

# 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

<b>Indhold</b>	<b>i</b>
<b>1 Indledning</b>	<b>1</b>
1.1 Projektformulering . . . . .	1
1.2 Risiko Analyse . . . . .	3
<b>2 Kravspecifikation</b>	<b>4</b>
2.1 Aktør-Kontekst . . . . .	4
2.2 Use-Cases . . . . .	5
2.3 Ikke-funktionelle krav . . . . .	9
<b>3 Accepttest Specifikation</b>	<b>11</b>
3.1 Test af Usecase 1 . . . . .	11
3.2 Test af use case 2 . . . . .	13
3.3 Test af Usecase 3 . . . . .	15
3.4 Accepttest ikke-funktionelle krav . . . . .	16
<b>4 System Arkitektur</b>	<b>19</b>
4.1 System Sekvens Diagrammer . . . . .	19
<b>5 Hardware Arkitektur</b>	<b>21</b>
5.1 Block Definition Diagram . . . . .	21
5.2 Internal Block Diagram . . . . .	23
<b>6 Software Arkitektur</b>	<b>25</b>
6.1 CPU matrix . . . . .	25
6.2 PSoC 5 Applikationsmodel . . . . .	25
6.3 Linux Platform / Devkit 8000 Applikationsmodel . . . . .	34

# 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

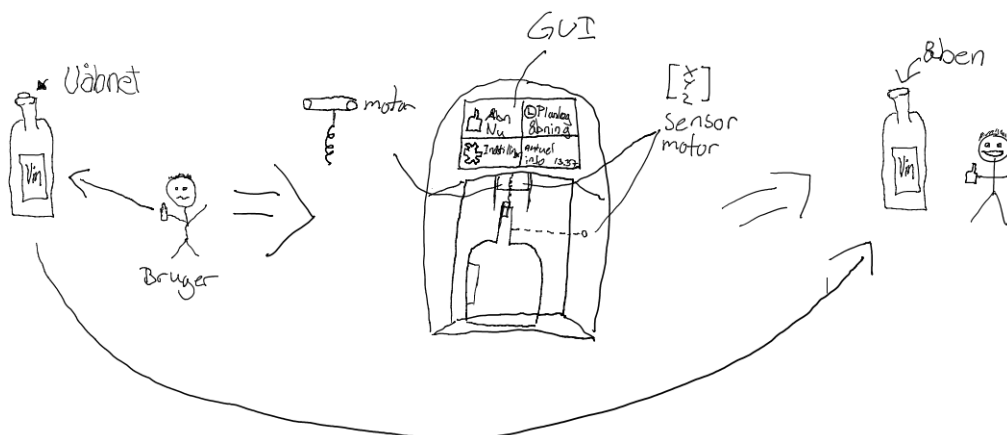
Ud fra ovenstående problemstillinger skal der fremstilles et produkt som skal automatisere vinåbning og iltning, samt hvis muligt oprette et netværk hvor vindrikkere og forhandlere kan sættes i forbindelse med hinanden. Produktet har følgende systemkrav:

Systemet:

- Skal trække korkproppen ud af en vinflaske, og ilte vinen korrekt. I denne proces indgår aktuator, sensor, PSoC, Devkit og motor.
- Skal betjenes via Devkit 8000, hvor der er installeret 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 bortskaffe vinpropper efter åbning.

- Skal kunne give status for vinåbningsprocessen.
- Skal kunne åbne vin hurtigt, når situation kræver det.
- 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.
- Kommer ikke til at kunne tilsluttes det danske el-net.

Figur 1.1: Rigt billede der beskriver wineprep



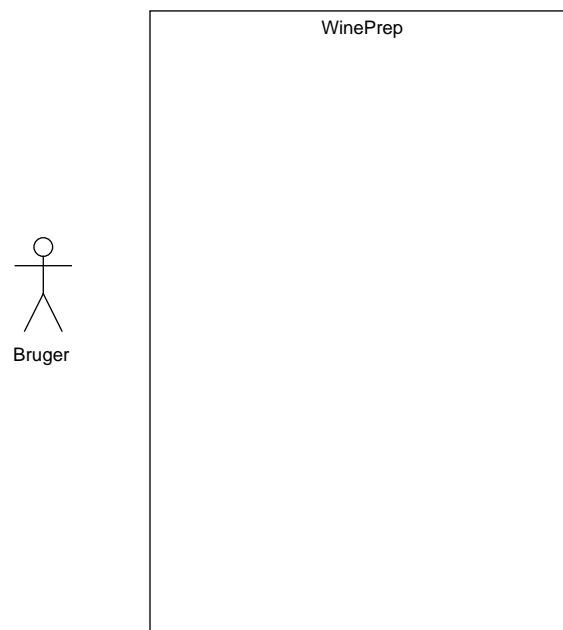
## 1.2 Risiko Analyse

Navn	Relevans	Risiko	Risk-factor
Mobil Applikation	3	10	30
GUI	10	5	50
Linux Device Driver	10	5	50
PSoC motorstyring	10	8	80
PSoC sensorstyring	9	8	72
PSoC temperatur føler	6	8	48
Winebook (social netværk)	2	10	20
Database	4	10	40
WIFI kommunikation	4	10	40
Vinhandel	1	10	10
Kamera scanning	3	10	30
Skelet/ramme	10	10	100
Låge	3	1	3
Tænd/sluk knap	10	1	10
Åbningsmekanisme	10	10	100
Mekanisk knap	10	8	80
Motor iskruning	10	10	100
Motor træk	10	7	70
Motor x-akse	8	7	56
Motor y-akse	8	7	56
Motor z-akse	10	10	100
Sensor x-akse	8	9	72
Sensor y-akse	8	9	72
Dispensering af prop	8	6	48
Reguleringstemperatur	1	1	1
Strømforsyning	10	1	10
Kamera	4	2	8

## Kapitel 2

# Kravspekifikation

### 2.1 Aktør-Kontekst



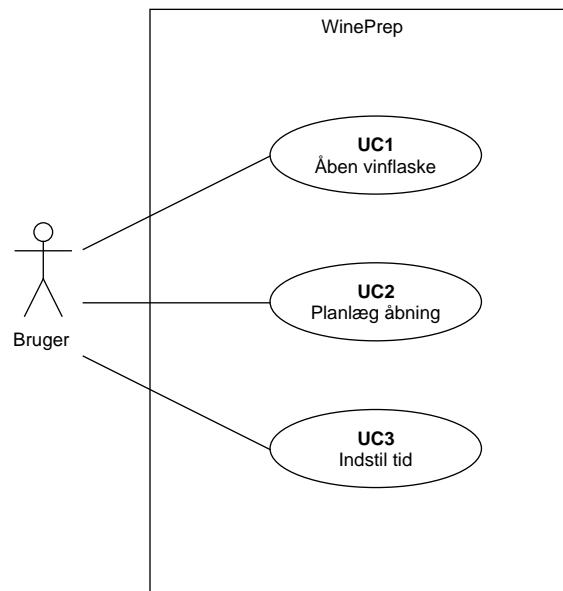
Figur 2.1: Aktør-kontekst Diagram

#### 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**

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

**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**

<b>Navn</b>	<b>UC 2: Planlæg Åbning</b>
<b>Mål</b>	Vinen er drikkeklar til et forudbestemt tidspunkt



