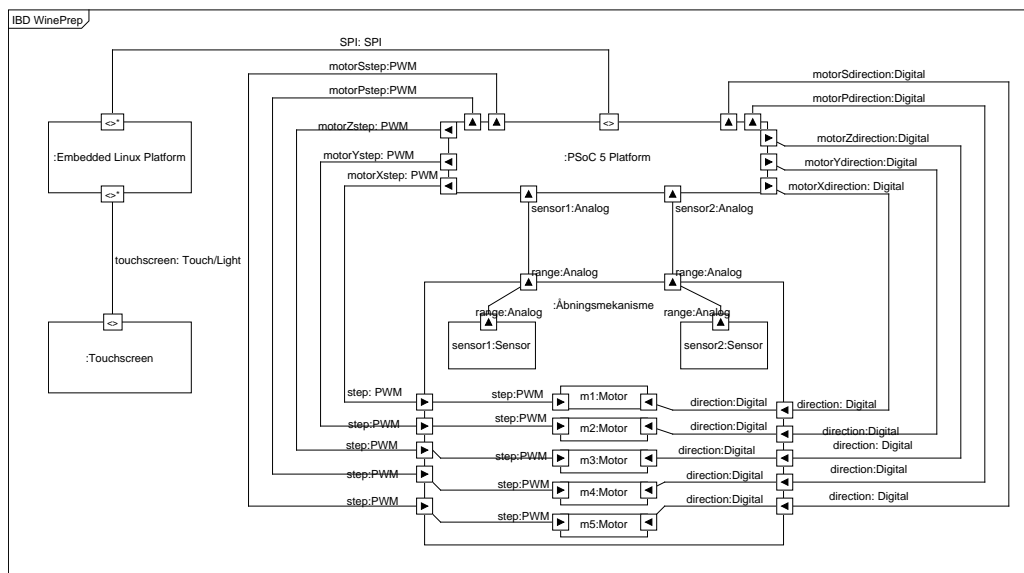
Motor Motorblokken er alle de motorer som anvendes i systemet til positionering og prop-træk. Denne blok skal eventuelt opdeles i flere forskellige blokke hvis vi får brug for at anvende andre typer motorer end steppermotorer.

Ting der først bliver fast besluttet på senere iterationer/sprints

Motor Valg er ikke 100% fastlagt, hvorfor det her i BDD modelleres med stepper motors, og portene er derfor heller ikke 100% korrekte da dette afhænger af motorstyringen.

Sensor typer og antal ligger kun delvist fast. Der vil være 2 afstandssensorer til detektering af vinflaskens placering, samt 3 sensorer af ikke nærmerer fastlagt type til at detekterer hvis en akse når til et yderpunkt. Afstands sensorer til detektering af vinflaskens position, bliver enten lys baserede eller lydbase-rede, der vil give et analog output signal i form af en spænding der afhænger af afstanden. Sensorer til detektering på aksernes yderpunkter overvejes implementeret med en switch, eller eventuelt strain gauge.

5.2 Internal Block Diagram



Figur 5.2: IBD for WinePrep

Signal Beskrivelser

Signal Type	Porte	Beskrivelse
Digital	MISO, MOSI, SS, MotorXDirection, MotorYDirection, MotorZDirection, MotorSDirection, MotorPDirection	0-5V firkant signal
Analog	Range, Sensor1, Sensor2	Analog Spænding mellem 0-5V
Clock	SCLK	Konstant firkantsignal på 0-5V med 50% dutycycle og fast frekvens
Touch	Touchscreen	kraftpåvirkning af skærmen
Light	Touchscreen	Lys i varierende farver i det synlige spektrum
PWM	Step, MotorXstep, MotorYstep, MotorZstep, MotorSstep, MotorPstep	0-5V firkant med varierende duty-cycle.
SPI	SPI	Serial Peripheral Interface Bus industri standard

Kapitel 6

Software Arkitektur

6.1 CPU matrix

Til brug for software arkitekturen er der udarbejdet en CPU Matrix som ses på tabel 6.1 side 23. Denne giver et overblik over hvilke CPUer der indgår i de enkelte usecases. Ud fra dette er der udarbejdet applikationsmodeller for de enkelte CPUer i systemet.

Tabel 6.1: CPU matrix

	PSoC5	Devkit8000
<i>UC1</i>	X	X
<i>UC2</i>	X	X
<i>UC3</i>		X