



**EKG**  
Monitorering

**Opsamling, behandling og præsentation af biologiske signaler**

**Projekt-gruppe 471 – Sundhedsteknologi  
Aalborg Universitet**

**Foråret 2002**

**Hovedrapport**

TITEL: EKG Monitorering  
TEMA: Opsamling, behandling og præsentation af biologiske signaler  
PROJEKTPERIODE: 4. semester, Forår 2002  
PROJEKTGRUPPE: 471

DELTAGERE:  
Mads Peter Andersen  
Claus Graff  
Thomas Bork Hardahl  
Mette Dencker Johansen  
Michael Kruuse  
Svend A. B. Poulsen  
Jens Krog Vistisen

VEJLEDER:  
Kim Dremstrup Nielsen

OPLAGSTAL: 10  
SIDEANTAL: 157 - CD-ROM vedlagt og Kildekode rapport  
AFSLUTTET: 30. maj 2002

### Synopsis

På en kardiologisk afdeling er pålidelige værktøjer til at skaffe sig indblik i patientens hjertefunktion nødvendige. Et sådant værktøj haves i et system, der optager EKG.

Gennem Struktureret Program Udvikling er et EKG-system udviklet. Input til kravspecifikation er fundet gennem interview med personale på en kardiologisk afdeling.

Systemet er inddelt i de fire delsystemer instrumentering, microcontroller, seriel kommunikation og pc. Instrumentering og microcontroller udgør en bed-side enhed, og pc udgør en central overvågningsenhed. På microcontrolleren AD konverteres de tre afledninger. Pulsen beregnes vha. QRS-detektering, og der alarmeres, hvis pulsen forlader et brugerdefineret normalområde. Puls og alarmer vises på et interaktivt display. På pc'en lagres og vises 6 afledninger. Der kan vises fortløbende EKG eller lagret EKG kan hentes og vises. Systemet er udviklet til én patient.

Kravspecifikationen er opfyldt, idet der er implementeret et EKG-system, der virker som forventet. Der er dog mindre problemer med bevægelsesartefakter samt pc'ens serielbuffer.



# Forord

Denne rapport er udarbejdet af gruppe 471 på 4. semester, uddannelsen i sundhedsteknologi, Institut for Sundhedsteknologi ved Aalborg Universitet. Temaet for 4. semester er “Opsamling, behandling og præsentation af biologiske signaler”. Rapporten henvender sig især til vejleder og censor samt medstuderende med interesse for emnet. Derfor forudsætter udbytterig læsning samme teknisk-faglige grundlag som projektgruppen.

## Rapportens opbygning

Rapporten er opdelt i 5 hoveddele: foranalyse, analyse, kravspecifikation, design/implementation samt afrunding, der indeholder konklusion og perspektivering. Til hver hoveddel findes en kort introduktion på forsiden til hoveddelen.

Rapporten indledes med en foranalyse i kap. 1, der fungerer som problemdokumentation og derudover indeholder de indledende valg i forbindelse med projektet, således at en problemformulering kan opstilles.

Derefter analyseres brugerkravene til det system, der kræves i problemformuleringen. Denne analyse i kap. 3 hører til udarbejdelsen af produktspecifikation for systemet.

Ud fra produktspecifikationen foretages systemdesign i kap. 4, hvor systemet inddelles i delsystemer. Krav til delsystemerne analyseres, og delsystemerne nedbrydes med henblik på opstilling af kravspecifikation i kap. 6. Nedbrydningsgraden for delsystemerne i kap. 4 varierer, da det er projektgruppens opfattelse, at en kravspecifikation skal beskrive afgrænsede systemdele, der let kan implementeres og stadig udgøre en funktionel enhed. Kravspecifikationen i kap. 6 stiller derfor krav til forstærker- og filterdesign i instrumenteringsdelsystemet. For mc'en opstilles krav på modulniveau og for pc-delsystemet stilles krav på procesniveau. For mc-delsystemet fastlægges grænsefladerne mellem modulerne.

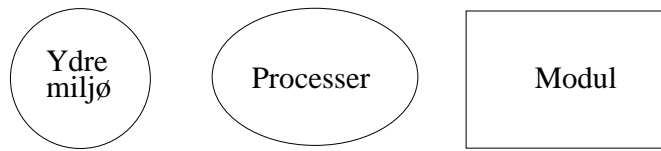
Ud fra kravspecifikationen designes, implementeres og testes i kapitlerne 8 til 14 de funktionelle enheder, der er stillet krav til i kravspecifikationen. Dog findes dette for instrumentering i appendiks D.

Det endelige system funktionalitetstestes i kap. 15.

I appendiks C findes en kort beskrivelse af den specifikke microcontroller anvendt i systemet.

## Figurer og diagrammer

Figurer er nummereret fortløbende gennem kapitlerne. På procesdiagrammer i kapitel 3 og 4 er ydre miljø angivet med cirkler, processer med ovaler og moduler med firkanter. På flowdiagrammer er rutinerne angivet med tal i en cirkel; i teksten henvises til specifikke steder på flowdiagrammer via dette tal, der angives i (parantes). Se modeller i figur 1.



*Figur 1: På figuren ses modeller for opstilling af flowdiagrammer.*

## Litteraturhenvisninger

Litteraturhenvisninger angives med fortløbende numre i firkantede paranteser[]. En samlet litteraturliste findes bagerst i rapporten.

På rapportens bagside findes en CD-ROM, der indeholder supplerende oplysninger om projektet, kildekode til mc-program og LabVIEW-program.

## Vi takker...

Projektgruppen ønsker at takke Kim D. Nielsen for gode råd, ideer og kyndig vejledning. Desuden er projektgruppen personalet, samt overlægerne Lars Hvilsted Rasmussen og Bodil Steen Rasmussen, på Kardiologisk afdeling samt Thorax-internsiv afdeling (TIA) på Aalborg Sygehus Syd taknemmelig for deres bidrag til at skabe et virkelighedsnært system. I forbindelse med det praktiske arbejde med microcontrolleren har Morten Fjordback og John Hansen været en stor hjælp. Tak!

---

Mads Peter Andersen

---

Claus Graff

---

Thomas Bork Hardahl

---

Mette Dencker Johansen

---

Michael Kruuse

---

Svend A. B. Poulsen

---

Jens Krog Vistisen

**Aalborg Universitet, 30. maj 2002**

# Indhold

<b>I FORANALYSE</b>	<b>1</b>
<b>1 Foranalyse</b>	<b>3</b>
1.1 Valg af EKG som projekttema . . . . .	3
1.2 Relevans af EKG som værktøj for sundhedspersonale . . . . .	3
1.2.1 Anvendelse af EKG som metode . . . . .	3
1.2.2 Anvendelsesområder . . . . .	3
1.2.3 Konkret anvendelse af EKG på en kardiologisk afdeling . . . . .	4
1.3 Problemformulering . . . . .	5
<b>II ANALYSE</b>	<b>7</b>
<b>2 EKG - et biologisk signal</b>	<b>9</b>
2.1 EKG-signalet . . . . .	9
2.1.1 Sammenhæng mellem EKG og hjertecyklus. . . . .	9
2.2 Størrelser af EKG-signalet . . . . .	11
<b>3 Opstilling af produktspecifikation</b>	<b>13</b>
3.1 Funktionaliteter . . . . .	13
3.1.1 Akutte alarmer ved pumpesvigt . . . . .	13
3.1.2 Parametre fra EKG samt kronologiske fordelinger af events . . . . .	14
3.1.3 Visning af EKG for alle patienter . . . . .	15
3.2 Adgang til funktionaliteter hos patient og centralt . . . . .	15
3.2.1 Bed-side . . . . .	16
3.2.2 Centralskærm . . . . .	17
3.3 Produktspecifikation . . . . .	19
<b>4 Systemdesign</b>	<b>21</b>
4.1 Inddeling i delsystemer . . . . .	21
4.2 Valg af opsamlingsmetode . . . . .	22
4.3 Instrumentering . . . . .	22
4.3.1 Forstærkning . . . . .	22
4.3.2 Filtrering . . . . .	24
4.4 Kommunikation mellem mc og pc . . . . .	27
4.5 Mc . . . . .	27
4.5.1 Struktur af eksekverbart program - main . . . . .	28
4.5.2 Proces: Konverter og hent fra ADC . . . . .	29
4.5.3 Proces: Pulsberegning . . . . .	29
4.5.4 Proces: Alarmering . . . . .	29
4.5.5 Proces: Brugergrenseflade . . . . .	30
4.5.6 Kommuniker med pc . . . . .	32
4.6 Pc . . . . .	32

<b>5</b>	<b>Afgrænsning af system</b>	<b>35</b>
<b>III</b>	<b>KRAVSPECIFIKATION</b>	<b>37</b>
<b>6</b>	<b>Kravspecifikation</b>	<b>39</b>
6.1	Indledning . . . . .	39
6.2	De specifikke krav . . . . .	39
6.2.1	Definitioner . . . . .	39
6.2.2	Krav til delsystemet <i>instrumentering</i> . . . . .	39
6.2.3	Krav til delsystemet <i>mc (5)/(6)</i> . . . . .	40
6.2.4	Serial kommunikation (7) . . . . .	43
6.2.5	Krav til pc-delsystemet (8) . . . . .	43
<b>IV</b>	<b>DESIGN, IMPLEMENTATION OG TEST</b>	<b>45</b>
<b>7</b>	<b>Introduktion til design og implementation</b>	<b>47</b>
7.1	Designmetoder . . . . .	47
7.1.1	Struktur i dokumentationen . . . . .	47
<b>8</b>	<b>Konverter og hent fra ADC</b>	<b>49</b>
8.1	Modul: Opsætning af ADC . . . . .	49
8.1.1	Design af Opsætning af ADC modul . . . . .	50
8.1.2	Implementation af Opsætning af ADC modul . . . . .	50
8.2	Modulet Lagring af konverterede signaler . . . . .	51
8.2.1	Design af Lagring af konverterede signaler . . . . .	51
8.2.2	Implementation af Lagring af konverterede signaler . . . . .	51
8.2.3	Test af Konverter og hent fra ADC . . . . .	53
<b>9</b>	<b>Kommunikér med PC</b>	<b>55</b>
9.1	Modul: Serial kommunikation og Opsætning af UART . . . . .	55
9.2	Design af Serial kommunikation modul . . . . .	55
9.3	Design af Opsætning af UART modul . . . . .	57
9.4	Implementation af Opsætning af UART modul . . . . .	57
9.5	Implementation af Serial kommunikation modul . . . . .	57
9.6	Test af Kommunikér med pc . . . . .	59
<b>10</b>	<b>Pulsdetektering</b>	<b>61</b>
10.1	Valg af detekteringsmetode . . . . .	61
10.1.1	QRS-detektering . . . . .	61
10.2	Afledet Beregning . . . . .	63
10.2.1	Implementation af Afledet beregnings modul . . . . .	65
10.2.2	Test af differentiationsalgoritme . . . . .	66
10.3	Kalibrerings modul . . . . .	66
10.3.1	Design af kalibreringsmodul . . . . .	66
10.3.2	Implementation af kalibreringsmodul . . . . .	68
10.3.3	Test af kalibreringsmodul . . . . .	70
10.4	QRS-detekteringsmodul . . . . .	71
10.4.1	Design af QRS-detekteringsmodul . . . . .	71
10.4.2	Implementation af QRS-detekteringsmodul . . . . .	71