<b>Initiering</b>	Bruger trykker <i>Planlæg åbning</i> 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 vælger tidspunkt på systemet [Ext. 1: Bruger ønsker ikke at åbne vin]</li><li>2. Bruger bekræfter valgt tidspunkt [Ext. 2: Vinen kan ikke iltes korrekt til det valgte tidspunkt]</li><li>3. System detekterer vinflaskens type og position [Ext. 3: System registrerer ugyldig type af vinflaske] [Ext. 4: System kan ikke registrere en vinflaske]</li><li>4. System venter til iltningstidspunktet [Ext. 5: Bruger annullerer planlagt åbning af vin]</li><li>5. System detekterer vinflaskens type og position [Ext. 3: System registrerer ugyldig type af vinflaske] [Ext. 4: System kan ikke registrere en vinflaske]</li><li>6. System låser vinflasken i dens position</li><li>7. System fjerner prop fra vinflasken</li><li>8. System frigiver vinflasken</li><li>9. System dispenserer prop</li><li>10. System venter til, at vinen er drikkeklar</li><li>11. System meddeler brugeren om, at vinen er drikkeklar</li></ol>

**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**

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

**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

<b>Use case under test</b>		U1: Åbn Vinflaske		
<b>Scenarie</b>		Hovedscenarie		
<b>Prækondition</b>		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

<b>Use case under test</b>		U1: Åbn Vinflaske		
<b>Scenarie</b>		Ext. 1: System registrerer ugyldig type af vinflaske		
<b>Prækondition</b>		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

<b>Use case under test</b>		U1: Åbn Vinflaske		
<b>Scenarie</b>		Ext. 2: System kan ikke registrere vinflaske		
<b>Prækondition</b>		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				
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				
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 fastlåsnings 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 fastlåsnings 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.		

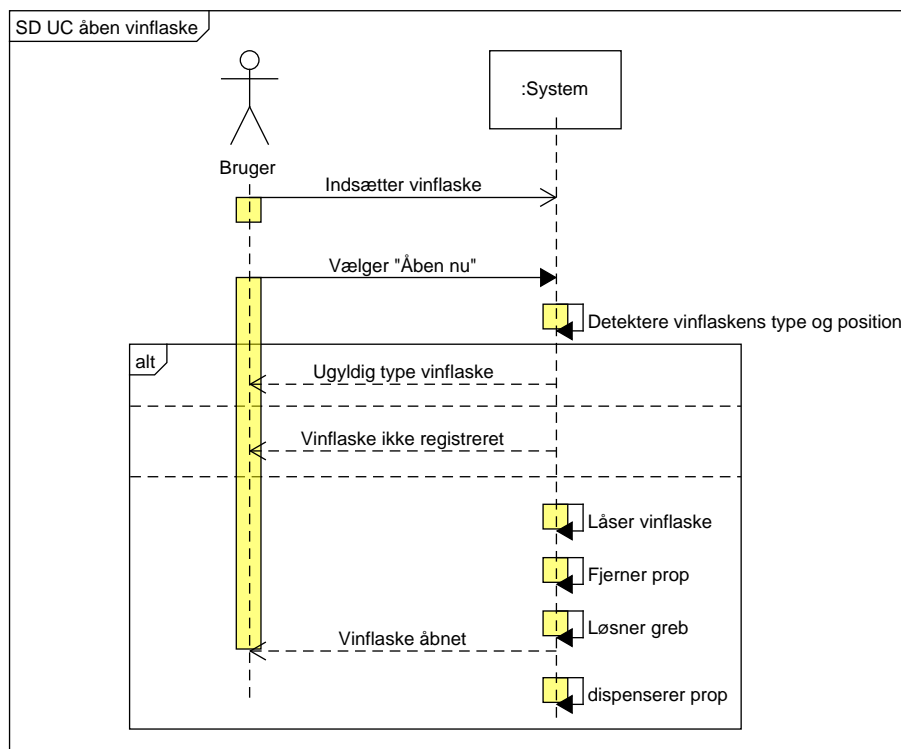
Hvad skal testes	Hvordan udføres testen	Forventet resultat	Observation/-resultat	Vurdering (OK/FAIL)
Koden til systemet skal skrives i programmerings sprogene c og c++.	Der kigges på koden til systemet.	Koden er skrevet i c/c++.		
Systemet skal betjenes via et embedded system hvorpå en Linux platform 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

