

# Dokumentation

## Semesterprojekt 3. Semester

### Gruppe 10

Vejleder: Søren Hansen

Gruppemedlemmer:

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

## Versions Historik

Version	Dato	Ændringer	Forfatter
1.0	02-10-2016	Første udkast efter sprint 2	Hele gruppen
1.1	09-10-2016	Udgave sendt til review, Applikationsmodeller tilføjet, layout rettet til mm.	Hele gruppen
1.2	30-10-2016	Rettelser skrevet ind efter review, Usecases updaterede, nye sekvensdiagrammer for PSoC, tabel layout rettet til efter råd fra review gruppen	Hele gruppen.

# Indhold

<b>Versions Historik</b>	<b>i</b>
<b>Indhold</b>	<b>ii</b>
<b>1 Indledning</b>	<b>1</b>
1.1 Projektformulering . . . . .	1
<b>2 Kravspecifikation</b>	<b>3</b>
2.1 Usecase Diagram . . . . .	3
2.2 Use-Cases . . . . .	4
2.3 Ikke-funktionelle krav . . . . .	11
<b>3 Accepttest Specifikation</b>	<b>12</b>
3.1 Test af Use-case 1 . . . . .	12
3.2 Test af use-case 2 . . . . .	14
3.3 Test af Use-case 3 . . . . .	17
3.4 Test ikke-funktionelle krav . . . . .	18
<b>4 System Arkitektur</b>	<b>22</b>
4.1 System Sekvens Diagrammer . . . . .	22
<b>5 Hardware Arkitektur</b>	<b>24</b>
5.1 Block Definition Diagram . . . . .	24
5.2 Internal Block Diagram . . . . .	27
<b>6 Software Arkitektur</b>	<b>29</b>
6.1 CPU matrix . . . . .	29
6.2 UC1: Åbn vinflaske . . . . .	30
6.3 UC2: Planlæg åbning . . . . .	34
6.4 UC3: Indstil tid . . . . .	38

# Kapitel 1

## Indledning

### 1.1 Projektformulering

Mange ældre har i dag svært ved at åbne deres vinflaske, da de ikke har den fornødne styrke til selv at trække korkproppen ud af vinflasken. Derfor ville det være ideelt for dem, at have en løsning hvor åbningen af vinflaskerne bliver automatiseret.

For at få den optimale oplevelse ud af en vin, skal den åbnes rettidigt så den iltes før indtagelse. Iltningstiden kan variere fra vin til vin, og derfor kan mange uerfarne vindrikkere have svært ved at ilte deres vin korrekt. Mange glemmer at åbne vinen i god tid, og opnår derfor ikke den optimale oplevelse. Det kan derfor være ideelt, hvis denne proces også automatiseres.

#### Projektdefinition

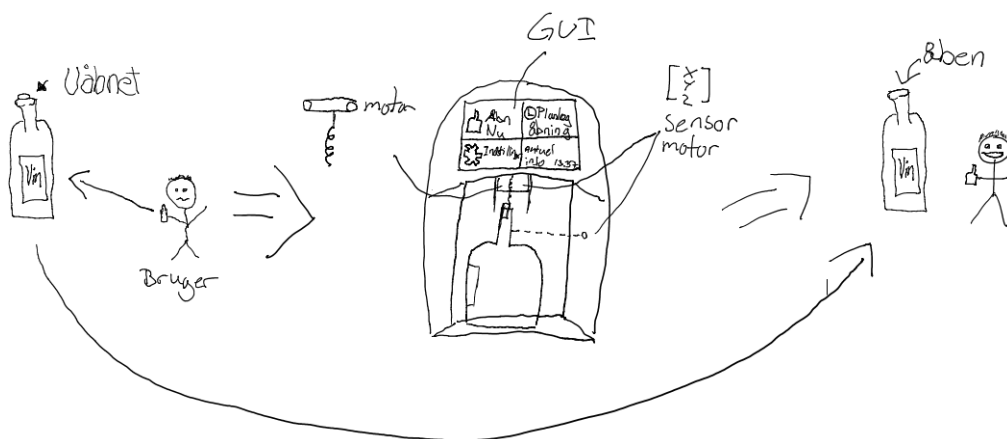
Ud fra ovenstående problemstillinger skal der fremstilles et produkt som skal automatisere vinåbning og iltning, samt hvis muligt oprette et netværk hvor vindrikkere og forhandlere kan sættes i forbindelse med hinanden. Produktet har følgende systemkrav:

Systemet:

- Skal trække korkproppen ud af en vinflaske, og ilte vinen korrekt (2 timers iltning). I denne proces indgår aktuator, sensor, PSoC, og Devkit8000.
- Skal betjenes via Devkit 8000, hvor der er installeret Angstrom Linux.
- Skal meddele brugeren når vinen er klar.
- Skal kunne indstilles til at trække korkproppen ud til et givent tidspunkt.
- Skal holde vinflasken fast under udtrækningen af korkproppen.
- Skal detektere afstanden fra toppen af flasken til åbningsmekanismen. Skal kunne dispensere vinpropper i beholder efter åbning.

- Skal kunne give status for iltningproces.
- Skal kunne åbne vin med det samme, når brugeren vælger denne.
- Skal have et grafisk brugerinterface til betjening af vinåbningen.
- Skal give brugeren mulighed for at indstille klokken på et indbygget realtidsur.
- Skal have et grafisk brugerinterface til betjening af vinåbningen.
- Bør have en sikkerhedsmekanisme til forebyggelse af personskader.
- Bør ud fra vinens type kunne ilte vinen korrekt.
- Kunne hvis muligt måle vinflaskens temperatur.
- Kunne hvis muligt fjernbetjenes med en mobil applikation, således at brugeren har mulighed for at ændre på et evt. åbningstidspunkt.
- Kunne hvis muligt tilkobles en database med information om forskellige vine og deres iltningstid, så det er muligt at automatisere iltningsprocessen ud fra de enkelte vine.
- Kunne hvis muligt forbindes til et online socialt netværk, så vindrikkere kan give anmeldelser af forskellige vine.
- Kunne hvis muligt give bruger mulighed for at bestille vine direkte fra en forhandler.