10.4.3	Test af QRSdetekteringsmodul . . . . .	73
10.5	Design af Beregn puls modul . . . . .	74
10.5.1	Implementation af tidstæller . . . . .	75
10.5.2	Implementation af Beregn puls modul . . . . .	75
10.5.3	Test af pulsberegningsmodul . . . . .	75
<b>11</b>	<b>Alarmering</b>	<b>77</b>
11.1	Alarmdetekteringsmodul . . . . .	77
11.1.1	Design af Alarmdetekteringsmodul . . . . .	77
11.1.2	Implementation af Alarmdetekteringsmodul . . . . .	77
11.1.3	Test af Alarmdetekteringsmodul . . . . .	78
11.2	Alarmlyd- og mutemodul . . . . .	79
11.2.1	Design af Alarmlyd- og mutemodul . . . . .	79
11.2.2	Implementation af alarmlyd- og mutemodul . . . . .	80
11.2.3	Test af alarmlyd- og mutemodul . . . . .	83
<b>12</b>	<b>Brugergrænseflade</b>	<b>85</b>
12.1	Optimering af udskrivning på displayet . . . . .	85
12.2	Modul: opsætning af tekstdisplay . . . . .	86
12.2.1	Design af opsætnings-modul . . . . .	86
12.2.2	Implementation af opsætnings-modul . . . . .	86
12.2.3	Test af opsætnings-modul . . . . .	87
12.3	Modul: Opdatering af tekstdisplay . . . . .	88
12.3.1	Design af opdaterings modul . . . . .	88
12.3.2	Implementation af visning af hovedmenu . . . . .	90
12.3.3	Test af visning af hovedmenuen . . . . .	91
12.3.4	Implementation af visning af grænsemenuen . . . . .	92
12.3.5	Test af visning af grænsemenuen . . . . .	92
12.3.6	Implementation af Muting . . . . .	94
12.3.7	Test af muting . . . . .	94
12.3.8	Implementation af grænseindstilling . . . . .	94
12.3.9	Test af grænseindstilling . . . . .	95
12.3.10	Implementation af aktivering af alarmskærm . . . . .	95
12.3.11	Test af aktivering af alarmskærm . . . . .	95
<b>13</b>	<b>Samling af moduler</b>	<b>97</b>
13.1	Main . . . . .	97
13.1.1	Struktur af headerfiler . . . . .	97
13.1.2	Variable . . . . .	98
13.1.3	Struktur i main . . . . .	98
13.1.4	Test af main . . . . .	98
<b>14</b>	<b>Design, implementation og test af brugerflade på PC'en</b>	<b>101</b>
14.1	Design . . . . .	101
14.1.1	Hvad er LabVIEW? . . . . .	101
14.1.2	Brugerflade og programdesign . . . . .	101
14.1.3	Programstruktur . . . . .	103
14.2	Implementation . . . . .	103
14.2.1	Tråd til opsamling . . . . .	103
14.2.2	Tråd til filhåndtering . . . . .	106
14.3	Test . . . . .	106
14.3.1	Test af subVI's . . . . .	106
14.3.2	Test af processer . . . . .	106
14.4	Problemer med opsamling af serielt signal . . . . .	108



<b>15</b>	<b>Funktionalitetstest</b>	<b>109</b>
15.1	Afgrænsning . . . . .	109
15.1.1	Pulsberegning . . . . .	109
15.1.2	Visualisering af afledninger . . . . .	109
15.2	Screenshot fra oscilloskop og Labview . . . . .	112
<b>V</b>	<b>KONKLUSION OG PERSPEKTIVERING</b>	<b>115</b>
<b>16</b>	<b>Konklusion</b>	<b>117</b>
<b>17</b>	<b>Perspektivering</b>	<b>119</b>
	Litteratur	121
<b>VI</b>	<b>APPENDIKS</b>	<b>123</b>
<b>A</b>	<b>Referat af besøg på Aalborg sygehus Syd</b>	<b>125</b>
A.1	Anæstesi- Intensiv- og operationsafdeling . . . . .	125
A.1.1	Funktionaliteter ved eksisterende system . . . . .	126
A.1.2	Vurdering af systemet . . . . .	126
A.2	Kardiologisk afdeling . . . . .	127
<b>B</b>	<b>Principper for EKG opsamling</b>	<b>129</b>
<b>C</b>	<b>Hardware specifikation - mc'en (MSP430F149)</b>	<b>133</b>
C.1	Anvendte mc features . . . . .	134
C.1.1	Hukommelsen . . . . .	134
C.1.2	Basic Clock module . . . . .	134
C.1.3	ADC . . . . .	134
C.1.4	Timer A . . . . .	134
C.1.5	Timer B . . . . .	134
C.1.6	Watchdog Timer . . . . .	135
C.1.7	USART0 . . . . .	135
C.1.8	I/O Port 1,2 . . . . .	135
C.1.9	I/O Porte 3,4,5 . . . . .	135
C.2	Interrupt stuktur for MC . . . . .	135
C.3	Anvendelsen af eksterne enheder . . . . .	135
<b>D</b>	<b>Design, implementation og test af instrumentering</b>	<b>137</b>
D.1	Design og implementation for forforstærker . . . . .	137
D.1.1	Krav til forforstærker . . . . .	137
D.1.2	Valg af forstærkertype . . . . .	137
D.1.3	Beregning af gain-modstand . . . . .	138
D.1.4	Driven-right-leg kredsløb . . . . .	138
D.1.5	Implementation . . . . .	141
D.1.6	Test af forforstærker . . . . .	141
D.1.7	Resultat af test . . . . .	141
D.2	Design og implementation af filter . . . . .	143
D.2.1	Krav til filter . . . . .	143
D.2.2	Højpasfilter . . . . .	143
D.2.3	Summer . . . . .	144
D.2.4	Lavpasfilter . . . . .	144
D.3	Test af filter . . . . .	145

<b>E</b>	<b>Hardware specifikation - Seriel kommunikation</b>	<b>147</b>
E.1	Design og implementation for seriel kommunikation . . . . .	147
E.1.1	Krav til seriel kommunikation . . . . .	147
E.1.2	Design . . . . .	147
E.1.3	Forsyning . . . . .	148
E.2	Testspecifikationer for seriel kommunikation . . . . .	148
E.2.1	Individuelle tests . . . . .	149
E.2.2	Test af samlet hardware til seriel kommunikation . . . . .	149
E.3	Test . . . . .	149
E.3.1	Spændingsforsynings-test . . . . .	149
E.3.2	Individuel test: Tx/Rx-optokobler . . . . .	150
E.3.3	Individuelle tests: MAXIM3223 . . . . .	150
E.3.4	Individuelle tests: Kabeltest . . . . .	150
E.3.5	Samlet test . . . . .	150
<b>F</b>	<b>Interfacing af LCD-display</b>	<b>153</b>
F.1	Displayets ben og funktioner . . . . .	153
<b>G</b>	<b>Konfiguration - Eksterne kredsløb</b>	<b>155</b>
<b>VII</b>	<b>BILAG A</b>	<b>157</b>



# DEL I FORANALYSE

*I foranalysen foretages de indledende valg i projektet, således at en problemformulering kan opstilles. Valgene funderes både på teori, logik og samtale med klinikere. Visse valg træffes dog alene ud fra projektgruppens interesser.  
Der vil i foranalysen blive refereret til appendiks A, hvor interview med klinikere er gengivet som referat.*



# Kapitel 1

## Foranalyse

### 1.1 Valg af EKG som projekttema

Projektgruppen har valgt at arbejde med hjertets funktion, som den beskrives vha. elektrokardiografi (herefter: EKG), da valget af EKG giver mulighed for at opsamle, behandle og præsentere et biologisk signal, hvilket er semestrets hovedtema.

### 1.2 Relevans af EKG som værktøj for sundhedspersonale

Den menneskelige organisme er en kompleks samling af systemer, der alle skal fungere, hvis livet skal opretholdes. Et af disse systemer er blodcirkulationssystemet, der transporterer livsvigtige stoffer til og fra vævet, og på den måde vedligeholder homøostase.

Hjertets funktion er en forudsætning for liv, da hjertet er den pumpe, der aktivt sørger for transporten af blod ud i organismen. Hvis hjertet ikke pumper en tilstrækkelig mængde blod rundt i kroppen, skal der derfor gribes ind, for at undgå at patienten mister livet.

For at der kan gribes ind, må det vides, hvad årsagen til hjertets fejlfunktion er; der må stilles en diagnose. For at kunne stille en diagnose må lægen først vide, hvordan patientens hjerte fungerer, dvs. præcist hvad fejlfunktionen skyldes. Men da han ikke altid kan lytte og føle sig til, hvordan en patients hjerte fungerer, må han have et værktøj til at se ind i hjertet. EKG er i den forbindelse et glimrende og meget udbredt værktøj til at se, hvordan hjertet stimulerer sig selv til at pumpe, og dermed hvor godt og hvordan hjertet pumper.

#### 1.2.1 Anvendelse af EKG som metode

EKG er i hovedtræk et fysiologisk redskab til at vise den fysiske udbredelse af de signaler, der skal få hjertet til at trække sig sammen. Afvigelser fra det normale EKG kan fortælle, hvor i hjertet der ikke er normal ledningsevne, og dermed hvad årsagen til, og lokaliteten af, dysfunktionen er. En læge opnår kendskab til patientens EKG ved at placere elektroder på patienten og måle med et EKG-apparat. Som formål med dette projekt vælges derfor at udvikle et EKG-apparat, der kan opsamle EKG og give lægen adgang til det optagne EKG. EKG som metode beskrives i kapitel 2.

#### 1.2.2 Anvendelsesområder

EKG er et vigtigt værktøj i behandlingen af mange forskellige typer patienter og bruges derfor bredt inden for de medicinske specialer, dvs. både af praktiserende læger og på de fleste sygehusafdelinger. Således anvendes EKG eksempelvis til at foretage den indledende diagnose, når en

patient med et hjerteanfald er bragt fra ambulancen og ind på skadestuen, og under en operation for at overvåge patientens tilstand. En anden og meget direkte anvendelse er i diagnosticering og behandling af ikke-akutte hjertepatienter på de kardiologiske afdelinger, og projektgruppen vælger at afgrænse anvendelsen af EKG i dette projekt til netop den kardiologiske afdeling. Denne afgrænsning vælges, da opsamling, behandling og præsentation af EKG alene, og ikke også andre biologiske signaler, er relevant på en kardiologisk afdeling. Dette er ikke tilfældet på fx en intensiv afdeling, hvor der monitoreres flere vitalparametre som fx iltmætning og respirationsparametre.

Med denne afgrænsning kan definitionen af projektets formål uddybes til, at det udformede EKG-apparat skal målrettes anvendelse ved diagnosticering og behandling på en kardiologisk afdeling.

### 1.2.3 Konkret anvendelse af EKG på en kardiologisk afdeling

Projektgruppen har i forbindelse med projektet besøgt kardiologisk afdeling på Aalborg Sygehus Syd for at få et indblik i, hvordan EKG anvendes, og hvordan de anvendte EKG-apparater fungerer. Et referat af besøget findes i appendiks A, men af hensyn til læsbarheden af dette afsnit resumeres her de vigtigste oplysninger.