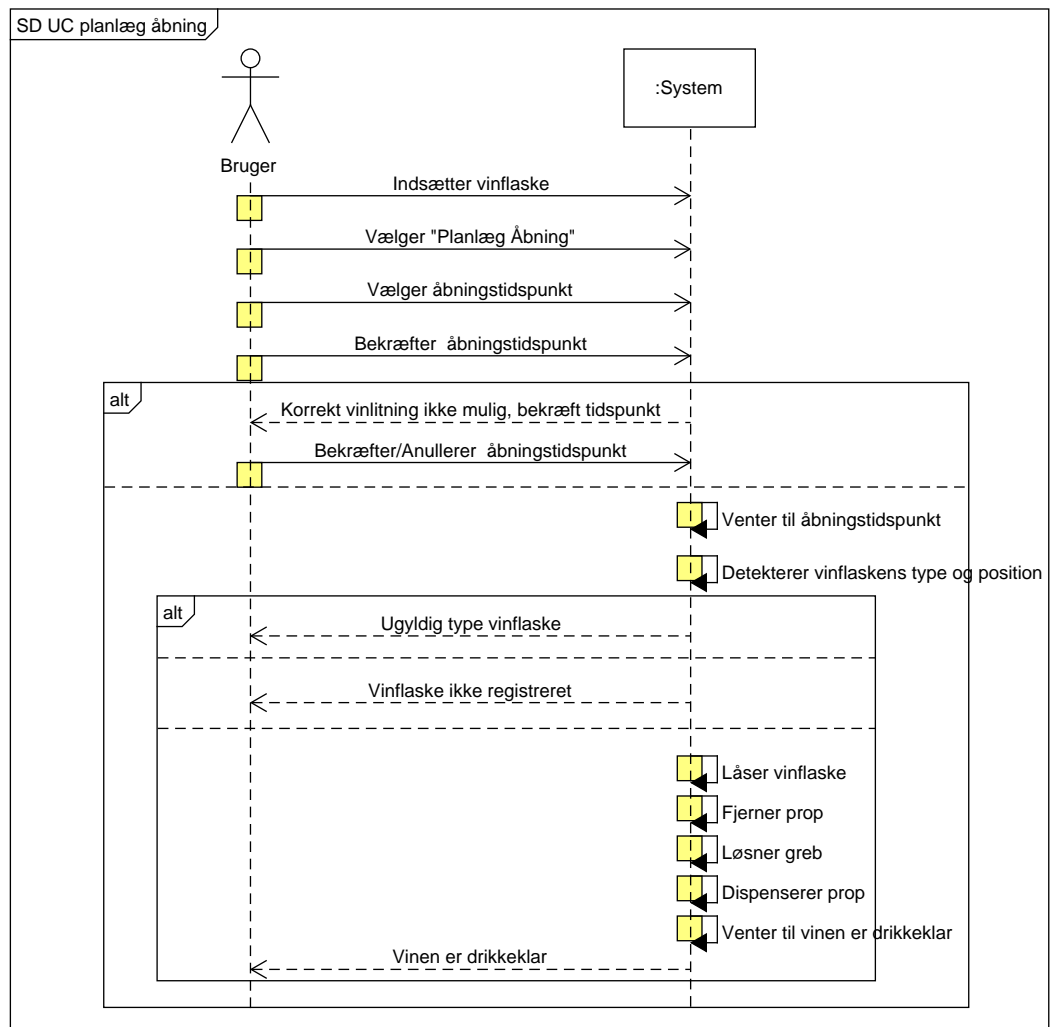
### 4.1 System Sekvens Diagrammer

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



Figur 4.1: System Sekvens diagram for UC 1

## System Sekvens Diagram for use-case 2

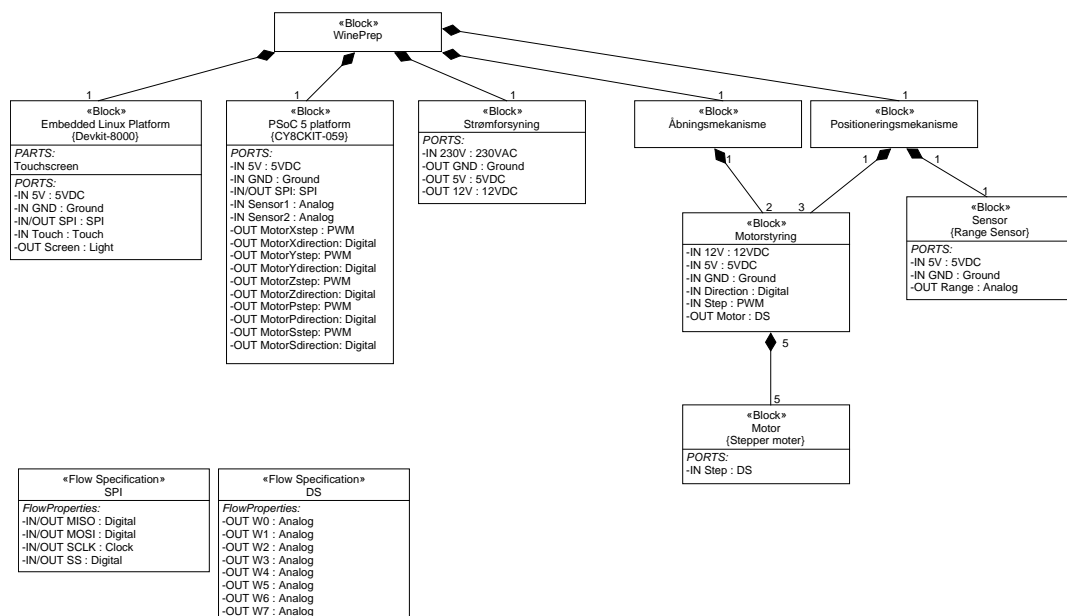


Figur 4.2: System Sekvens diagram for UC 2

## 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 21 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.

**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.

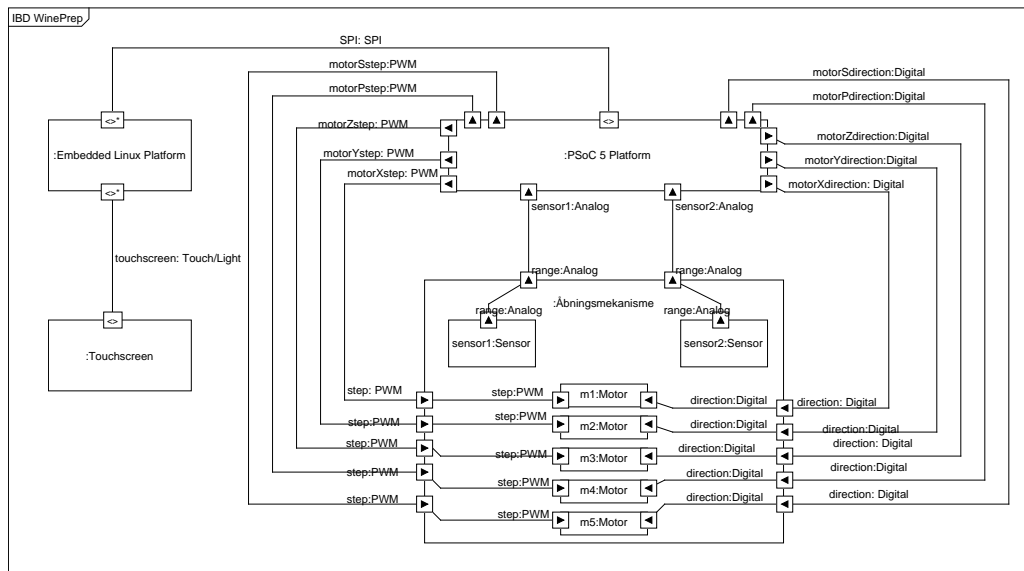
### **Ting der først bliver fast besluttet på senere iterationer/sprints**

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
<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 duty-cycle.
<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 25. 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