Figur 1.1: Rigt billede der beskriver wineprep

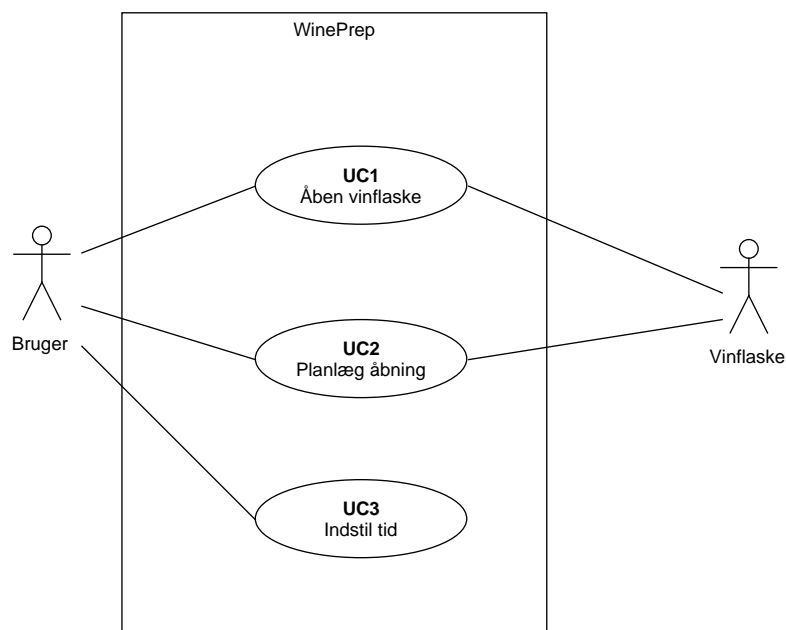


## Kapitel 2

# Kravspecifikation

### 2.1 Usecase Diagram

Her ses aktør usecase diagrammet for det realistiske system, dette erstatter aktør-kontekst diagrammet som er så simpelt at det ikke er nødvendigt:



Figur 2.1: Use-case 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 1: Åben Vinflaske

Navn	Use-case 1: Åben 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"><li>1. Bruger trykker <i>Åbn nu</i> på brugergrænsefladen</li><li>2. System detekterer vinflaskens type og position<ul style="list-style-type: none"><li>[Ext. 1: System registrerer ugyldig type af vinflaske]</li><li>[Ext. 2: System kan ikke registrere en vinflaske]</li></ul></li><li>3. System låser vinflasken i dens position</li><li>4. System fjerner prop fra vinflasken</li><li>5. System frigiver vinflasken</li><li>6. System meddeler brugeren om at vinflasken er åbnet og klar til brug.</li><li>7. System dispenserer prop.</li></ol>

**Udvidelser/Undtagelser**

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] UC1 afsluttes



**Use-case 2: Planlæg Åbning**

<b>Navn</b>	<b>Use-case 2: Planlæg Åbning</b>
<b>Mål</b>	Vinen er drikkeklar til et forudbestemt tidspunkt
<b>Initiering</b>	Bruger trykker Planlæg åbning-knappen på brugergrænsefladen
<b>Aktører</b>	Primær: Bruger
<b>Antal Samtidige forekomster</b>	1
<b>Prækondition</b>	Vinflasken er anbragt i systemet og systemet er klar til brug. Desuden er vinflasken uåbnet og forseglingen er fjernet.
<b>Postkondition</b>	Vinflasken er drikkeklar til det valgte tidspunkt
<b>Hovedscenarie</b>	<ol style="list-style-type: none"><li>1. Bruger trykker Planlæg åbning-knappen på brugergrænsefladen</li><li>2. Bruger vælger tidspunkt på systemet [Ext. 1: Bruger ønsker ikke at åbne vin]</li><li>3. Bruger bekræfter valgt tidspunkt [Ext. 2: Vinen kan ikke iltes korrekt til det valgte tidspunkt]</li><li>4. System venter til iltnings-tidspunktet [Ext. 3: Bruger annullerer planlagt åbning af vin]</li><li>5. fortsæt med Usecase 1</li></ol>

**Udvidelser/Undtagelser**

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 Bruger annullerer planlagt åbning af vin

[3.1] Bruger trykker stop

[3.2] System beder bruger bekræfte valg

[3.3] 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 på brugergrænsefladen
Postkondition	Tiden er indstillet korrekt

**Hovedscenarie**

1. Bruger trykker på indstillinger
2. Brugergrænsefladen skifter til indstillingsmenu.
3. Bruger trykker på dropdownmenu for timer.  
[Ext. 1: Bruger trykker på tilbageknap.]
4. Bruger vælger antal timer.  
[Ext. 1: Bruger trykker på tilbageknap.]
5. System gør bekræftelsesknappen tilgængelig.
6. Bruger trykker på dropdownmenu for minutter.  
[Ext. 2: Bruger trykker på bekræftelsesknap]  
[Ext. 1: Bruger trykker på tilbageknap.]
7. Bruger vælger antal minutter.
8. Bruger trykker på bekræftelsesknappen  
[Ext. 1: Bruger trykker på tilbageknap.]
9. System viser valgt klokkeslet på brugergrænsefladen

**Udvidelser/Undtagelser**

Ext.1 :Bruger trykker på tilbageknap.

[1.1] Bruger trykker på tilbageknappen.

[1.2] System kommer tilbage til hovedmenu.

[1.3]UC afsluttes.

Ext.1 :Bruger trykker på tilbageknap.

[1.1] Bruger trykker på tilbageknappen.

[1.2] System kommer tilbage til hovedmenu.

[1.3]UC afsluttes.

Ext. 2 : Bruger trykker på bekræftelsesknep.

: Spring til punkt 8 i UC 3.

## 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.

### Ydeevne

1. Når systemet tændes, skal det kunne starte op, og være klar til modtage brugersinput på max. 2 minutter.
2. Den grafiske brugergrænsefladen skal have en reaktionstid på max 1 sek fra brugersinput via touchskærmen til opdatering af det grafiske layout.
3. Systemet skal kunne starte motorer til fastlåsning af vinen indenfor max 5 sekunder efter brugersinput "Åben nu" på brugergrænsefladen, og kunne færdiggøre åbningen af vinen indenfor max 1 minut efter brugersinput.
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.
5. Systemet skal kunne håndtere en vinflaske af typen x

### 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 Use-case 1

Tabel 3.1: Accepttestspecifikation Use-case 1: Åben Vinflaske: Hovedscenarie

Use-case under test		Use-case 1: Åben 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 låses og åbnes af systemet, og bruger meddeles via brugergrænsefladen om, at vinen er åbnet		

Tabel 3.2: Accepttestspecifikation Use-case 1: Åben Vinflaske Ext. 1: System registrerer ugyldig type af vinflaske

Use-case under test		Use-case 1: Åben 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 Åbnu på brugergrænsefladen	System meddeler bruger via brugergrænsefladen at vinflasken er af ugyldig type.		