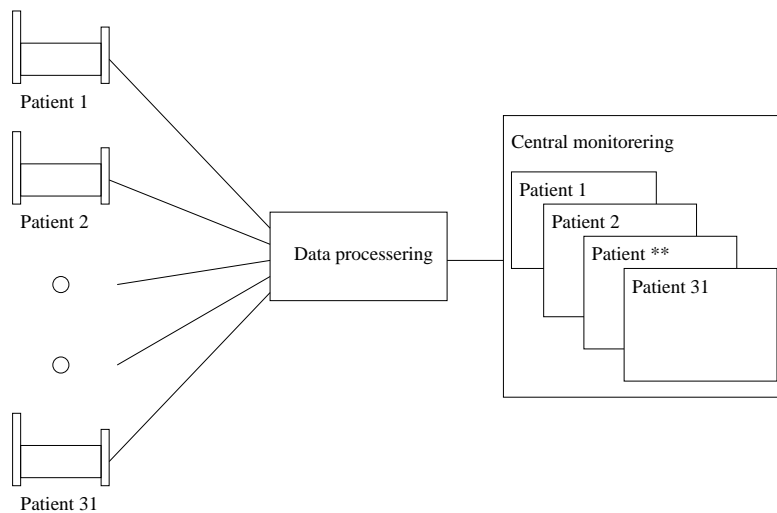
Besøget viste, at EKG som metode primært bruges til længerevarende monitorering af patienter, der skal diagnosticeres og behandles. Lægen/sygeplejersken kan realtime overvåge patienten, og lægen har desuden mulighed for at hente det optagne og gemte EKG frem, når han skal overveje den videre behandling. Dermed kan hans vurdering af hjertets funktion baseres på data fra en længere tidsperiode, i stedet for blot et tidsmæssigt kort øjebliksbillede af, hvad der vil kunne være på en skærm. Systemet kan desuden alarmere personalet, hvis EKG'et indikerer hjertefejl, såkaldte events.

En sygeplejerske overvåger EKG centralt fra alle monitorerede patienter, og vurderer i tilfælde af alarm hvilke handlinger, der er nødvendige. Hvis der fx er tale om akut pumpevigt skal hjertets normale pumpefunktion genoprettes øjeblikkeligt ved hjælp af stød (defibrillering).

Der kan også være tale om mindre akutte events, fx hvis hjertet slår for hurtigt eller langsomt. Dette er ikke akutte trusler mod patientens liv, men er u hensigtsmæssige på længere sigt.

EKG-systemet på kardiologisk afdeling har forskellige funktioner, der understøtter alarmering og diagnosticering vha. EKG, og en del af disse alarmeringsfunktioner er også tilgængelige fra en bed-side del, der viser patientens EKG og eventuelle alarmer. Denne bedside del af systemet er nødvendig for, at andet personale end den sygeplejerske, der overvåger centralt, kan gribe ind i tilfælde af alarmer. Desuden kan patientens tilstand følges, når der arbejdes med patienten bed-side, fx efter defibrillering eller ved almindelig pleje.

På figur 1.1 ses informationsflowet af EKG fra patientafsnittene til den centrale overvågning, som det ser ud på Aalborg Sygehus Syd.



*Figur 1.1: På kardiologisk afdeling findes 31 sengepladser. EKG fra alle patienterne opsamles og vises på 4 monitorer lokaliseret i en overvågningsstue, der bemands af en sygeplejerske*

### 1.3 Problemformulering

De læger og sygeplejersker, projektgruppen var i kontakt med under besøget, gav udtryk for tilfredshed med de eksisterende EKG-systemer på afdelingerne og således de funktionaliteter og arbejdsgange, der er beskrevet i ovenstående afsnit 1.2.3 og yderligere i appendiks A. Det vælges derfor, at det udformede EKG-system på de væsentligste punkter skal kunne bruges identisk med systemet anvendt på kardiologisk afdeling.

**Et funktionelt system til opsamling, behandling og visualisering af EKG og puls skal analyseres, designes og implementeres på en microcontroller og pc. Systemet skal designes med henblik på anvendelse på en kardiologisk afdeling, hvor nøjagtig og præcis visualisering af EKG og detektering af arytmier er en nødvendighed.**

**EKG, puls og alarmer skal vises både bed-side og centralt, og alarmer skal fremtræde både akustisk og visuelt.**

**Det vurderes, at det ikke er relevant, at systemet kan benyttes samtidig med defibrillering eller af patienter med pacemaker, da der kendes faste designmetoder til løsning af denne type tilpasningsproblemer.**

Den nøjagtige funktion af og muligheder med projektgruppens EKG-system fastlægges i den følgende analyse.





# DEL II ANALYSE

*I den følgende del af rapporten analyseres brugerønsker til et funktionelt EKG-system. På grundlag af interview, litteraturstudie, samt egne overvejelser/undersøgelser, bliver ønskerne opstillet som reelle krav til et EKG system.*

*Analysen vil blive indledt ved en beskrivelse af det valgte biologiske signal, EKG, i kapitel 2.*

*Hernæst analyseres de undersøgte brugerønsker, og en produktspecifikation opstilles i kapitel 3*

*På grundlag af produktspecifikationen foretages et foreløbigt systemdesign i kapitel 4, og der opstilles krav til de designede processer og moduler.*

*Inden opstillingen af den endelige kravspecifikation i kapitel 6 foretages en afgrænsning af systemets omfang i kapitel 5, så implementationsopgaven er overkommelig indenfor projektperioden.*



# Kapitel 2

## EKG - et biologisk signal

*I dette kapitel beskrives EKG-signalets fysiologi. Dets oprindelse og udseende gennemgås, og i appendiks B beskrives forskellige opsamlingsmetoder af EKG.*

### 2.1 EKG-signalet

Når hjertets muskelceller depolariserer i forbindelse med kontrahering, opstår der spændingsændringer, som kan detekteres på kroppens overflade. Det vil sige, at det aktionspotentiale, der udbreder sig gennem hjertet og får det til at kontrahere, resulterer i et målbart, elektrisk potentiale [1]. Det er dette elektriske potentiale, der kaldes EKG. Et typisk EKG signal kan ses på figur 2.1.



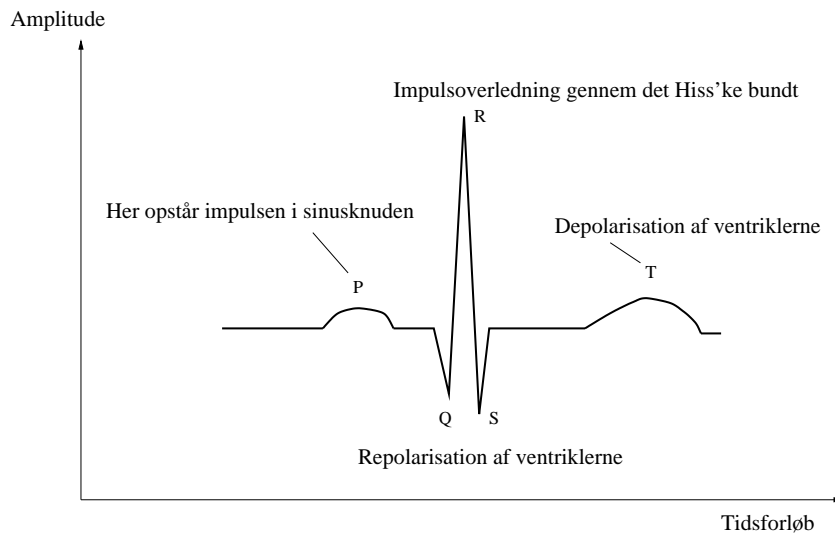
**Figur 2.1:** Figuren viser de elektriske potentialeændringer, der pga. hjertets aktivitet kan måles, opsamles og visualiseres. Ad  $x$ -aksen er den forløbne tid (25 mm/s), ad  $y$ -aksen amplituden af signalet (10mV/10 mm)

De spændingsændringer, der ses på EKG'et, skyldes væsentlige ændringer i polariseringen af muskelmassen i hjertets kamre, dvs. repolarisering eller depolarisering. I de tidsrum, hvor hjertets kamre er helt repolariserede eller depolariserede, vil der på et EKG ses en isoelektrisk linie, altså en vandret linie.

#### 2.1.1 Sammenhæng mellem EKG og hjertecyklus.

På figur 2.2 ses et fortegnat EKG. EKG'et består af PQRST-takkerne, der er de spændingsændringer, der kan måles i en hjertecyklus, dvs. ved et enkelt pulsslag. På figur 2.2 ses sammenhængen mellem tid og amplitude af den udbredte impuls i hjertet. Sammenhængen mellem udseende af EKG'et på figuren og den normale hjertecyklus beskrives i det følgende [1].

**Sinusknuden** Sinusknuden fyrer det pacemakerpotentiale, der udbredes i hjertet og som er startskuddet til hele hjerterytmen. Sinusknuden reguleres af det sympatiske og parasympatiske nervesystem via vagusnerven, men vil også være aktiv i tilfælde af defekt på nervesystemet.



**Figur 2.2:** På figuren ses de forskellige takker i et EKG-signal.

Øget sympatisk aktivitet frigiver noradrenalin, hvilket *øger* hjertefrekvensen, og øget parasympatisk aktivitet frigiver acetylcholin, hvilket *nedsætter* hjertefrekvensen. Fordi cellerne i hjertet er syncytialt forbundet, vil udbredelsen af impulsen fra sinusknuden automatisk blive udbredt i atriemuskulaturens fibre.

Selve sinusknudens firing ses ikke på EKG'et, da signalet er meget svagt sammenlignet med de- og repolariseringerne, der opstår efterfølgende.

Sinusknuden har en spontan frekvens på 70-80 Hz.

**P-takken** viser, at der sker en depolarisering af atriernes muskulatur, startende med højre atrium. Atrierne kontraherer på grund af depolariseringen; først kontraherer den superiore del, dernæst den inferiore del, så blodet pga. det øgede tryk presses gennem atrioventrikulærklapperne ned i ventriklernerne.

**AV-knuden** I PQ-intervallet ses en isoelektrisk overgang, der skyldes depolarisering af atrioventrikulærknuden (AV-knuden), samt impulsudbredelsen ned gennem det Hiss'ke bundt. AV-knuden har samme funktion på ventriklernerne som sinusknuden har på atrierne, men AV-knuden forsinker impulsudbredelsen. Det medfører at aktiveringen af ventriklernerne forsinkes, og dette gør, at atrierne har mulighed for at nå at tømme indholdet ned i ventriklernerne, inden ventriklernerne kontraherer.

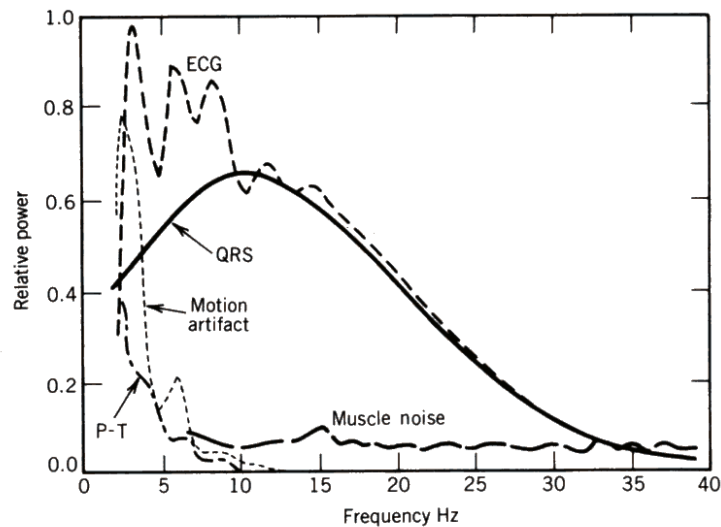
AV-knuden har en spontan frekvens på 40-60 Hz.