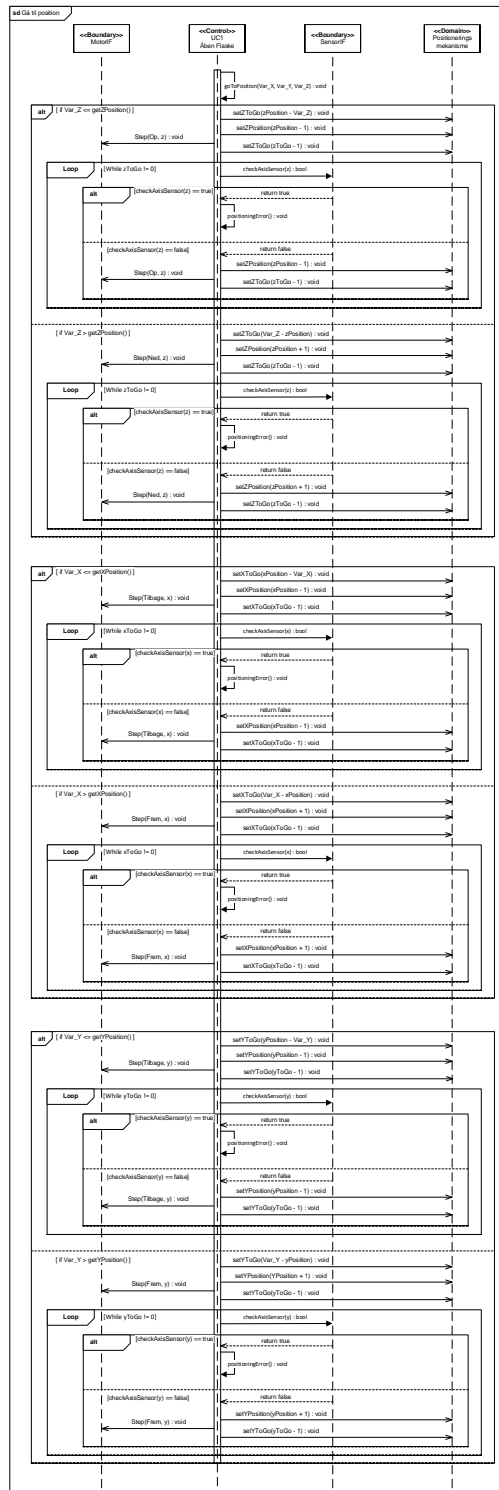
### 6.2 PSoC 5 Applikationsmodel

Alle diagrammer i dette kapitel er vedlagt som bilag i zip filen i mappen AP-PSoC 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.

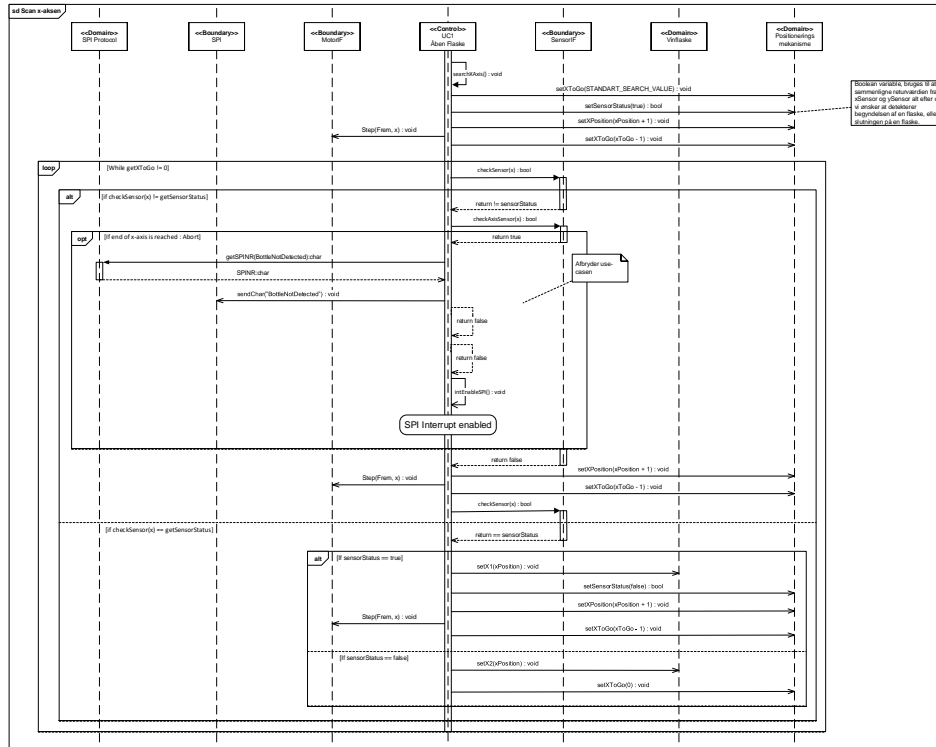
#### Sekvensdiagrammer der anvendes i Applikationsmodel for use-case 1 og 2

Da sekvensdiagrammerne for use-case 1 og 2 er meget store er det opdelt i mindre bider for at gøre de overordnede diagrammer mere overskuelige, diagrammerne i dette afsnit vil blive refereret til i de sekvensdiagrammer der anvendes til at beskrive use-case 1 og 2 i de følgende afsnit.

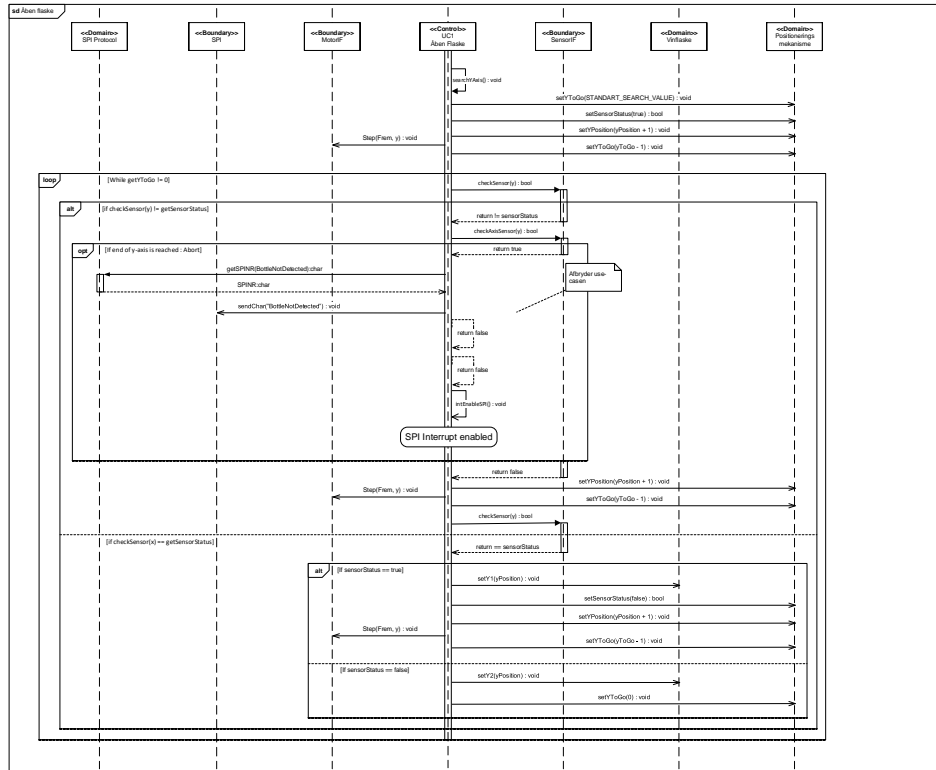
Figur 6.1: Sekvensdiagram for funktionen gå til position