Tabel 3.3: Accepttestspecifikation Use-case 1: Åben Vinflaske: Ext. 2: System kan ikke registrere vinflaske

Use-case under test		Use-case 1: Åben 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 Åbnu på brugergrænsefladen	System meddeler bruger via brugergrænsefladen at ingen vinflaske kan registreres		



## 3.2 Test af use-case 2

Tabel 3.4: Accepttestspecifikation Use-case 2: Planlæg Åbning Hovedscenarie

Use-case under test			Use-case 2: 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 planlæg åbning på brugergrænsefladen	Undermenuen planlæg åbning vises på brugergrænsefladen		
2	Indstil på brugergrænsefladen klokkeslættet 2 timer frem	Den indstillede tid vises til det valgte klokkeslæt		
3	Tryk på bekræft	Hovedmenuen vises og det valgte klokkeslæt vises i aktuel info på brugergrænsefladen		
4	Vent 2 timer	Vinflasken er åben og systemet meddeler bruger via brugergrænsefladen om at vinen er drikkeklar		

Tabel 3.5: Accepttestspecifikation Use-case 2: Planlæg Åbning Ext. 1: Bruger ønsker ikke at åbne vin

Use-case under test			Use-case 2: Planlæg Åbning	
Scenarie			Ext. 1: Bruger ønsker ikke at åbne vin	
Prækondition			UI viser planlæg åbning	
Step	Handling	Forventet observation/resultat	Faktisk observation/resultat	Vurdering (OK/FAIL)
1	Tryk tilbage på brugergrænsefladen	Hovedmenuen vises		

Tabel 3.6: Accepttestspecifikation Use-case 2: Planlæg Åbning Ext. 2: Vinen kan ikke iltes korrekt til det valgte tidspunkt

Use-case under test			Use-case 2: Planlæg Åbning	
Scenarie			Ext. 2: Vinen kan ikke iltes korrekt til det valgte tidspunkt	
Prækondition			Bruger befinder sig i undermenuen planlæg åbning. Bruger har valgt klokkeslæt til 10 minutter fra nuværende tidspunkt	
Step	Handling	Forventet observation/resultat	Faktisk observation/resultat	Vurdering (OK/FAIL)
1	Tryk bekræft 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 annuller	Undermenuen planlæg åbning vises		

Accepttests for Ext.3 og 4 er ikke medtaget her fordi de testes under accepttest for Use-case 1: Åben Vinflaske.

Tabel 3.7: Accepttestspecifikation Use-case 2: Planlæg Åbning Ext. 5: Bruger annullerer planlagt åbning af vin

<b>Use-case under test</b>		Use-case 2: Planlæg Åbning		
<b>Scenarie</b>		Ext. 5: Bruger annullerer planlagt åbning af vin		
<b>Prækondition</b>		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 stop på brugergrænsefladen	System be-der bruger via brugergrænsefladen bekræfte valg		
2	Tryk bekræft	Hovedmenuen vises		

### 3.3 Test af Use-case 3

Tabel 3.8: Accepttestspecifikation ?? Hovedscenarie

Use-case under test			??	
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 dropdown-menu med antal timer fra 0 til 23 dukker op.		
3	Tryk på timeantallet 12	Dropdownmenu forsvinder og det valgte timeantal står nu i tekstboksen, og bekræftelsesknappen bliver synlig.		
4	Ændre minutter til 30	Bekræftelsesknappen dukker op		
5	Tryk på bekræftelsesknappen	Tiden er nu skiftet til 12:30.		

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

Use-case under test		??		
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	Brugergrænseflæ er tilbage i hovedmenu		

### 3.4 Test ikke-funktionelle krav

Tabel 3.10: Test af ikke-funktionelle krav

Hvad skal 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 vinens 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.		

Hvad skal testes	Hvordan udføres testen	Forventet resultat	Observation/- resultat	Vurdering (OK/FAIL)
GUI skal have en reaktionstid på MAX. 1 sekund.	Der trykkes på GUIs virtuelle knapper og filmes således der bagefter på baggrund af klippet kan konkluderes om der er gået et sekund.	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.		
Systemet skal kunne starte motorer til fastlåsning af vinen indenfor max 5 sekunder efter brugerinput "Åben nu".	Der indstættes 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.		

Hvad skal testes	Hvordan udføres testen	Forventet resultat	Observation/-resultat	Vurdering (OK/FAIL)
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 vinflaksen med en afvigelse på max 1 minut fra det indstillede åbnings-tidspunkt.	"Planlæg åbning" vælges, og der ind-tastes et åbnings-tidspunkt. Der tages tid med stopur fra åbnings-tidspunkt til åbning af vinen påbegyndes.	Der går MAX. 1 minut fra åbnings-tidspunkt til åbning af vinen påbegyndes.		
Koden til systemet skal skrives i programmerings sprogene c og c++.	Koden kompiletes med en c/c++ kompiler	Kompilering af koden er succesfuld.		

Hvad skal testes	Hvordan udføres testen	Forventet resultat	Observation/-resultat	Vurdering (OK/FAIL)
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.		