PQ-intervallet ses fra P-takkens begyndelse til Q takkens begyndelse og har en varighed på 0,15 sek.

**QRS-komplekset** viser ventriklernes depolarisering. I QRS-komplekset betegnes den første negative tak Q, den følgende positive tak R, og den negative tak efter R-takken kaldes S-takken. Ikke alle EKG-målinger viser både Q, R og S, og QRS-komplekset varierer i udseende og størrelse, afhængigt af hvordan impulsen i hjertet udbredes, samt hvordan EKG'et måles. Depolariseringen af ventriklernerne får dem til at kontrahere, og pga. det Hiss'ke bundts forløb i hjertet, kontraherer først den inferiore del af ventriklernerne, derefter den superiore, således at blodet presses ud i lungearterien (A. pulmonalis) og aorta.

I løbet af QRS-komplekset repolariseres atrierne, men dette er en lille spændingsændring sammenlignet med ventriklernes depolarisering, så repolariseringen er ikke synlig på EKG-signalet.

Varigheden af QRS-komplekset er ca. 0,10 sek.



**Figur 2.3:** Figuren viser de enkelte frekvenskomponenter i et EKG signal. Til diagnostisk anvendelse benyttes frekvensområdet 0,05-150 Hz. Til ambulant monitorering benyttes en mere restriktiv båndbredde. Fordelen, der opnås ved at benytte en begrænset båndbredde, er en mere stabil pulsdetektering, idet højfrekvent muskelstøj samt lavfrekvente bevægelsesartefakter frafiltreres. Fra [3]

**ST-segmentet** Mellem S- og T-takkerne ses endnu en isoelektrisk overgang som skyldes fuld ventrikulær depolarisering.

**T-takken** viser repolariseringen af ventriklernes.

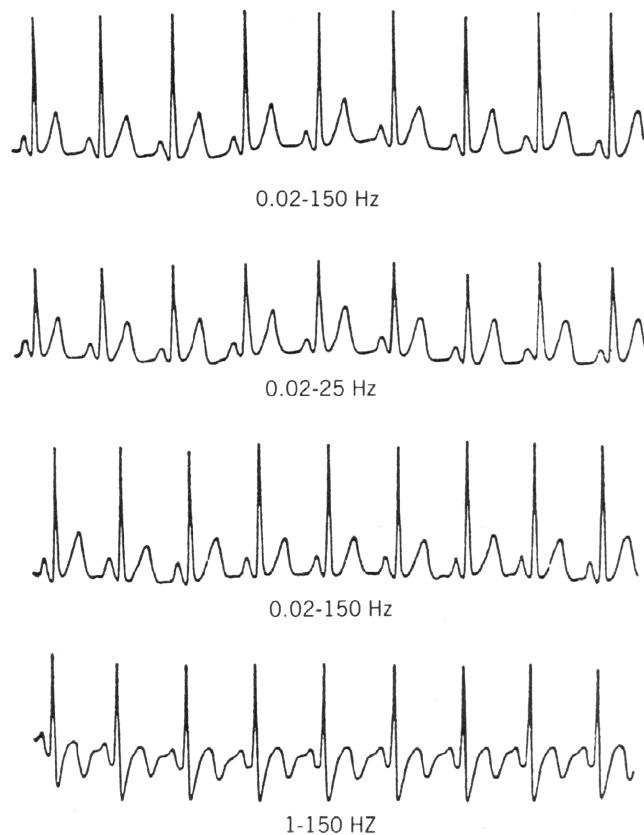
## 2.2 Størrelser af EKG-signalet

EKG signalet er blevet undersøgt for at opnå kendskab til amplituden af signalet og frekvensindholdet. Disse størrelser er nødvendige at analysere inden systemudvikling starter, således at disse erfaringer kan benyttes efterfølgende i designfasen.

Amplituden af EKG signalet angives i [2] til at ligge i området 0,5-4 mV. Signalet er ikke symmetrisk omkring base-linien, men afhænger af, hvilken afledning der betragtes. I bilag A findes et EKG-signal fra en af gruppens medlemmer opsamlet på Aalborg Sygehus Syd. Her ses det tydeligt, at amplitudeudsvinget er afhængigt af afledningen. For afledningerne 1-3 er udsvinget i positiv retning ca. 2 mV og ca. 0,5 mV i negativ retning. Afhængigt af, hvordan signalet opsamles, er det samlede udsving på op til 4 mV [2]. Se appendiks B for beskrivelse af de forskellige måder at opsamle EKG-signalet på.

Det interessante frekvensområde for EKG-signalet til diagnostisk anvendelse er 0,05-150 Hz. Til monitorering benyttes ofte frekvensområdet 0,5-40 Hz [3].

På fig. 2.3 ses et typisk powerspektrum for en person, der bevæger sig. EKG'ets komponenter, P-takken, QRS-komplekset og T-takken er angivet hver for sig. Det er tydeligt, at QRS-komplekset er koncentreret i frekvensområdet under 40 Hz, men der er stadig komponenter af QRS-komplekset op til 150 Hz. Energien for P og T-takkerne er koncentreret i det lave frekvensområde under 5 Hz, og lavfrekvente bevægelsesartefakter ligger i samme område. I anvendelser, hvor frekvensområdet 0,5-40 Hz benyttes, opnås den fordel, at en stabil puls kan detekteres fra QRS-komplekset, idet en stor del af de bevægelsesrelaterede frekvenser frafiltreres, samtidig med at muskelstøj reduceres. I kardiotalymetre, hvor det kun er pulsen, der bestemmes, kan en båndbredde på helt ned til 8-21 Hz anvendes [3].



**Figur 2.4:** Figuren viser hvorledes EKG signalet forvrænges, når en begrænset båndbredde anvendes. Overfiltrering bør undgås, hvis EKG signalet skal bruges til diagnostisk anvendelse. Fra [3].

Hvis der ikke vælges den rette båndbredde for det filter, der designes, forvrænges EKG signalet. Øges den nedre knæfrekvens, forvrænges P og T-takkerne samt S-T segmentet. Sænkes den øvre knæfrekvens, forvrænges QRS-komplekset. Fig. 2.4 viser en sammenligning af et behandlet signal, båndbredde 0,02-150 Hz, med to forvrængede signaler, hvor båndbredden er begrænset.

## Kapitel 3

# Opstilling af produktspecifikation

*I dette kapitel fastlægges det, hvilke muligheder en bruger skal have med EKG-systemet, som projektgruppen søger at udforme. Valgene tager udgangspunkt i ønsker præsenteret af personale på kardiologisk og intensiv afdeling på Aalborg Sygehus Syd, samt de ressourcemæssige muligheder for at efterkomme kravene.*

*Kapitlet munder ud i den egentlige produktspecifikation, der beskriver funktioner og muligheder, der ønskes tilgængelige i EKG-systemet.*

### 3.1 Funktionaliteter

For lægen eller sygeplejersken, der skal anvende EKG som værktøj, er adgang til at se EKG fra patienten naturligvis essentiel. En uomtvistelig nødvendig funktion ved ethvert EKG-apparat og altså også apparatet, der skal udvikles i dette projekt, er derfor, at EKG skal optages kontinuerligt over længere tid (timer, dage). Desuden skal det vises real-time, således at personalet på afdelingen kan se og handle ud fra EKG'ets udseende. Desuden er det for især lægens vurdering af patientens tilstand nødvendigt, at han på et vilkårligt tidspunkt kan hente et tidligere optaget EKG frem og granske det [4].

Et system, der optager EKG, viser det real-time og senere som et billede, kan desuden have forskellige ekstrafunktioner alt efter systemets specifikke anvendelse. Der er to hovedgrupper af ekstrafunktioner:

- 1: Mønstergenkendelsesalgoritmer til detektering og diagnose af forskellige arytmier
- 2: Algoritmer til at beregne forskellige parametre som fx puls.

På en kardiologisk afdeling er personalet i høj grad eksperter i hjertesygdomme, så både diagnosticerende og behandlende læger, kan aflæse EKG og identificere afvigelserne fra et normalt, sundt EKG. Derfor vælger projektgruppen at koncentrere arbejdet om at præsentere lægen/sygeplejersken for data på den bedst mulige måde, fremfor at give decideret eksperthjælp til vurderingen; hjælpen er ikke nødvendig, når der er tale om eksperter på det kardiologiske område [4]. Hermed er det valgt at begrænse de oplysninger systemet selv skal uddrage fra det optagne EKG til et minimum, som omhandler tilstande, der kræver omgående indsats, hvis ikke patienten skal miste livet, samt puls.

#### 3.1.1 Akutte alarmer ved pumpevigt

Omgående indsats er krævet ved tilstande, hvor cardiac output (CO) falder til under en vis grænse, således at væv og hjerne ikke forsynes med tilstrækkeligt ilt.

Pludseligt lavt CO kan hos hjertepatienter skyldes to fysiologiske tilstande, såkaldte arytmier:

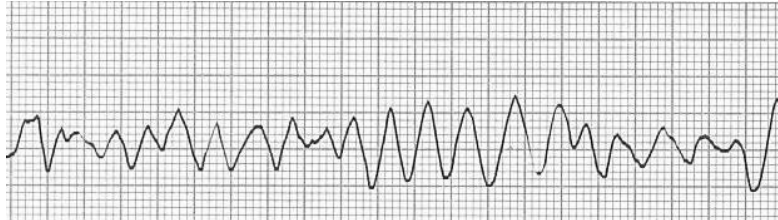


**1: Ventrikelflimmer****2: Asystoli**

Tilstand 1 og tilstand 2 er vist på figur 3.1 og 3.2.

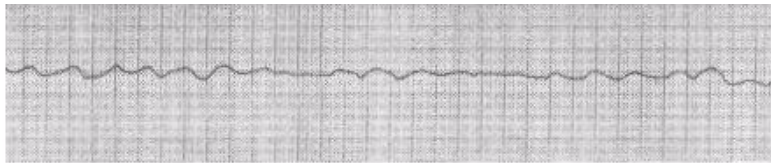
Ventrikelflimmer betyder, at ventriklerne ikke kontraherer systematisk, men i stedet tilfældigt som i krampe med en frekvens på helt op imod 600 slag pr. minut [5]. Det betyder, at kontraheringen ikke er tilstrækkelig lang og kraftfuldt nok til at tømme ventriklerne ud i aorta og A. pulmonalis, med lav slagvolumen og dermed lavt cardiac output til følge.

Det ses på figur 3.1, at komplekserne er afrundede, der er ingen isoelektriske intervaller mellem dem, og de forskellige takker er ikke til at skelne fra hinanden.