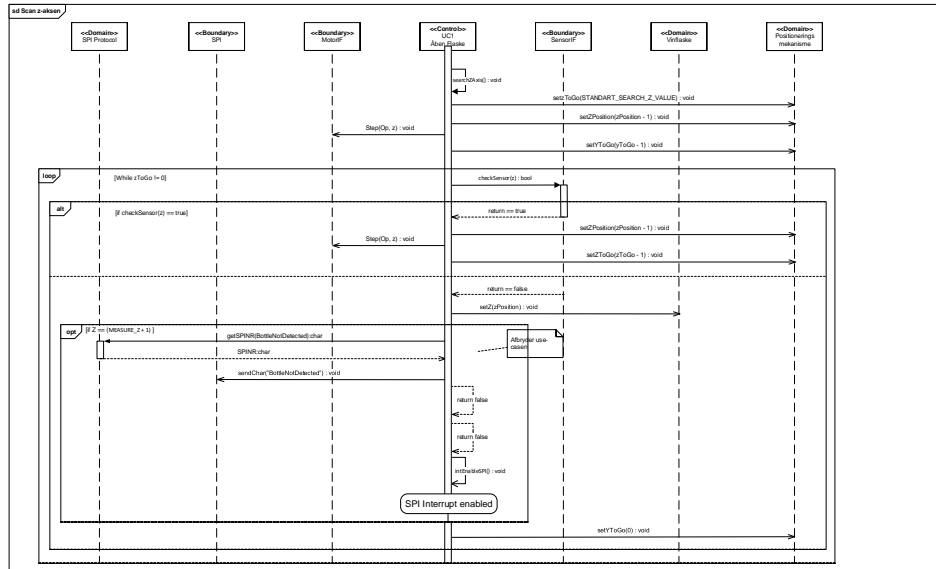
Figur 6.2: Sekvensdiagram for funktionen Scan x aksen



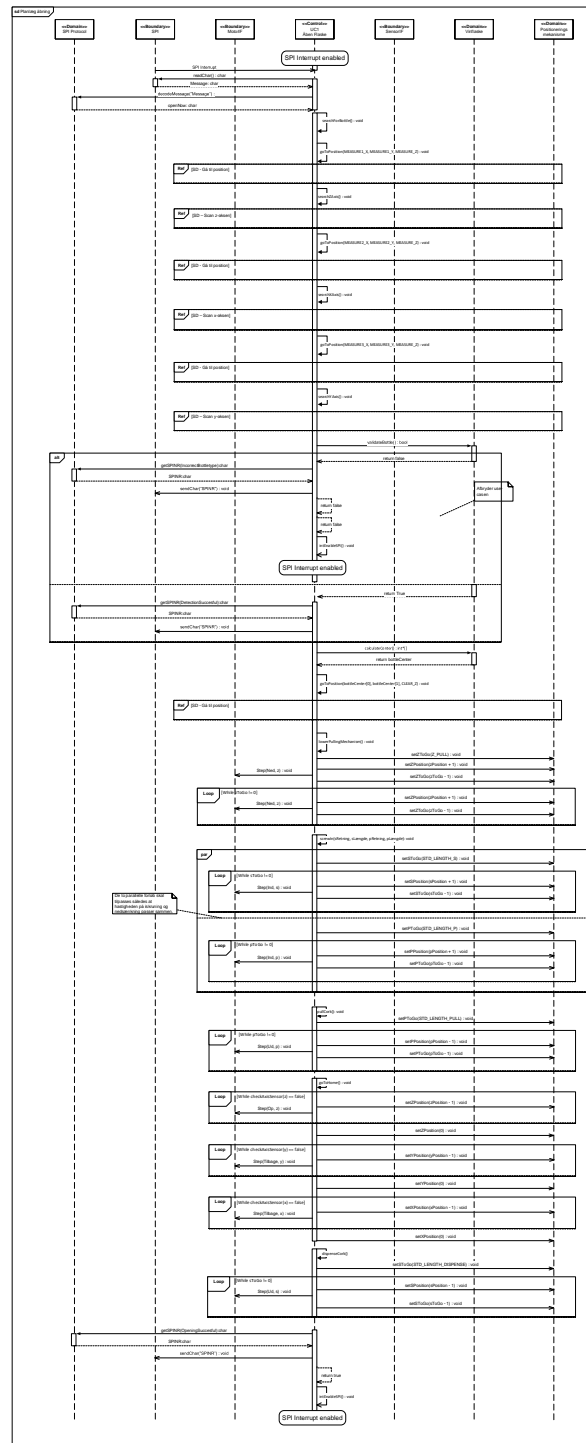
Figur 6.3: Sekvensdiagram for funktionen Scan y aksen



Figur 6.4: Sekvensdiagram for funktionen Scan z aksen



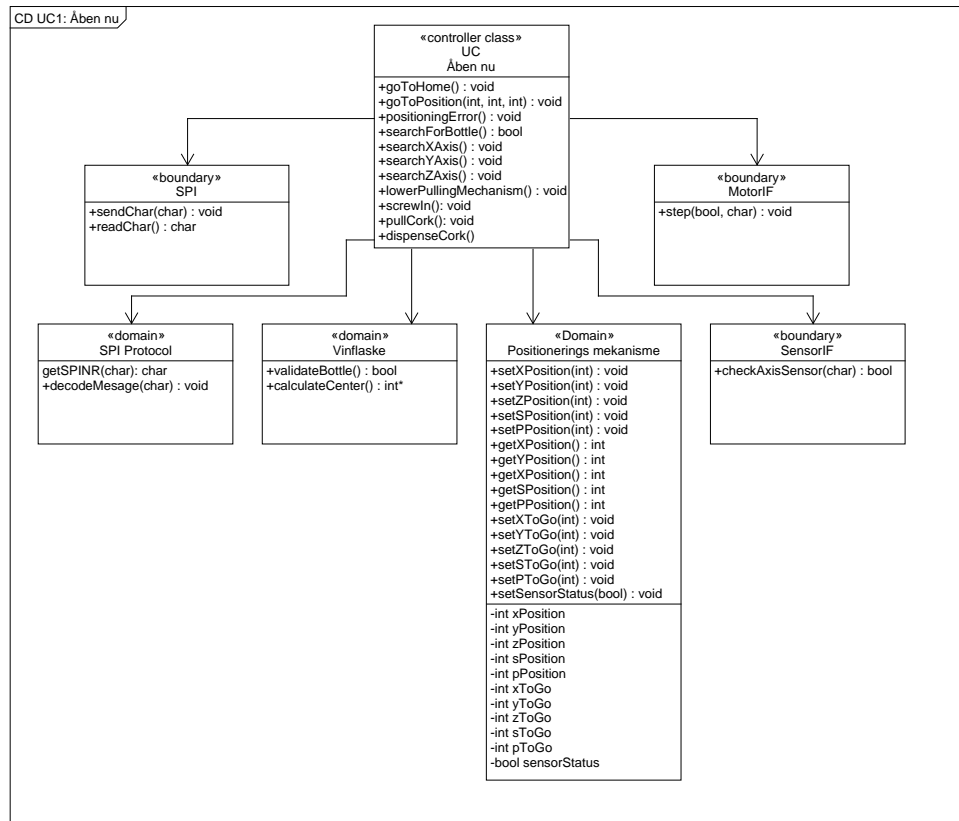
Figur 6.5: Sekvensdiagram for use-case 1: åbn nu