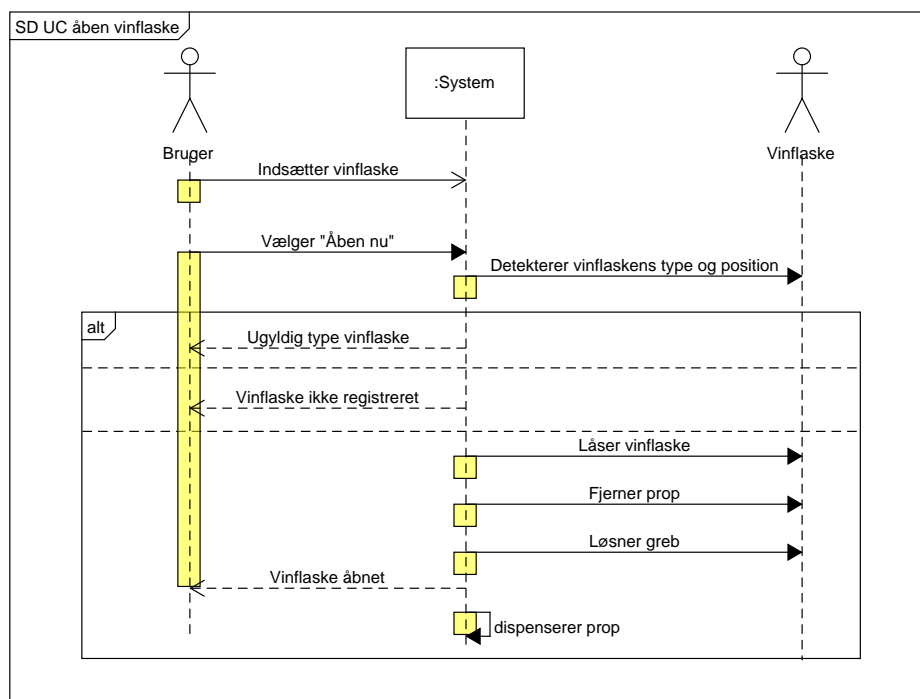


## Kapitel 4

# System Arkitektur

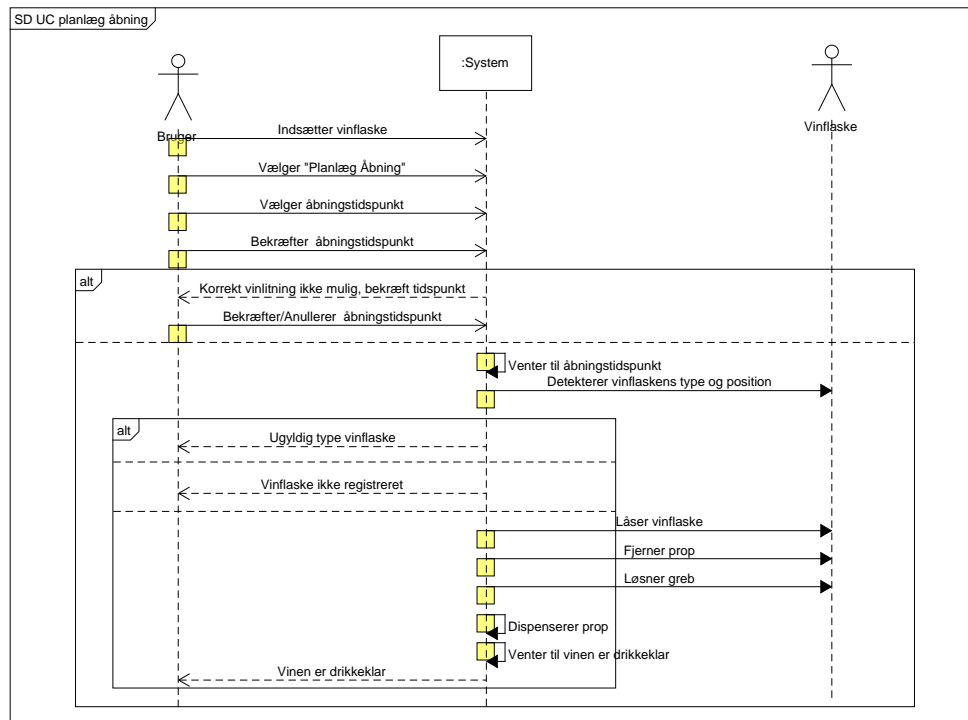
### 4.1 System Sekvens Diagrammer

#### System Sekvens Diagram for use-case 1



Figur 4.1: System Sekvens diagram for UC 1

## System Sekvens Diagram for use-case 2



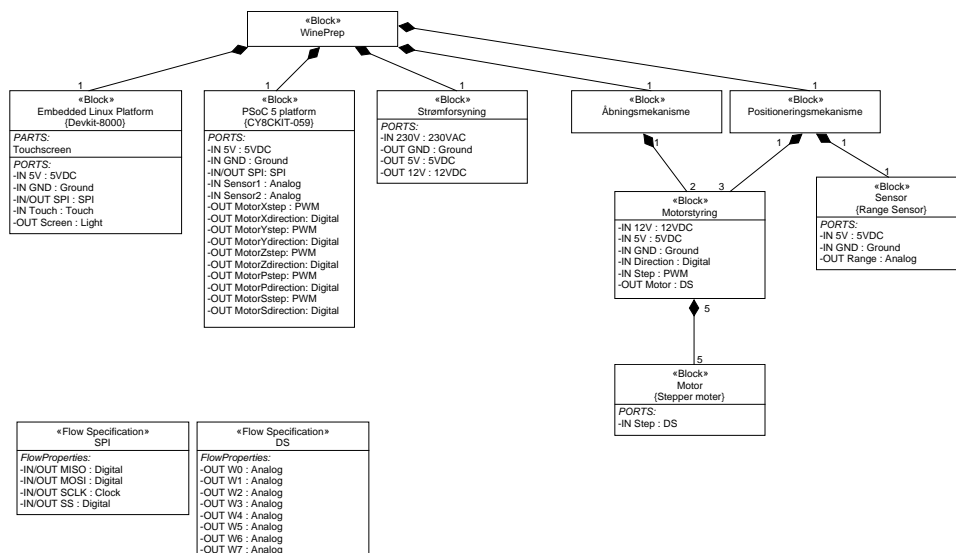
Figur 4.2: System Sekvens diagram for UC 2

## Kapitel 5

# Hardware Arkitektur

### 5.1 Block Definition Diagram

Vores system kaldet WinePrep, består af en embedded Linux-platform (Devkit-8000), hvor der er mulighed for bruger-input. Linux-plattformen er forbundet til en PSoC5 (CY8CKIT-059) via SPI. PSoC platformen anvendes til at styre positionerings- og åbnings-mekanismerne, som hver består af nogle aktuatorer og sensorer. Positioneringsmekanismen består af de 3 akser (X,Y,Z) og steppermotorer til at styre disse, åbningsmekanismen er så fastmonteret herpå, således denne kan positioneres i forhold til vinflasken, så aktuatorer på åbningsmekanismen kan anvendes til at trække proppen.



Figur 5.1: BDD for WinePrep

## Blok beskrivelser

Her følger beskrivelser af de enkelte blokke på vores BDD, se side 24 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 åbningsmekaniske 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 styrer 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. Anvendes ved at kører aksen ud indtil sensoren aktiveres, og så indstille aksens position til en forud fastlagt værdi.