**Figur 3.1:** Figuren viser EKG'ets udseende ved ventrikelflimmer. X-aksen 25 mm/sek, Y-aksen 10mV/10 mm

Asystoli er identisk med betegnelsen hjertestop. Ventriklerne arbejder således ikke, og når ventriklerne ikke arbejder, er CO naturligvis nul. På figur 3.2 ses, at EKG'et ved asystoli er en næsten isoelektrisk linie.



**Figur 3.2:** På figuren ses EKG'ets udseende ved asystoli. Ad x-aksen 25 mm/s, ad y-aksen 10 mV/10 mm

Ved de to ovennævnte arytmier kræves indsats inden for 5 minutter, hvis patienten skal overleve uden varige men [4], så det er af afgørende betydning, at arytmien opdages omgående. På en kardiologisk afdeling med plads til 31 patienter, som det er tilfældet i Aalborg, betyder dette, at den sygeplejerske, der overvåger patienternes EKG, skal "have et øje på hver finger", hvis hun skal kunne holde øje med alle patienter på én gang. Alternativt skal hun have hjælp fra EKG-udstyret, således at der gives en alarm i tilfælde af ventrikelflimmer eller asystoli, hvilket da også er tilfældet med det udstyr, der anvendes, og Lars Hvilsted Rasmussen, overlæge på kardiologisk afd., kan ikke forestille sig et EKG-system uden alarmering for ventrikelflimmer og asystoli. Dette understreger nødvendigheden af netop disse to alarmer i ethvert EKG-system.

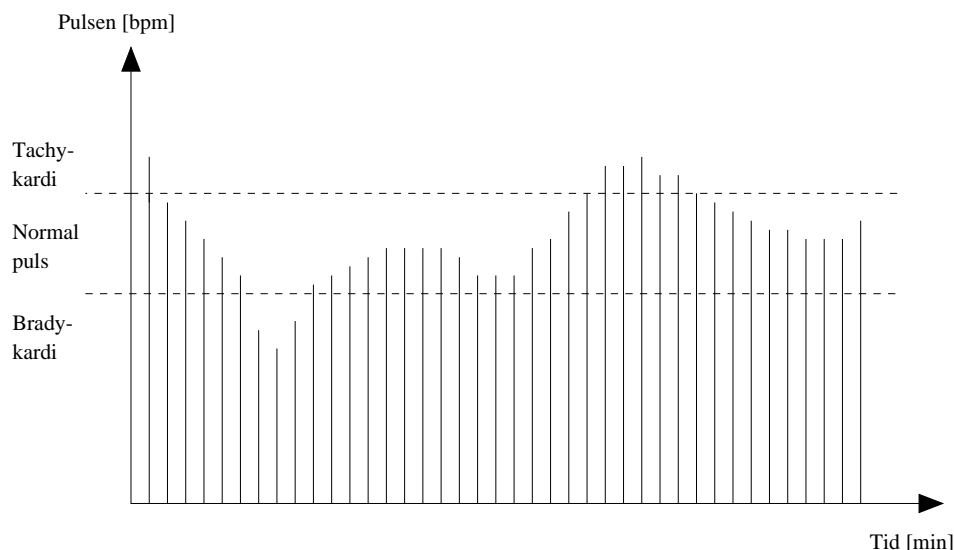
### 3.1.2 Parametre fra EKG samt kronologiske fordelinger af events

Ud over de to akut livstruende arytmier, som kræver omgående alarm og indsats, er der parametre, som kan være svære at identificere med det blotte, omend trænede, øje, og hjælp fra systemet er da ønskeligt. Projektgruppen vælger at afgrænse de parametre systemet skal beregne, til alene at være puls-relaterede, samt vise den kronologiske fordeling af de events, der måtte forekomme.

#### Pulsberegning og trendkurve

Puls er en af de vitalparametre, en læge eller sygeplejerske bruger til at fastslå patientens tilstand ud fra, så projektgruppen vælger, at EKG-systemet skal udføre pulsregning og vise pulsen.

Pulsens varierende over en periode kan være diagnostisk interessant, så det vælges, at systemet også skal kunne vise trendkurve for puls. Et eksempel er givet i figur 3.3



**Figur 3.3:** Figuren viser den beregnede puls som funktion af tiden. Med et passende tidsinterval plottes den beregnede puls.

### Alarm ved unormal puls

Unormal høj eller lav puls - hhv. tachykardi og bradykardi - indikerer fejlfunktion i hjertet, og selv om disse to arytmier ikke er akut livstruende, er det tilstande, der på længere sigt er uhenigtsmæssige, enten pga. for stor belastning af hjertemuskulaturen eller pga. for lav ilttilførsel til det systemiske væv. Derfor er alarm også ønskelig på tachykardi og bradykardi, og da pulsen allerede beregnes, kan dette krav let imødekommes. På figur 3.3 ses tilfælde af tachy- og bradykardi tydeligt på en trendkurve. Forskellige patienter har forskellige normal-områder for puls, så definitionerne af tachy- og bradykardi skal kunne justeres individuelt i form af øvre og nedre grænseværdier.

### Markering af events

Det kan være af diagnostisk og behandlingsmæssig interesse, hvordan alarmerne tidsmæssigt er fordelt, så det vælges at markere på EKG'et, hvis der har været en alarm. Det er så let for personalet at hente EKG'et frem og identificere de tidspunkter, hvor der har været alarm. De events, der skal markeres er asystoli, ventrikelflimmer, tachykardi og bradykardi.

### 3.1.3 Visning af EKG for alle patienter

Udseendet af EKG kan variere fra patient til patient, fx som følge af forskellig vævsdensitet eller eventuelle fejl i hjertefunktion. For at opnå de ønskede funktionaliteter i systemet er det derfor nødvendigt, at systemet tilpasser sig disse individuelle forskelle. Denne tilpasning skal, foregå automatisk, så personalet ikke skal bruge tid på indstilling.

## 3.2 Adgang til funktionaliteter hos patient og centralt

Behovet for at kunne monitorere både centralt og hos patienten selv, ses bedst af situationen på kardiologisk afdeling, hvor antallet af sygeplejersker og læger ikke svarer til antallet af patienter. Lægen og sygeplejersken kan således ikke altid være hos patienten, men skal alligevel

kunne overvåge patientens tilstand. Når personalet arbejder ved patienten, skal der også kunne overvåges, og derfor er det nødvendigt at have adgang til patientens EKG både centralt og bedside. Bed-side er det relevant med adgang til de oplysninger, der beskriver patientens tilstand her og nu, mens der centralt skal gives et mere detaljeret billede til brug for både monitorering og senere granskning af EKG.

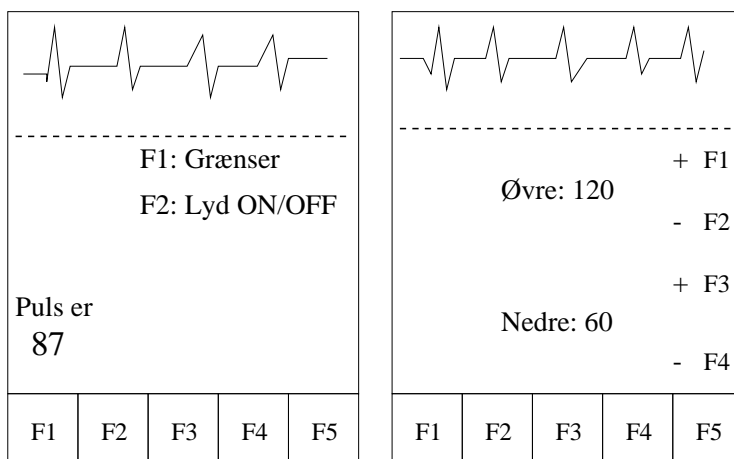
### 3.2.1 Bed-side

Bed-side er det hensigtsmæssigt med et display, der giver viser en af afledninger fra EKG-signalet med tilhørende værdier for puls, samt mulighed for at indstille grænseværdier for tachy- og bradykardi. Visningen på displayet ses ved et kontinuert real-time EKG samt pulsværdi. Desuden gives en menu med mulighed for at ændre på grænser for pulsalarmer samt lydindstillinger.

#### Visualisering bedside

På Aalborg Sygehus Syd, vises der bedside en enkelt af de optagne 3 afledninger, på en monitor på væggen. Se appendiks B for uddybende forklaring af opsamlingsmetoder. Det er sygeplejersken ved centralskærmen, som afgør hvilken af afledningerne, der skal vises. Oftests er de kardiologiske læger interesserede i P-takkens udseende, hvorfor den afledning, som tydeligst viser denne, visualiseres bedside [4].

Der ønskes på displayet desuden en klar indikation af pulsen, og hvert pulsslæg skal samtidig indikeres ved blink fra en diode. Et skelet for opbygningen af displayet er vist på figur 3.4.



(a) Figuren viser skelettet for opbygning af bedside displayet med visning af puls og Einthovens II afledning. Denne skærm tænkes vist som hovedskærm.

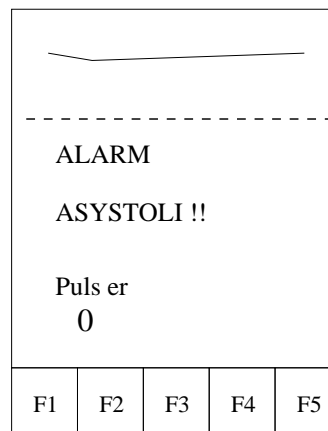
(b) Figuren viser skelettet for opbygning bedside displayet ved indstilling af grænser for pulsalarm.

**Figur 3.4:** Eksemplificering af display bedside.

Ændring af grænser for arytmierne tachy- og bradykardialarmer placeres bedside. Dette skyldes sikkerhedsmæssige årsager, idet en sygeplejerske, der centralt overvåger 30 patienter, ved en fejl kan komme til at justere på den forkerte patient. Der må ikke kunne ændres detekteringskriterier for arytmierne ventrikelflimmer og asystoli, da disse er akut livstruende og kræver omgående behandling.

## Alarmeringen bed-side

Reaktion fra personalet på alarmering er en nødvendighed, og derfor skal der både gives alarm ved hjælp af lyd, lys samt tekstindikation. Af hensyn til personalets videre behandling skal der på displayet stå, hvilken alarm der er blevet detekteret. Samtidig skal der være en hyletone, som straks fanger sygeplejerskens opmærksomhed. På figur 3.5 ses et skelet af displayets udseende i tilfælde af en alarm for asystoli.



*Figur 3.5: Figuren viser skelettet for opbygning af bed-side displayet i tilfælde af alarm - her asystoli*

Ikke alle arytmier er lige akutte, så sygeplejersken skal ud fra den akustiske alarm kunne kende forskel på akut livstruende og ikke-akutte alarmer. Derfor skal der være to typer af alarm-lyd; en meget gennemtrængende ved asystoli eller ventrikelflimmer og en mindre gennemtrængende ved tachy- eller bradykardi.

Akustisk alarmindikation er en god sikkerhedsforanstaltning, men i nogle situationer er det ikke passende med en hyletone, hver gang en grænseværdi overskredes. Både på Kardiologisk afdeling og Thorax-Intensiv afdeling (herefter TIA), sigter man nemlig mod at skabe en naturlig døgnrytme, så patienterne får ro til at sove om natten. Derfor skal der være en mulighed for at slå lyden af alle typer alarmer til og fra. Alarmen skal dog stadig indikeres, hvilket så kun gøres med lysdioder og tekst. På samme måde vil det være hensigtsmæssigt, at lyden kan slås fra i et kort tidsrum, når man arbejder hos patienten. På Aalborg Sygehus Syd er det i korte tidsrum af 2 minutter, lyden kan slås fra. På displayet skal det vises, om lyden er slået fra i 2 minutter eller permanent.