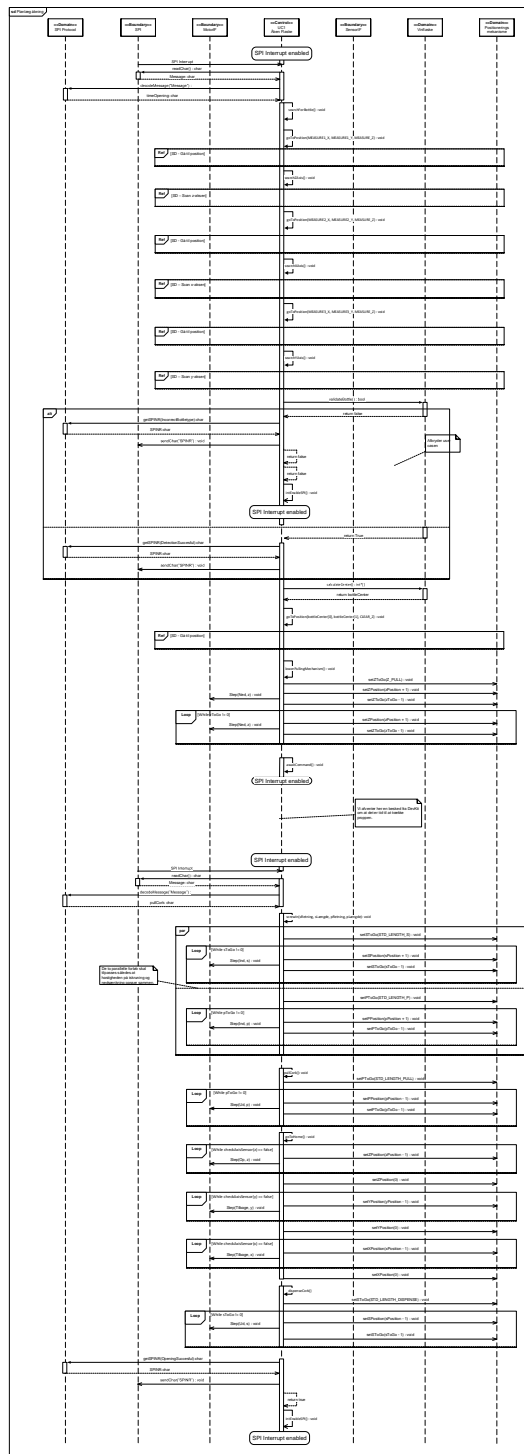


Ud fra sekvensdiagrammet på figur 6.5 side 30 samt sekvensdiagrammerne sektionen "Sekvensdiagrammer der anvendes i Applikationsmodel for use-case 1 og 2" er der udført metodeidentifikation hvilket har resulteret i klassediagrammet på figur 6.6 side 31.

Figur 6.6: Klassediagram for use-case 1: åbn nu

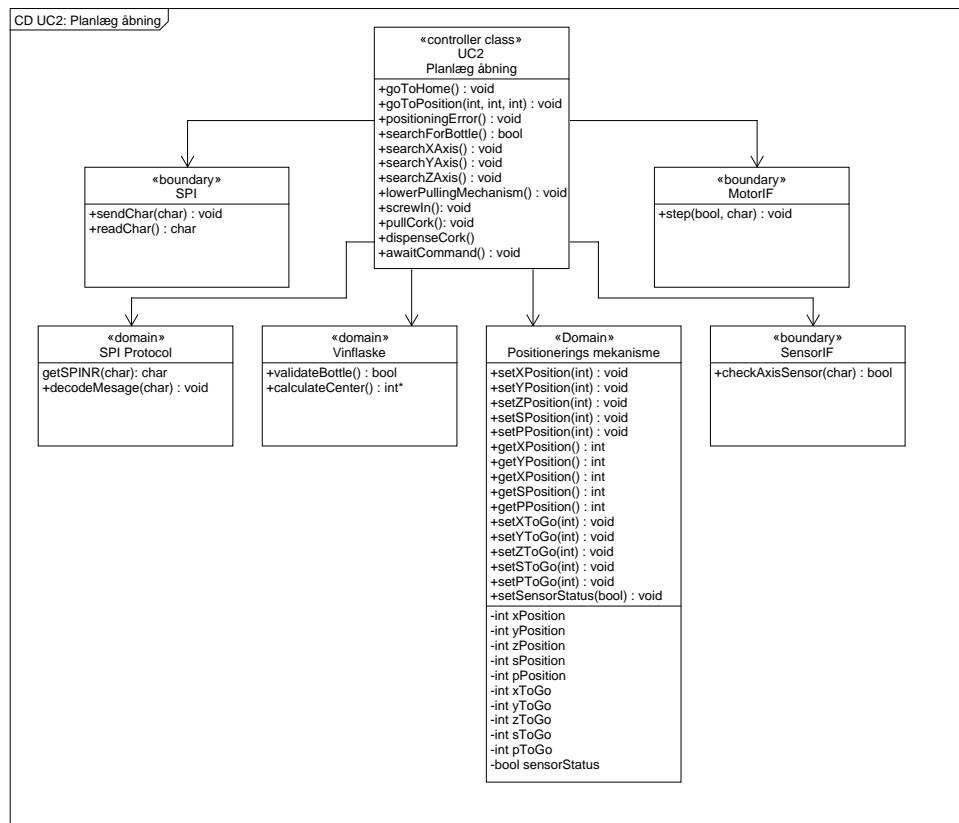


Figur 6.7: Sekvensdiagram for use-case 2: Planlæg åbning



Ud fra sekvensdiagrammet på figur 6.7 side 32 samt sekvensdiagrammerne sektionen "Sekvensdiagrammer der anvendes i Applikationsmodel for use-case 1 og 2" er der udført metodeidentifikation hvilket har resulteret i klassediagrammet på figur 6.8 side 33.