**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.



**Signal Beskrivelser**

<b>Signal Type</b>	<b>Porte</b>	<b>Beskrivelse</b>
<b>Digital</b>	MISO, MOSI, SS, MotorXDirection, MotorYDirection, MotorZDirection, MotorSDirection, MotorPDirection	0-5V firkant signal
<b>Analog</b>	Range, Sensor1, Sensor2	Analog Spænding mellem 0-5V
<b>Clock</b>	SCLK	Konstant firkantsignal på 0-5V med 50% dutycycle og fast frekvens
<b>Touch</b>	Touchscreen	kraftpåvirkning af skærmen
<b>Light</b>	Touchscreen	Lys i varierende farver i det synlige spektrum
<b>PWM</b>	Step, MotorXstep, MotorYstep, MotorZstep, MotorSstep, MotorPstep	0-5V firkant med varierende dutycycle.
<b>SPI</b>	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 29. 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

Alle diagrammerne for PSoC5 i dette kapitel er vedlagt som bilag i zip filen i mappen APPSoC, da de kan være meget svære at læse efter, at de er blevet skaleret ned til at passe i PDF filen. Her kan der ligeledes findes et sekvensdiagram for hver af de 2 use-cases, hvor der ikke benyttes ref-blokke til at simplificere dem.

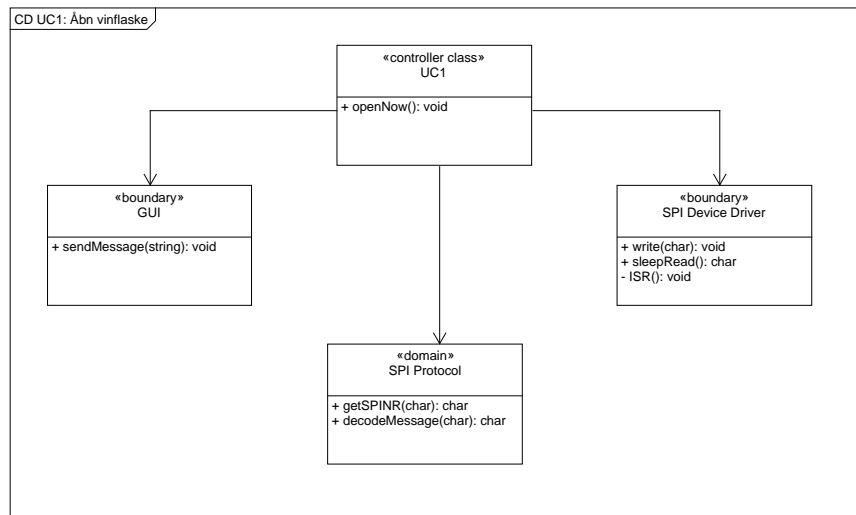
Da sekvensdiagrammerne for PSoC 5 er meget store, er de opdelt i mindre bidder for at gøre de overordnede diagrammer mere overskuelige. Diagrammerne, der refereres til i sekvensdiagrammerne for UC1 og UC2, kan ses på side 40 ff.