## 3.2.2 Centralskærm

Ved centralskærmen skal foruden de elementære parametre som puls, alarm m.m., gives de muligheder, som ikke har været hverken hensigtsmæssige eller nødvendige at vise bed-side, men som er relevante i forbindelse med monitorering og granskning.

### Visualisering ved centralskærmen

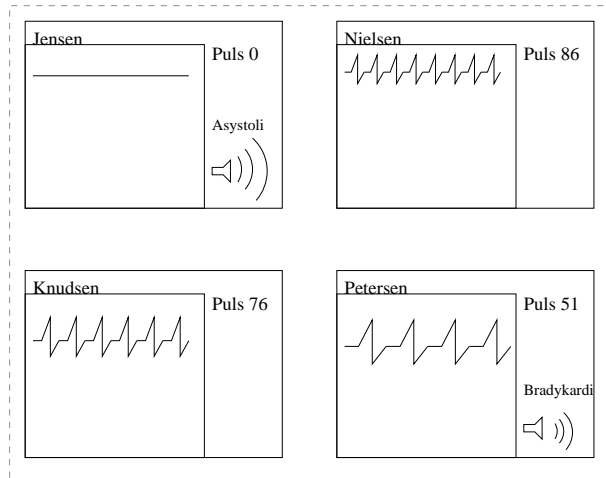
De optagede og beregnede afledninger - Einthoven og Goldberg B - skal alle vises på centralskærmen, hvor lægen og sygeplejersken har mulighed for nøje at studere patientens EKG.

Af praktiske hensyn skal der være mulighed for at indtaste patientdata, når en ny patient bliver indlagt, fx CPR-nummer, navn, indlæggelsesdato m.m.

Der ses et behov for mulighed for anvendelse af to skærmfunktioner i forbindelse med centralskærmen; en dynamisk, der bruges ved monitorering, og en statisk, der bruges ved granskning.

*Dynamisk skærm*

Der skal altid vises en standard, dynamisk skærm for hver patient, som viser de af sygeplejersken valgte afledninger samt pulsen og en indikation af hvert pulsslæg i form af blink. Desuden skal de indstillede pulsgrænser vises. Dette er vist som eksempel i figur 3.6



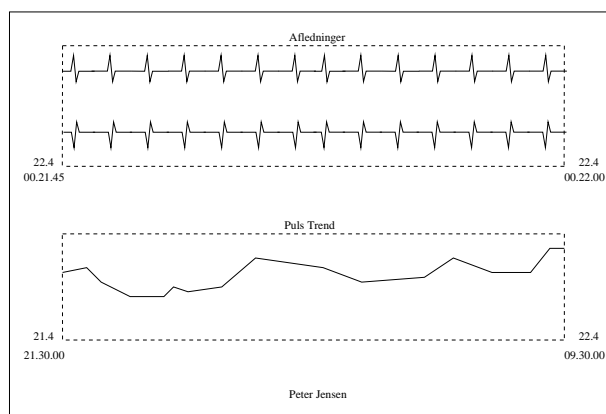
**Figur 3.6:** På figuren ses skelettet for opbygningen af den dynamiske skærm, så flere patienter kan overvåges samtidig. Der ses forskellige tilstande ved forskellige patienter; asystoli giver højere alarm-lyd end bradykardi

**Statisk skærm**

På den statiske skærm skal lægen eller sygeplejersken have mulighed for at bestemme hvilke afledninger, de vil se og en zoomfunktion skal hjælpe til at se detaljer på EKG’et i forbindelse med granskning.

Trends for puls og alarmer skal først vises, hvis lægen/sygeplejersken vælger det, da de ikke altid er relevante.

Alarmer skal indikeres på det optagne EKG, hvor indikationen kan foregå ved en farvning af det pågældende alarmområde. Her skal farven variere afhængig af alarmtypen, så lægen ikke behøver at undersøge hele EKG-optagelsen, men i stedet se efter farve-indikationerne. Den statiske skærm er vist som et eksempel i figur 3.7.



**Figur 3.7:** På figuren ses principperne i den statiske skærm. Der er i dette tilfælde valgt 2 afledninger fra EKG, optaget mellem kl. 00.21.45 og 00.22.00. Pulstrend vises for den beregnede puls mellem kl. 21.30 og 09.30. Der har ikke været alarm mellem kl. 00.21.45 og 00.22.00, så en sådan er ikke markeret.

**Lagring**

Før lægen kan hente et tidligere optaget EKG, skal EKG’et naturligvis være gemt. Af hensyn til lagringskapaciteten lagres i apparatet anvendt på kardiologisk afdeling ikke data for mere end 72 timer iht. appendiks A.

## Alarmering ved centralskærmen

Monitorering centralt sker med henblik på at detektere og handle på alarmkrævende situationer. Derfor skal der alarmeres i de samme situationer og med samme indikationer som bed-side, dvs. med tekstindikation, lys og akustisk alarm.

Ved centralskærmen er det altafgørende, at der reageres på en alarm, og derfor er det ikke hensigtsmæssigt, hvis alarmeringsfunktionen kan slås fra. En alarm skal derfor fremtræde klart, hver gang en alarm optræder. Afledninger skal vises sammen med pulsen og tekstindikation typen af detekteret arytmie.

I tilfælde af alarm hos 2 eller flere patienter på samme tid er det ikke muligt for sygeplejersken at vurdere alle alarmer samtidig, og der skal derfor designes en "alarm-bank", hvor alle alarmer inden for et vist tidsrum gemmes, så de efterfølgende hurtigt kan hentes frem til overvejelse.

## 3.3 Produktspecifikation

- Bed-side monitorering
  - Systemet ønskes at tilpasse sig alle patienter, således at puls kan detekteres troværdigt
  - For bed-side monitoren ønskes, at EKG vises i form af Einthovens II afledede. Pulsen ønskes vist sammen med EKG, og hvert pulsslæg skal markeres af blink fra en diode.
  - Alarmfunktion for arytmierne tachy- og bradykardi samt asystoli og ventrikelflimmer. Alarmer ønskes indikeret ved lys, lyd og tekst.
  - Forskellig lydintensitet for alarmer for hhv. tachy- og bradykardi og asystoli og ventrikelflimmer: mest dominerende lydintensitet for asystoli og ventrikelflimmer.
  - Menufunktion med mulighed for ændring af alarmgrænser for tachy- og bradykardi. On/Off mulighed for alarmering ved lyd bedside.
  - 2 minutters mute funktionalitet.
  - Indikation på display, at lyd er slået fra, både ved 2 minutters mute og permanent deaktivering af lydalarm.
- Centralmonitorering
  - Mulighed for visning af Einthovens og Goldbergs i alt 6 afledninger på statisk skærm.
  - Mulighed for visning af en dynamisk skærm med en afledning for hver patient.
  - En zoom-funktion skal være tilgængelig på den statiske skærm.
  - Alarmering ved tachy- og bradykardi, asystoli og ventrikelflimmer.
  - Mulighed for visning af tidligere optaget EKG med markering af et antal detekterede alarmer. Visning af pulstrend samt trends for arytmier.
  - Skal lagre målt EKG for 72 timer.
  - Alarmfunktion ved lyd, lys og tekst. Alarmering ved lyd må ikke kunne slås fra. Alarmer skal altid fremtræde klart.
  - Alarmgrænser for arytmier må ikke kunne ændres.
  - Puls ønskes beregnet og vist.
  - Nye alarmer ønskes gemt i en "alarm-bank", hvorfra de hurtigt kan åbnes og vurderes.



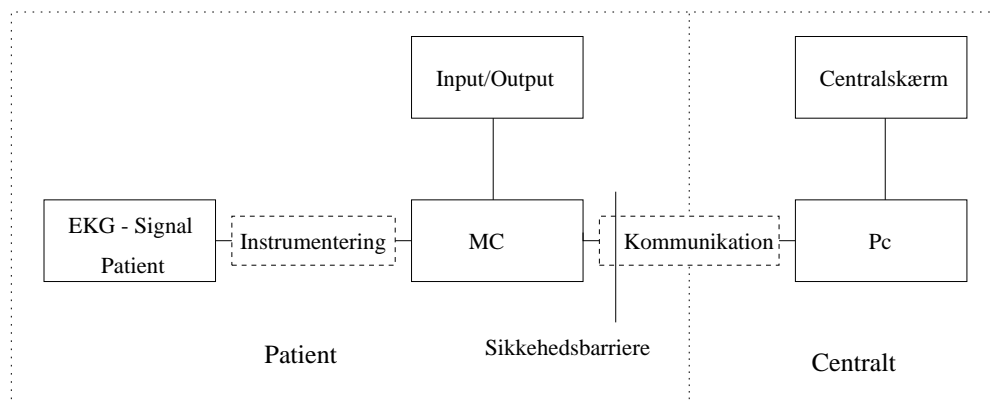
# Kapitel 4

## Systemdesign

I kapitlet udledes det, hvordan produktspecifikationen kan opfyldes gennem implementation af processer og moduler. Desuden fastlægges det til slut, hvilke dele af produktspecifikationen det er tids- og ressourcemæssigt forsvarligt at arbejde videre med.

### 4.1 Inddeling i delsystemer

Systemet indeholder delsystemerne vist på fig. 4.1; begrundelse for opdelingen i delsystemer findes i det følgende.



**Figur 4.1:** Delsystemer i EKG-systemet

Patientens EKG-signal samt data uddraget herfra ønskes præsenteret for sygehuspersonalet på en bed-side monitor og en centralskærm, hvor der begge steder skal være mulighed for interaktion fra brugeren iht. kapitel 3. Der kræves for opfyldelse af studieordningen desuden anvendelse af 3 hardwaremæssige elementer; en microcontroller, herefter mc, MSP430F149 fra Texas Instruments, en pc, samt en kommunikationsforbindelse mellem disse.

Ud fra fig. 4.1 haves inddelingen i delsystemer:

EKG-signalet skal samples, så en form for instrumentering som forberedelse til dette er nødvendigt.

Mc'en er kernen i bed-sidemonitoren og varetager al signalbehandling og beregning her.

Pc bruges som grundsten i det centrale delsystem.

Et bindeled i form af seriel kommunikation mellem mc og pc benyttes for at få data fra mc til pc.

I analysen af delsystemerne er der for delsystemet mc desuden foretaget inddeling i processer, der videre inddeles i moduler. For delsystemet pc er der kun foretaget inddeling i processer.



Det er nemlig hverken muligt eller relevant at inddele pc-programmet i egentlige moduler, da SPU er udviklet til imperative sprog som C, og pc-delsystemet ikke programmeres i et imperativt sprog. Begrundelse herfor findes i kap. 14.

## 4.2 Valg af opsamlingsmetode

På kardiologisk afdeling, Aalborg Sygehus Syd, benyttes Einthovens 3 afledninger, når patientens EKG skal vurderes, mens der på andre afdelinger, som fx intensiv-afdelingen benyttes i alt 12 afledninger, se appendiks B.