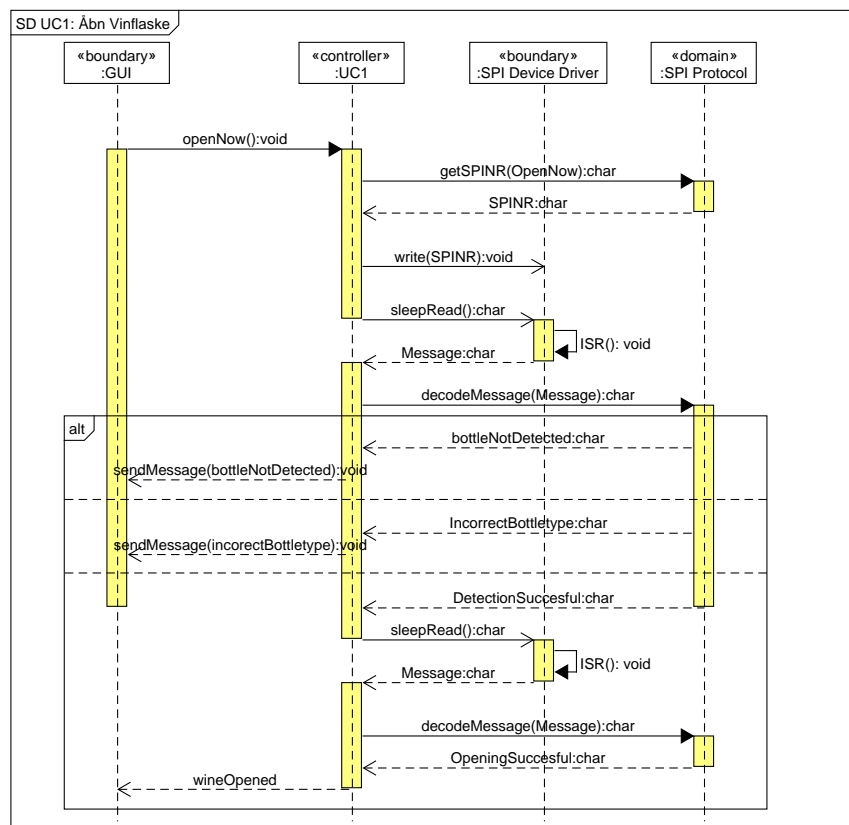
Figur 6.8: Klassediagram for use-case 2: Planlæg åbning



## 6.3 Linux Platform / Devkit 8000 Applikationsmodel

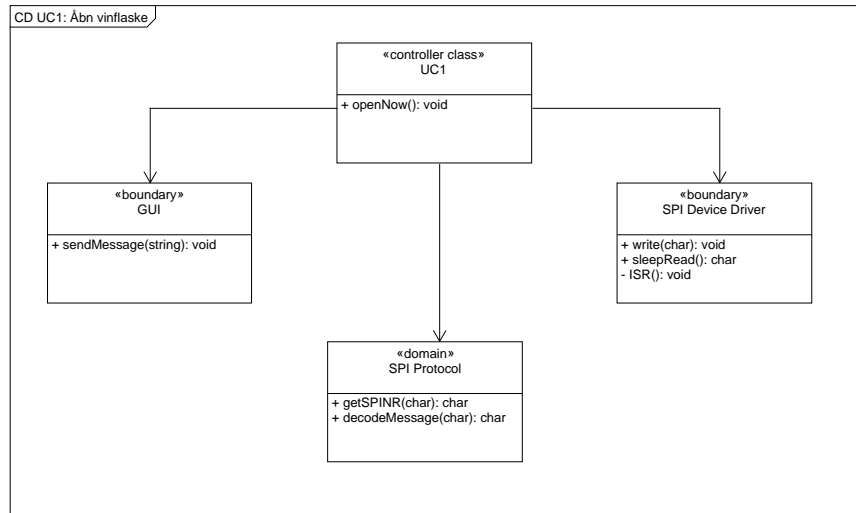
### Applikationsmodel usecase 1

Figur 6.9: Sekvensdiagram usecase 1: åbn vinflaske på devkit 8000



På figur 6.10 side 35 ses det resulterende klassediagram efter metodeidentifikation udført på basis af sekvensdiagrammet på figur 6.9 side 34.

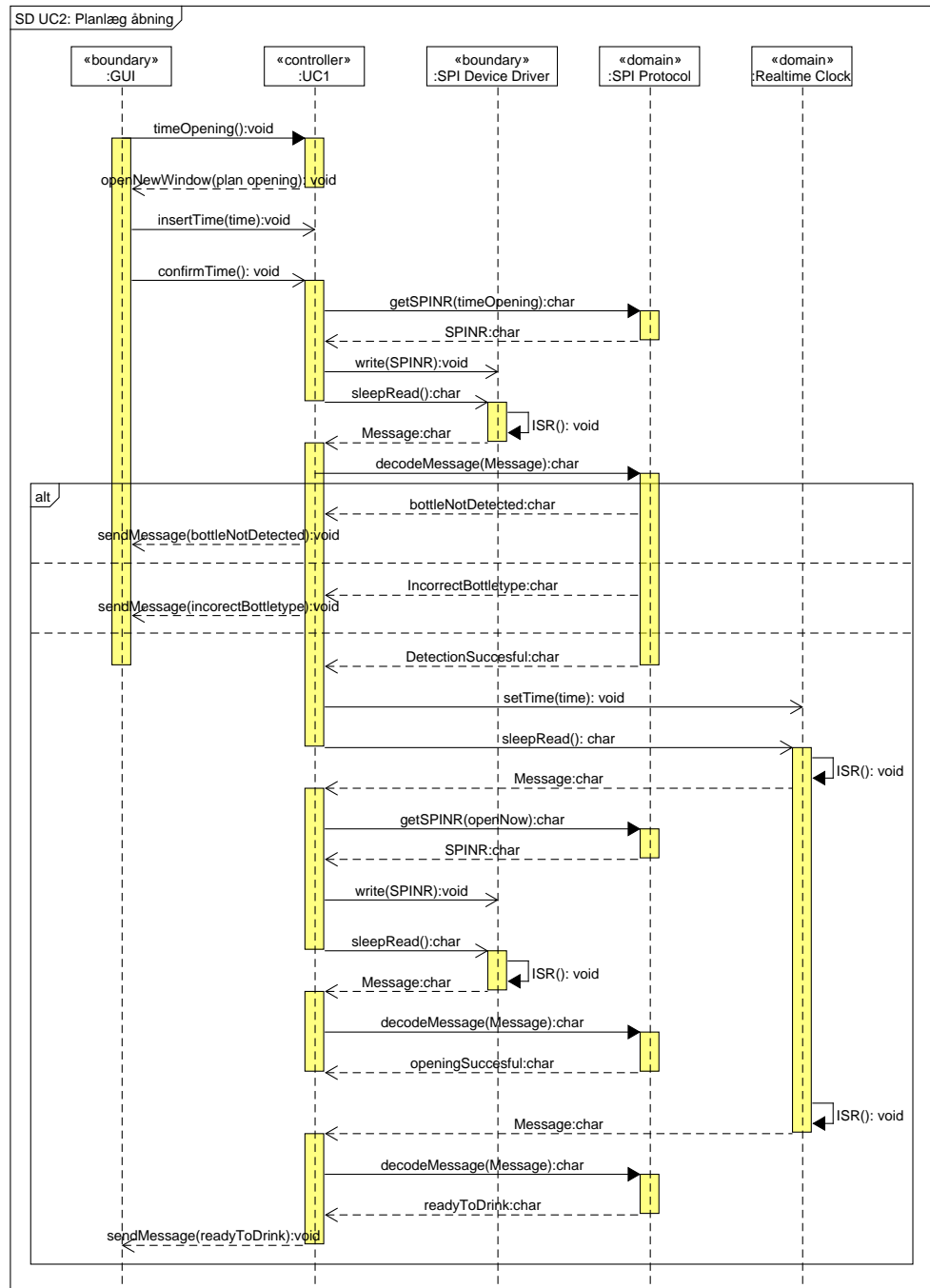
Figur 6.10: Klassediagram usecase 1: åbn vinflaske på devkit 8000