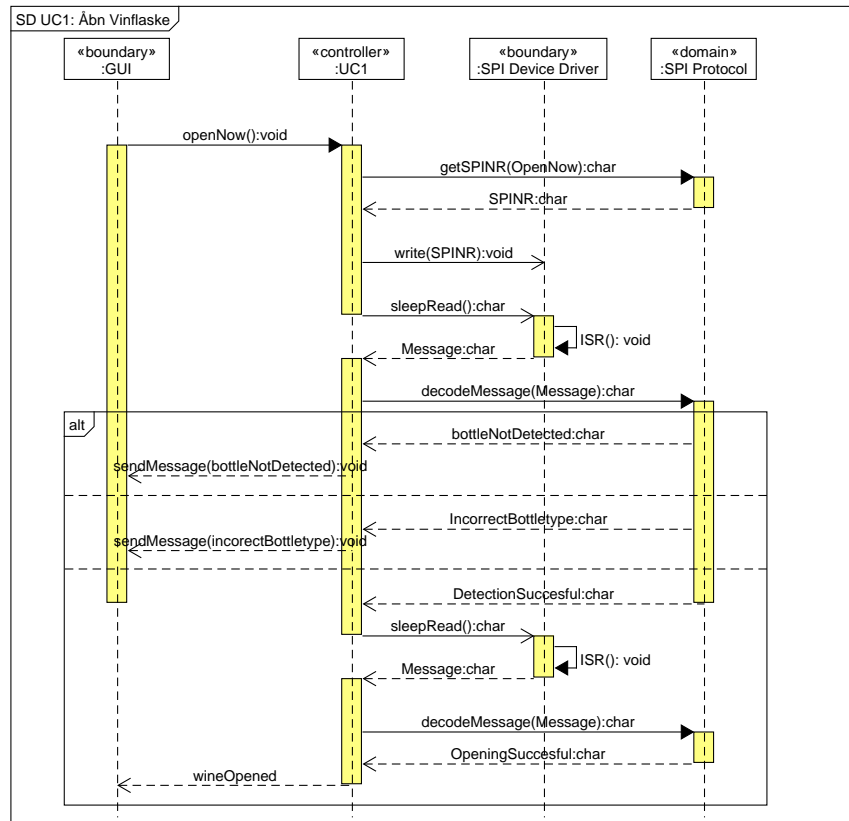
## 6.2 UC1: Åbn vinflaske

Linux Platform / DevKit8000

Figur 6.1: Klassediagram Use-case 1: Åben Vinflaske på DevKit8000

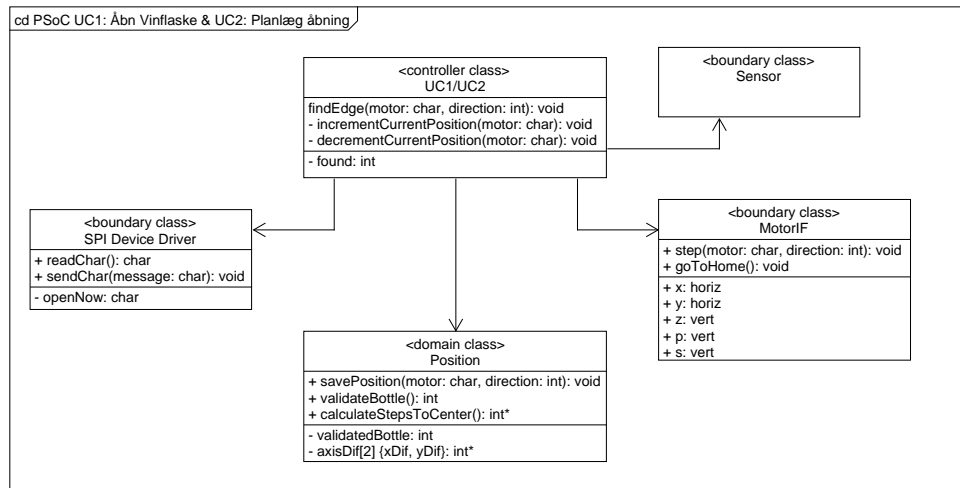


Figur 6.2: Sekvensdiagram Use-case 1: Åben Vinflaske på DevKit8000

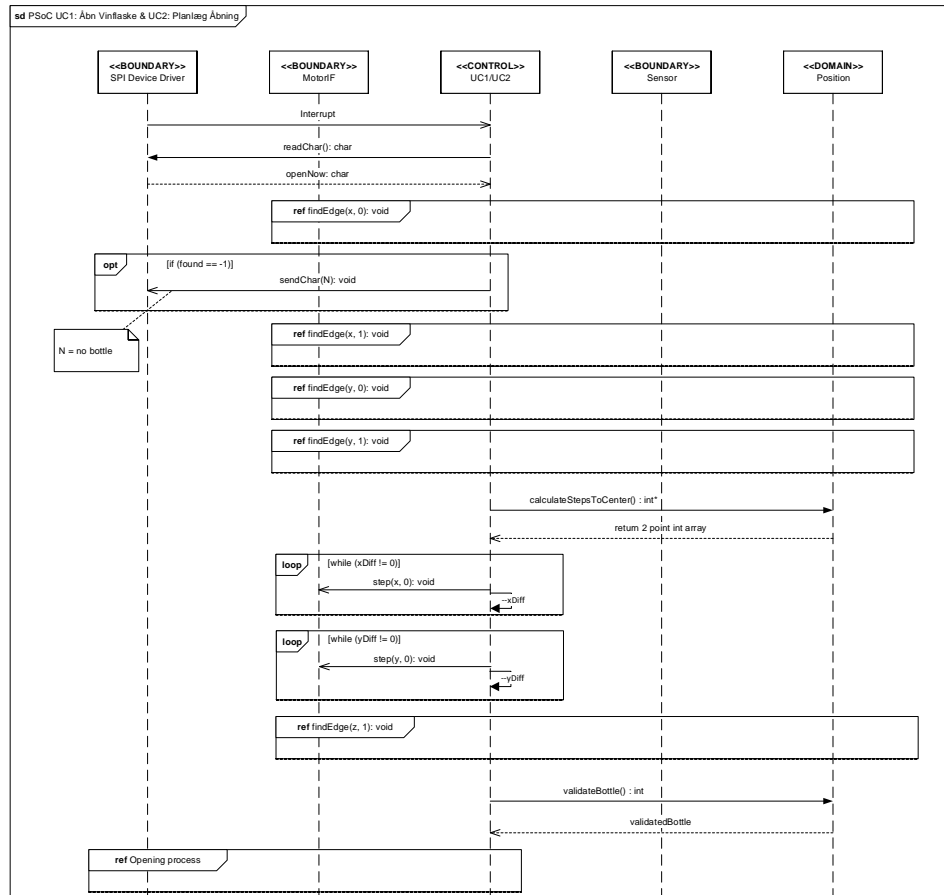


## PSoC 5

Figur 6.3: Klassediagram Use-case 1: Åben Vinflaske på PSoC 5



Figur 6.4: Sekvensdiagram Use-case 1: Åben Vinflaske på PSoC 5

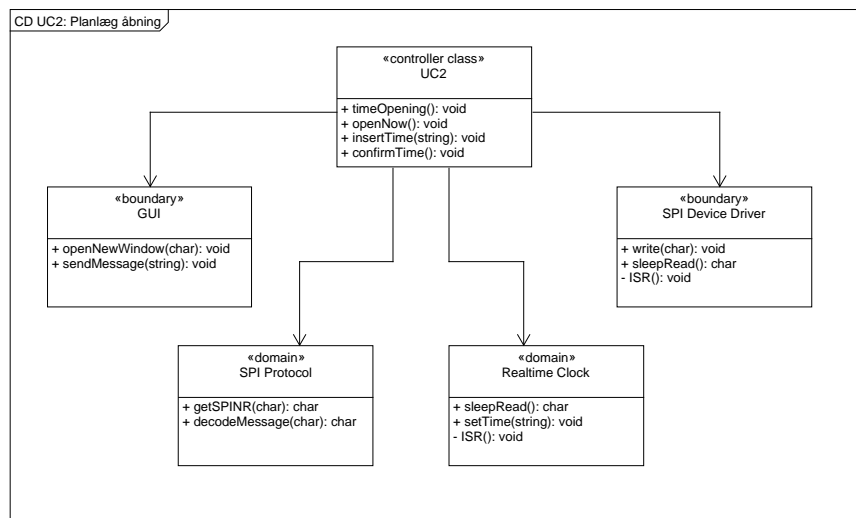


## 6.3 UC2: Planlæg åbning

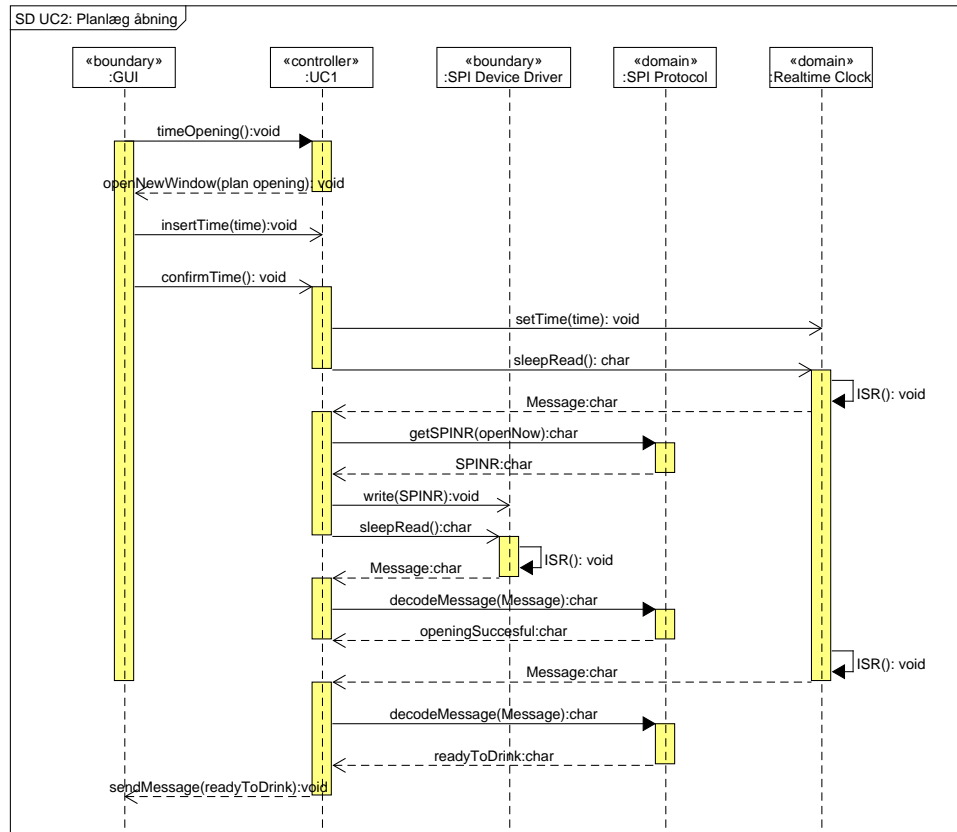
### Linux Platform / DevKit8000

Enkelte extensions er udeladt på sekvensdiagrammet, da de blot resulterer i en terminering af use-case-sekvensen.