Grunden til, at der på kardiologisk afdeling ikke anvendes Wilsons afledningsmetode, skyldes at det ikke er praktisk for patienterne at gå rundt med de mange elektroder i døgndrift. I de tilfælde hvor lægen ønsker et mere fyldestgørende billede af hjertets funktion, er der på afdelingen adgang til en række EKG-apparater, der kan optage og udskrive Einthovens, Goldbergs og Wilson afledninger.

I dette studieprojekt er det ikke relevant at opsamle EKG fra 12 afledninger. Derfor vælges det i projektet kun at anvende Einthovens og Goldbergs model, der giver mulighed for i alt 6 afledninger. 6 afledninger vil ikke sætte begrænsninger i forhold til de funktionaliteter, der ønskes opnået - Muligheden for at detektere asystoli og ventrikelflimmer er bibeholdt, og pulsen kan detekteres ud fra en hvilken som helst afledning. Egentlig ville Einthovens model være tilstrækkelig, men de tre nye afledninger, der kan opnås med Goldbergs metode, kan fås ved simple matematiske modeller uden brug af ekstra elektroder/kanaler og de er således lette at implementere.

## 4.3 Instrumentering

Som nævnt ovenfor er det nødvendigt at behandle de opsamlede EKG-signaler analogt, inden de kan udgøre et brugbart input til mc'en. Denne behandling består i at forstærke signalet til et passende spændingsniveau, samt at frasortere uønskede frekvenser i signalet.

### 4.3.1 Forstærkning

Det typiske EKG signal fra Einthovens 3 afledninger spænder maksimalt fra -0.5 mV til +2 mV. Dette ses både i kapitel 2, samt bilag A over optaget EKG fra en af gruppens medlemmer. Eftersom mc'ens forsynings-spænding er 0 til 3,3 V, kan mc'ens indbyggede ADC konvertere input i spændingsområdet 0 til 3,3 V. Det står derfor klart, at signalet skal forstærkes væsentligt, hvis ADC'ens konverteringsområde skal udnyttes bedst muligt.

#### Forforstærkerens egenskaber

De 3 Einthoven-afledninger, der i afsnit 4.2 er valgt som input til systemet er alle bipolære afledninger. Der er altså tale om et differenssignal, og det er derfor et naturligt krav, at forforstærkningen skal foregå vha. en differensforstærker. For at opnå den bedst mulige forstærkning af det rene differenssignal, skal forforstærkeren så vidt muligt frasortere common mode signaler på indgangene. Der ønskes derfor en forforstærker med høj CMRR - ideelt set uendelig høj - men mindst 86 dB [3].

For yderligere at forbedre forforstærkerens CMRR designs et driven-right-leg-kredsløb som virtuel reference. Ifølge [2] kan et sådant kredsløb effektivt reducere forstærkningen af common mode signaler og dermed højne forforstærkerens CMRR.

Et driven-right-leg-kredsløb er et feedback-kredsløb, der forstærker det resterende common mode signal og sender det i modfase tilbage til reference-elektroden på kroppen - som oftest

højre ben - idet det antages, at common mode signalet er ens for alle elektroder. Impedansen mellem forstærkerens reference og reference-elektroden på kroppen er mindre i dette tilfælde, end hvis der benyttes en ikke-drevet reference-elektrode, og der opnås derfor en bedre reference med driven-right-leg [6]. Da kredsløbet er et feedback-kredsløb med deraf følgende fare for positiv feedback og fortsatte oscillationer er det vigtigt, at der tages højde for stabilitet ved designet.

Det er ønskeligt, at det opsamlede EKG-signal fra kroppen overføres 100 % til forforstærkerens indgange. Signalet vil imidlertid spændingsdeles mellem elektrode-/hud-impedansen og forforstærkerens indgangsimpedans og der vil derfor ske en dæmpning af signalet allerede inden forstærkningen. Det synes et rimeligt krav, at signalet kun må svækkes 1 %, og indgangsimpedansen skal derfor være 100 gange større end elektrode-/hud-impedansen. I følge [7] kan elektrode-/hud-impedansen i nogle tilfælde være helt op til 200 K $\Omega$ , og forforstærkerens indgangsimpedans skal derfor være mindst 20 M $\Omega$ .

### Gain og tilpasning til ADC

Som nævnt ovenfor befinder de valgte EKG-afledninger sig i området -0,5 mV til +2,0 mV - dvs. et spænd på 2,5 mV - og ADC'ens konverteringsområde er 0 til 3,3 V - altså et spænd på 3,3 V. For at sikre systemet mod uventet variation i inputsignalets størrelse undlades det at udnytte hele ADC'ens konverteringsområde. Det formodes at et samlet headroom på 25 % af konverteringsområdet er tilstrækkeligt - svarende til 0,8 V; dermed udnyttes et spænd på 2,5 V. Forstærkningsopgaven lyder derfor på at forstærke input-signalets spænd på 2,5 mV til et outputspænd på 2,5 V. Der skal altså forstærkes 1000 gange.

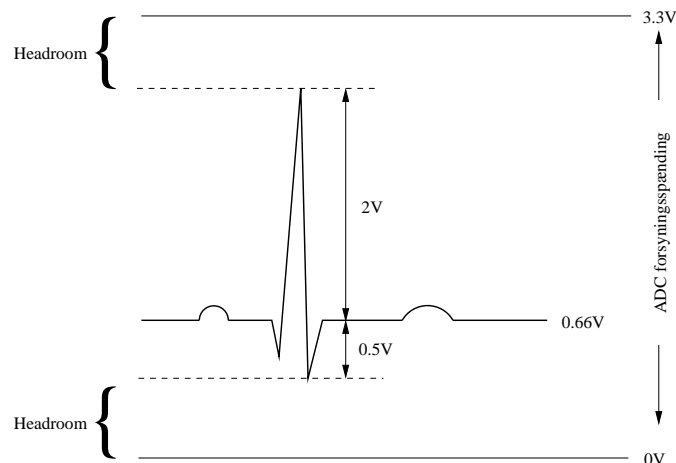
I projektet anvendes Ag/AgCl elektroder til opsamling af EKG signalet. Denne elektrodetype er velegnet til opsamling af biopotentialer som EKG og står til projektgruppens rådighed, i form af Medicotest Neuroline-elektroder. Half-cell potentialet over elektroderne kan være forskelligt og kan drifte i forskellig grad. Disse faktorer giver anledning til et indlejret varierende offset, der amplitudemæssigt kan være op til  $\pm 300$  mV [2]. For at undgå mætning af forforstærkeren foretages kun en brøkdel af de 1000 ganges forstærkning, inden signalet højpasfiltreres og DC-komponenten elimineres.

Systemet skal forsynes med batterispændinger på  $\pm 9$  V. Antages det, at forforstærkerens rails befinder sig max. 2 V under/over forsyningsspændingen, kan forstærkeren forstærke signaler til  $\pm 7$  V uden at mætte. Med et maksimalt offset på  $\pm 300$  mV og et signal på -0,5 mV til +2 mV kan der maksimalt forstærkes  $7 \text{ V} / 302 \text{ mV} \approx 23$  gange. Der vælges et gain på 20 for forforstærkeren.

Den resterende forstærkning skal foretages efter højpasfiltreringen, hvor DC offset elimineres, og det kan derfor med fordel foretages i et aktivt højpasfilter. For at opnå et samlet gain på 1000, skal filterets gain være 50.

Efter 1000 gange forstærkning spænder signalet fra -0,5 V til +2,0 V, og det er derfor nødvendigt at hæve DC-niveauet inden ADC'en. Signalet hæves, så det indarbejdede headroom fordeles forholdsmæssigt jævnt i bund og top. Da den positive del af signalet udgør 4/5 af amplituden, skal toppen af konverteringsområdet tildes 4/5 af headroomet på 0,8 V, og den negative del af signalet skal tildeles 1/5 af headroomet på 0,8 V svarende til 0,16 V i bunden af konverteringsområdet. Med 0,16 V headroom i bunden af konverteringsområdet og herpå 0,5 V negativt signal skal baselinen hæves i alt 0,66 V inden ADC'en, som det er vist på figur 4.2.

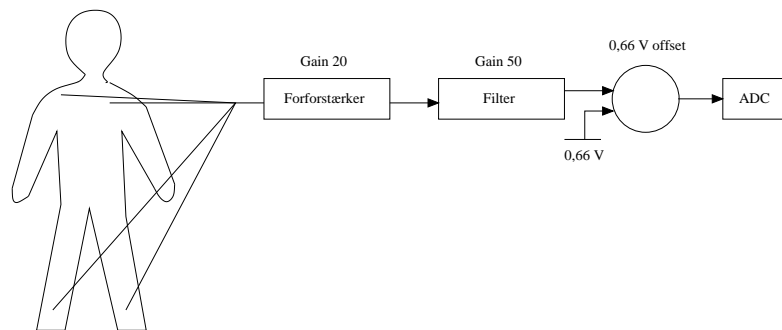
Kraftige EMG-artefakter ( $\pm 5$  V efter forstærkning) i samme frekvensbånd som EKG-signalet kan medføre, at grænserne for ADC'ens konverteringsområde overskrides. Mc'en er meget følsom overfor negative inputspændinger, der overskrider 0 V, og positive inputspændinger, der overskrider 3,3 V, og det kan derfor ikke tillades, at EMG-artefakterne slipper igennem til mc'en med fuld styrke. Der skal derfor designes et kredsløb, der kan sikre ADC'en mod overspændinger, så kun spændinger i området 0-3,3 V kan forekomme på ADC'ens indgange.



**Figur 4.2:** Efter forstærkning hæves DC-niveaulet +0,66 V, så hele signalet ligger i ADC’ens konverteringsområde. Dette giver 0,16 V headroom i bunden af konverteringsområdet og 0,64 V headroom i toppen af konverteringsområdet. Headroomet sikrer, at selv EKG-signaler med større amplitude end forventet konverteres korrekt.

Det er vigtigt at beskytte patienten mod mulige stød ved brug af apparatet. Ifølge IEC601-1-1-standarden, må patienten ved “single fault conditions” maksimalt udsættes for strømme på 0,5 mA, jvf IEC 1992, stk. 16.201 [8]. Der skal derfor designses strømbegrænsende foranstaltninger mellem patient og forforstærker, der sikrer, at denne grænse overholdes.

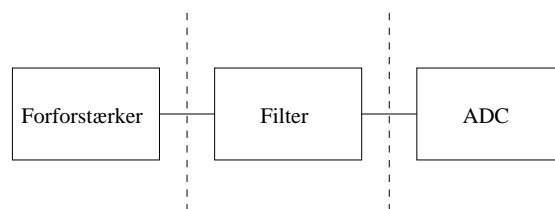
Instrumenteringen fra patient til ADC fremgår af figur 4.3.



**Figur 4.3:** Signalet forstærkes 20 gange i forforstærkeren og yderligere 50 gange i filter-sektionen. Herpå hæves niveaulet med 0,66 V, så det passer til ADC’ens konverteringsområde.