## Applikationsmodel usecase 2

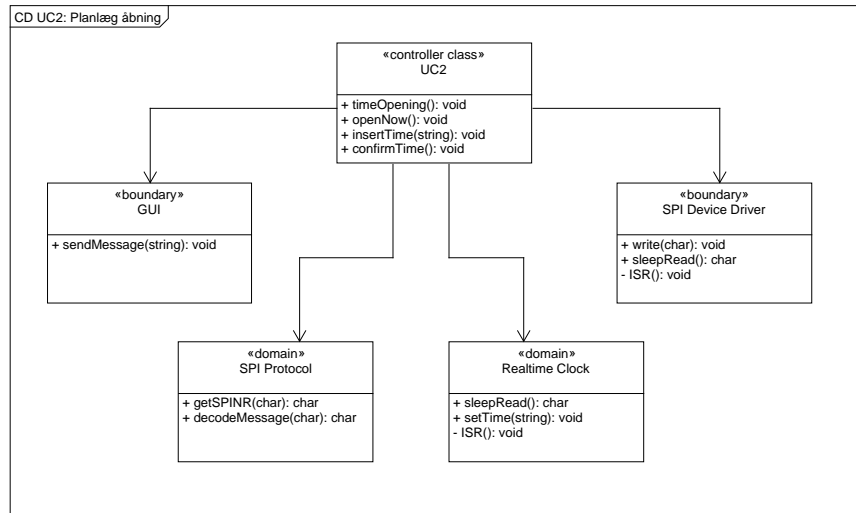
Enkelte ekstensions er udeladt på sekvensdiagrammet da de blot resultere i en terminering af usecase sekvensen.

Figur 6.11: Sekvensdiagram usecase 2: Planlæg åbning på devkit 8000



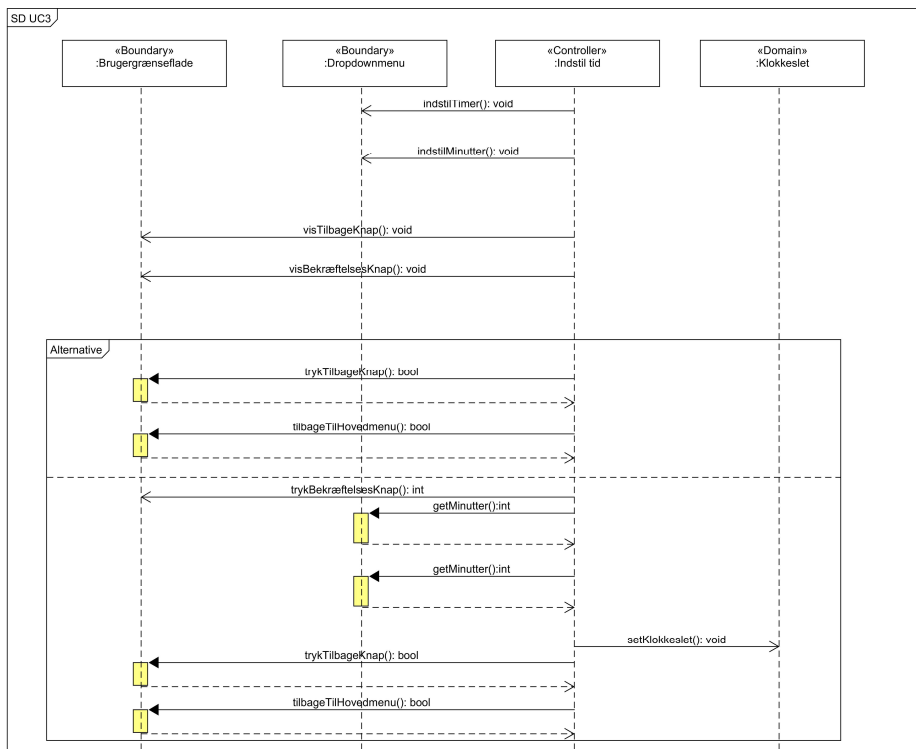
På figur 6.12 side 37 ses det resulterende klassediagram efter metodeidentifikation udført på basis af sekvensdiagrammet på figur 6.11 side 36.

Figur 6.12: Klassesdiagram usecase 2: Planlæg åbning på devkit 8000

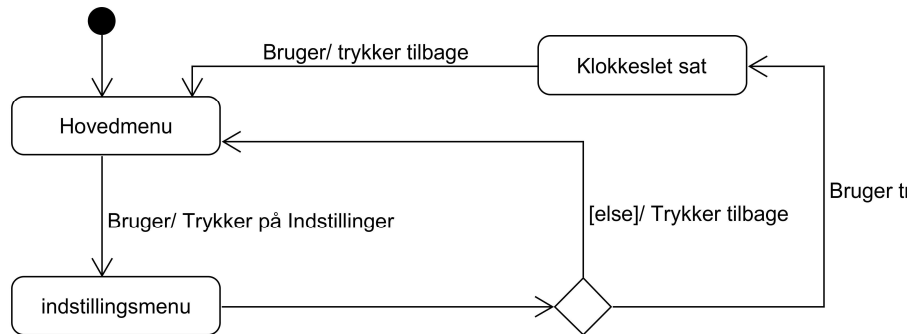


### Applikationsmodel usecase 3

Figur 6.13: Sekvensdiagram usecase 3: Indstil tid på devkit 8000



Figur 6.14: Statemachine Diagram usecase 3: Indstil tid på devkit 8000



På figur 6.15 side 38 ses det resulterende klassediagram efter metodeidentifikation udført på basis af sekvensdiagrammet på figur 6.13 side 37.

Figur 6.15: Klassediagram usecase 3: Indstil tid på devkit 8000