Figur 6.5: Klassediagram Use-case 2: Planlæg Åbning på DevKit8000

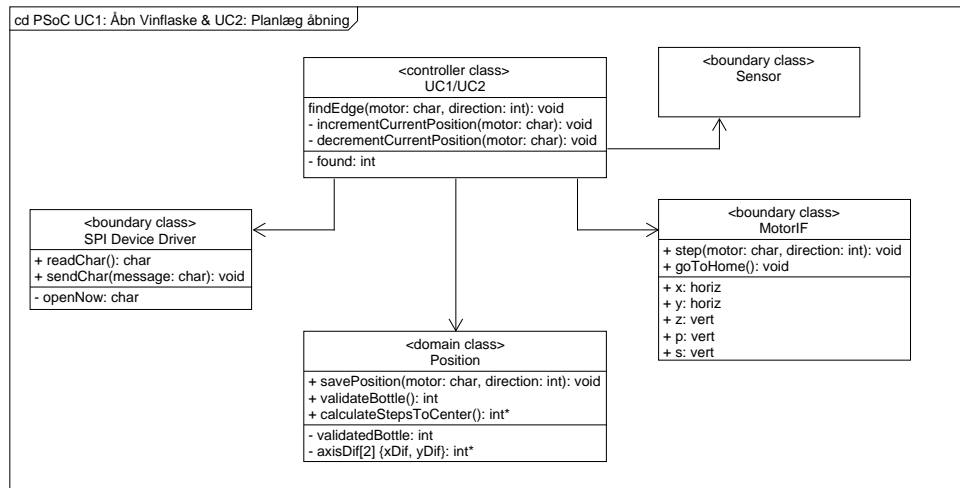


Figur 6.6: Sekvensdiagram Use-case 2: Planlæg Åbning på DevKit8000

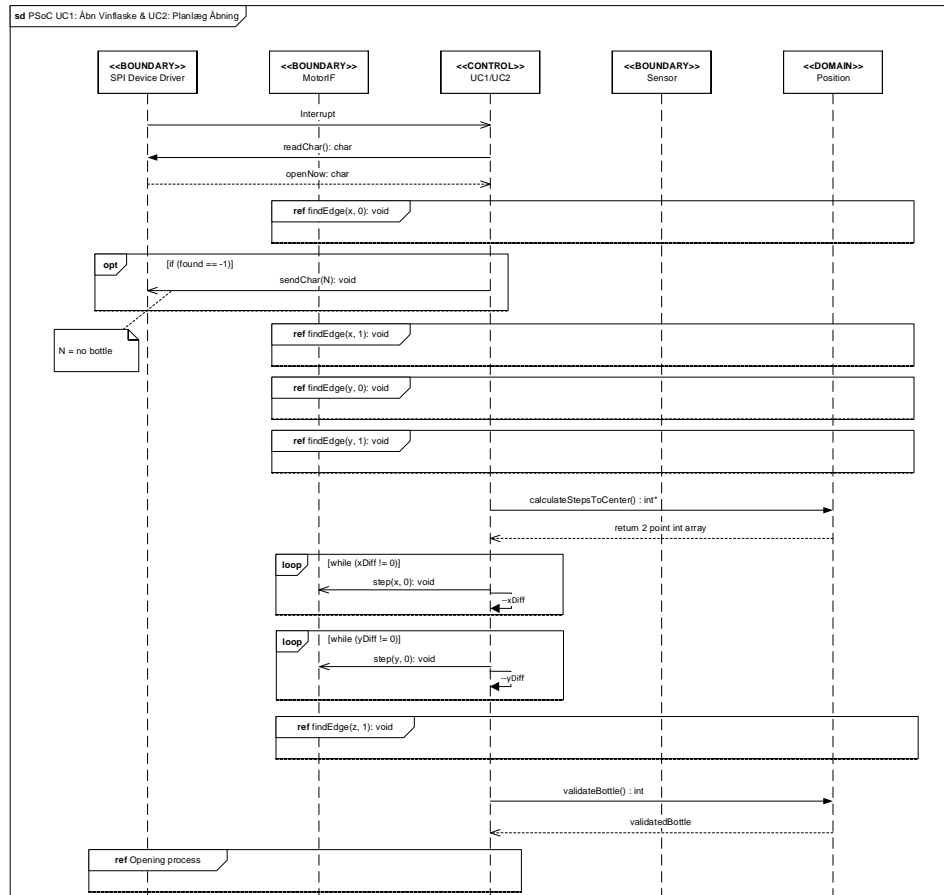


## PSoC 5

Figur 6.7: Klassesdiagram Use-case 2: Planlæg Åbning på PSoC 5



Figur 6.8: Sekvensdiagram Use-case 2: Planlæg Åbning på PSoC 5

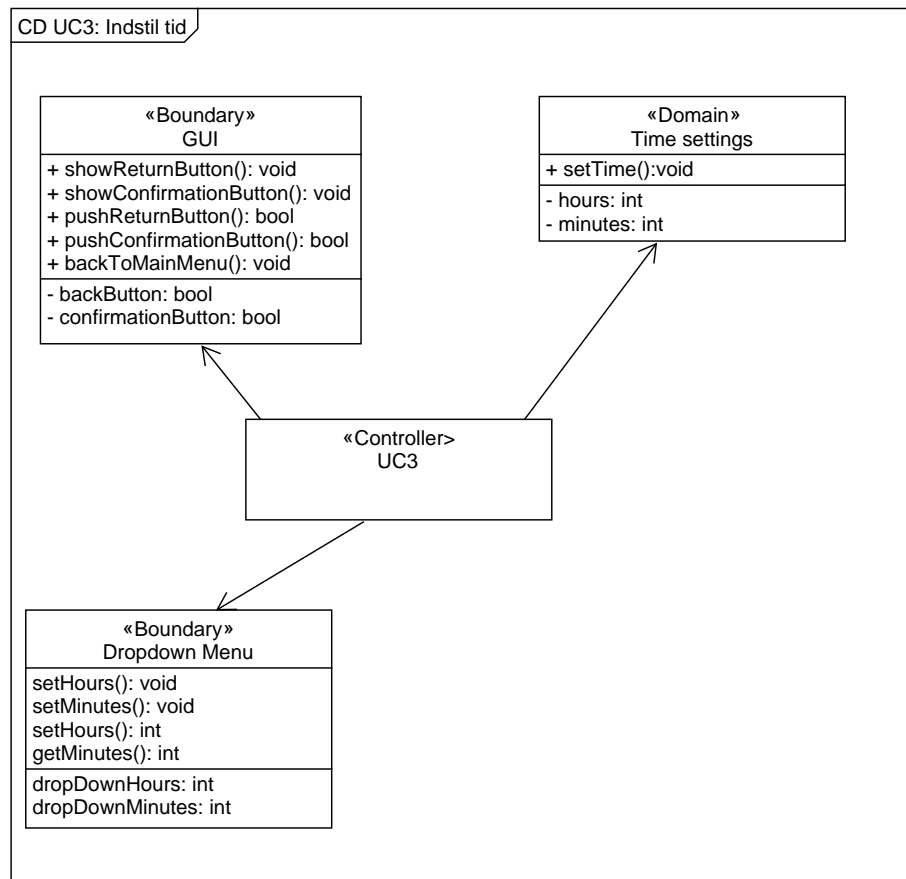




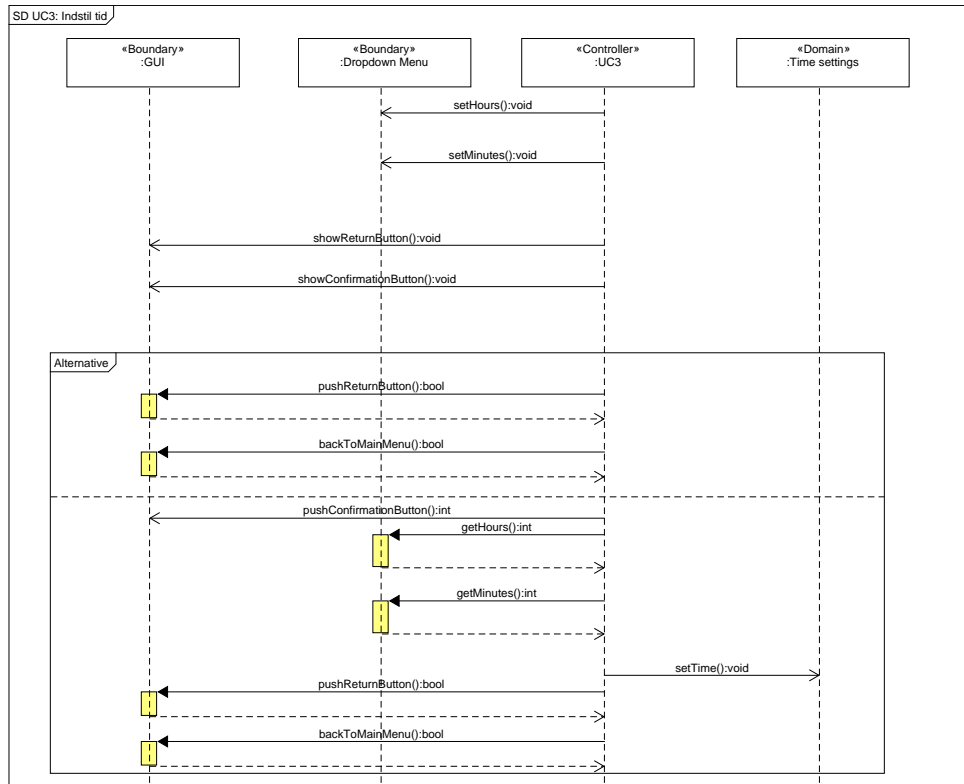
## 6.4 UC3: Indstil tid

Linux Platform / DevKit8000

Figur 6.9: Klassediagram ?? på DevKit8000

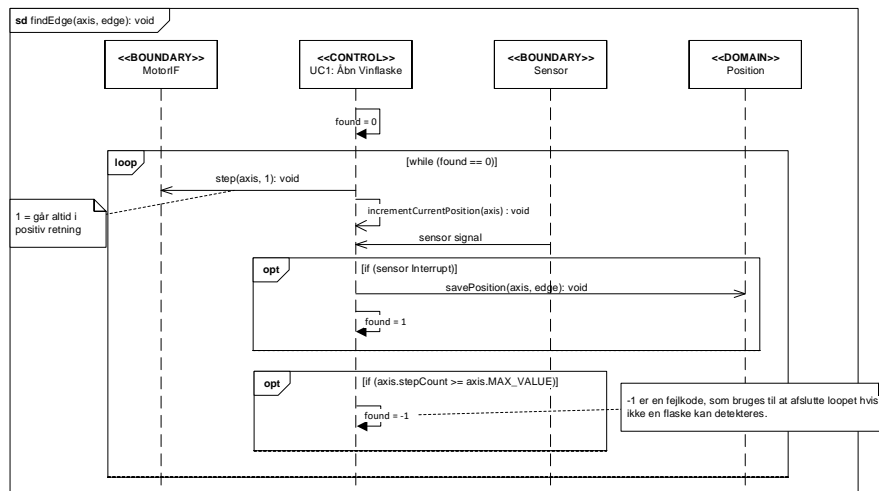


Figur 6.10: Sekvensdiagram ?? på DevKit8000



## Sekvensdiagrammer der anvendes i applikationsmodel for PSoC 5

Figur 6.11: Sekvensdiagram for funktionen findEdge



Figur 6.12: Sekvensdiagram for funktionen opening process