### 4.3.2 Filtrering

Det nødvendigt at filtrere signalet inden mc’en kan behandle det. Filterets grænseflader er derfor forforstærkeren og AD-konverteren som det ses på figur 4.4



**Figur 4.4:** På figuren skitseres filterets grænseflader.

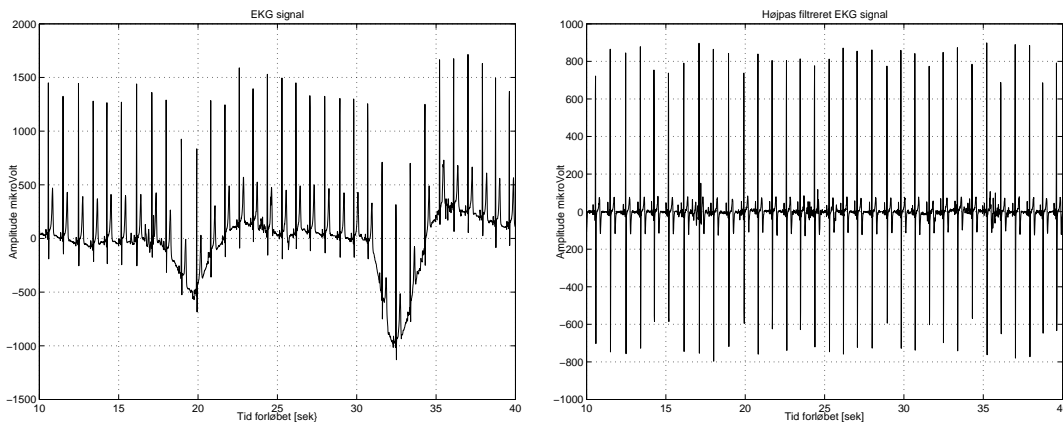
Filtersektionen ligger i umiddelbar forlængelse af forforstærkeren og der er ingen elektrisk isolering til patienten. Af denne grund skal filtersektionen forsynes med batterispændinger på  $\pm 9$

V. Nedenfor gennemgås overvejelserne, der gøres inden der opstilles en kravspecifikation for filteret.

### Højpasfilter - DC offset og knækfrekvens

Interferens fra muskelstøj kan give store problemer ved opsamling af EKG. Dette skyldes, at EMG-signaler overlapper frekvensbåndet for EKG og derfor ikke kan filtreres bort. I dette projekt vælges derfor at betragte muskelstøj i frekvensbåndet for EKG som en artefakt. Måling af EKG signalet vanskeliggøres desuden af offset og drift fra elektroder samt bevægelsesartefakter. Det kan resultere i baseline drift, hvilket er illustreret på figur 4.5

Baseline drift er lavfrekvent, og kan derfor reduceres ved at dæmpe lavfrekvente signaler. Signalet på figur 4.5, der er et udleveret EKG-signal samlet med 100 Hz, er blevet højpasfiltreret i MATLAB 6.0 for at undersøge ved hvilken knækfrekvens, baselinen stabiliseres. Denne er fundet til at være omkring 0,5Hz, hvilket stemmer overens med litteraturen omtalt i afsnit 2.2



(a) På figuren ses baseline drift fra EKG. Denne drift er hovedsageligt et resultat af drift fra elektrode-hud overfladen og bevægelsesartefakter

(b) Figuren viser EKG med stabil baseline som resultat af højpasfiltrering. Behandling af signalet er foretaget i MATLAB 6.0 og der er anvendt en knækfrekvens på 0,5Hz. Denne knækfrekvens giver en rimelig stabil baseline og anvendes i dette projekt

**Figur 4.5:** Figurene viser effekten af højpasfiltrering. Der er tale om det samme EKG-signal, samlet med 100 Hz, på begge figurer

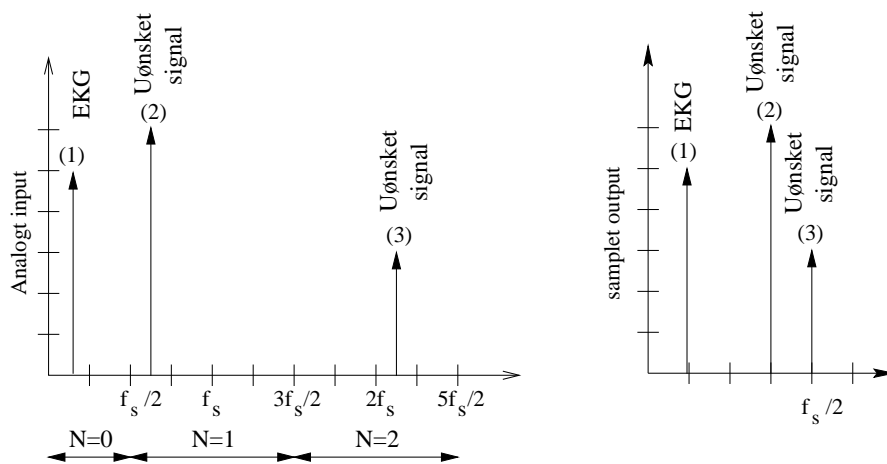
Det er muligt at eliminere drift på signalet digitalt. Dette kan gøres ved at anvende moving average algoritmen til behandling af signalet. For at muliggøre dette skal AD konverterens spændingsforsyning tillade at signalet kan drifte betydeligt, dvs. knækfrekvensen kan være 0,05 Hz som anbefalet. Men i dette projekt er der fastsat et headroom på kun 25 % og af denne grund er der ikke plads til en baselinedrift på ca. 100 % af signalets amplitude som det eksempelvis ses på figur 4.5. I stedet vælges at benytte et analogt højpasfilter, hvor knækfrekvensen sættes til 0,5 Hz.

### Lavpasfilter - antialiasering - AD konvertering

Det ønskes et filter, som sikrer et pasbånd på 0,5-150 Hz, der generelt anvendes for EKG til diagnostisk brug [3]. Lavpasfilteret skal således have en knækfrekvens omkring 150 Hz, og der skal i designet tages højde for aliasering. Nedenfor diskuteres de overvejelser, der gøres inden

kravspecifikationen for lavpasfilteret optilles.

Ifølge Nyquist's samplingsteori [9] digitaliseres analoge inputsignaler til AD konverteren under Nyquist frekvensen  $f_N = \frac{f_s}{2}$  til korrekte værdier, mens inputsignaler over  $f_N$  aliaseres tilbage i filterets pasbånd. Aliaseringen sker med bevaret styrke og gør det umuligt at skelne mellem signaler med frekvenser under  $f_N$  og signaler med frekvenser over  $f_N$ . Dette er illustreret ved figur 4.6



Figur 4.6: Figuren illustrerer, hvordan frekvenser højere en  $f_N$  aliaseres tilbage i pasbåndet efter sampling

Frekvensbåndet  $N = 0$  på fig. 4.6 spænder fra DC til  $\frac{f_s}{2}$ . Indenfor denne båndbredde digitaliseres analoge inputsignaler korrekt. For  $N > 0$  digitaliseres inputsignaler til båndbredden for  $N = 0$ . Denne aliasering af signaler beregnes som:

$$f_{alias} = |f_{in} - Nf_s| \tag{4.1}$$

Derfor skal frekvenskomponenter over Nyquist-frekvensen dæmpes, således at aliasering ikke forekommer.

AD konverterens opløsning og "Signal to Noise Ratio" (SNR) skal analyseres for at kunne lave en realistisk vurdering af hvilke krav dette stiller til filteret.

AD konverteren er en 12-bits konverter, der forsynes med 3,3 V. Opløsningen er derfor:

$$\frac{3,3V}{2^{12}} = 0.8mV \tag{4.2}$$

Hvis signaler dæmpes til et niveau mellem 0 - 0,8 mV kan der stadig forekomme aliasering. Dette skyldes AD konverterens quantization fejl. Et signal, som er dæmpet til 0,5 mV, er tættere på 0,8 mV end på 0 og fortolkes således som et signal med værdien 1 Least Significant Bit (LSB). For den perfekte 12-bitskonverter undgås aliasering ved at dæmpe til under  $\frac{1}{2}$  LSB.

Dette betyder, at der skal dæmpes:

$$|A_{min}| = 20 \cdot \log\left(\frac{3.3V}{2 \cdot 2^{12}}\right) \approx 78dB \tag{4.3}$$

Hvis de 78 dB ønsket dæmpet på en enkelt dekade, kan et 4. ordens filter benyttes, hvilket er let implementerbart. Med den ønskede knækfrekvens på 150 Hz vil alle signaler højere end 1500 Hz være frafilteret, jvf. dæmpningen på 78 db/decade, og samplingsfrekvensen bliver heraf minimum 3 kHz.

Der ønskes at minimere den mængde data, der skal behandles, og derfor ønskes en lavere samplingsfrekvens.

AHA-standarden [10] anbefaler sampling af EKG-signaler med 500 Hz for at gøre fremstillingen af det samlede EKG så nøjagtig som mulig, og således at parametre som Heart Rate Variability er mulige at beregne. Sampling ved denne frekvens forekommer datamæssigt mere rimelig, og det undersøges derfor, om denne samplingsfrekvens er mulig at implementere.

Dette gøres ved at se nærmere på SNR, idet det er unødvendigt at dæmpe til et niveau under støjniveauet.

De støjforhold, som designet af lavpasfilteret baseres på, er støjen, som kan forventes på indgangen til filteret, dvs. forforstærkers og højpasfilters interne støj sammen med den eksterne 50 Hz støj fra forsyningsnettet. Eftersom den eksterne støj er svær at estimere eller måle præcist for alle lokaliteter, valgte gruppen kun at måle og tage højde for den interne støj i forforstærker og højpasfilter. Denne målte med kortsluttet indgang på forforstærker, og var 3,5 mV peak-peak. Den krævede dæmpning, som bestemmes af SNR, beregnes til:

$$SNR = 20 \cdot \log\left(\frac{3,3V}{3,5mV}\right) \approx 59dB \quad (4.4)$$

AD konverteren er forringet til en opløsning på:

$$\frac{3,3V}{2^x} = 3,5mV \quad (4.5)$$

$$x \approx 9bit \quad (4.6)$$

## 4.4 Kommunikation mellem mc og pc

Data skal som nævnt overføres fra mc til pc gennem et kommunikationsdelsystem. Det vælges at gøre dette via den serielle kommunikationsstandard RS-232 [11]. Denne form for seriel kommunikation er valgt, fordi RS-232 er kendt og dokumenteret standard, og fordi pc'en understøtter RS-232.

Mc understøtter imidlertid kun seriel kommunikation via sin UART med 0-3,3V spændinger, og benytter således ikke de digitale spændingsniveauer foreskrevet af RS-232 på +5 - +15V for logisk 0 og -5 - -15V for logisk 1 ved transmitteren og en støjmargin på 2 V ved minimum spændingen på receiver og op til  $\pm 15V$ . Derfor er en enhed til konvertering mellem UART's signaler og RS-232-kompatible signaler nødvendig.

Da direkte forbindelse mellem patient (gennem mc) og lysnettet (gennem pc) risikeres, skal patienten isoleres mod fejlstrømme af en galvanisk barriere, bestående af en optokobler. Den galvaniske barriere placeres i forbindelse med den serielle forbindelse mellem mc og pc.

Begrundelsen for at placere den galvaniske barriere i serielforbindelsen er, at alle enheder på mc-siden er batteridrevne, og at man derfor kan opnå en reel isolering. Desuden betyder optokoblerens forvrængning af signalerne mindre i den digitale verden end i den analoge.

## 4.5 Mc

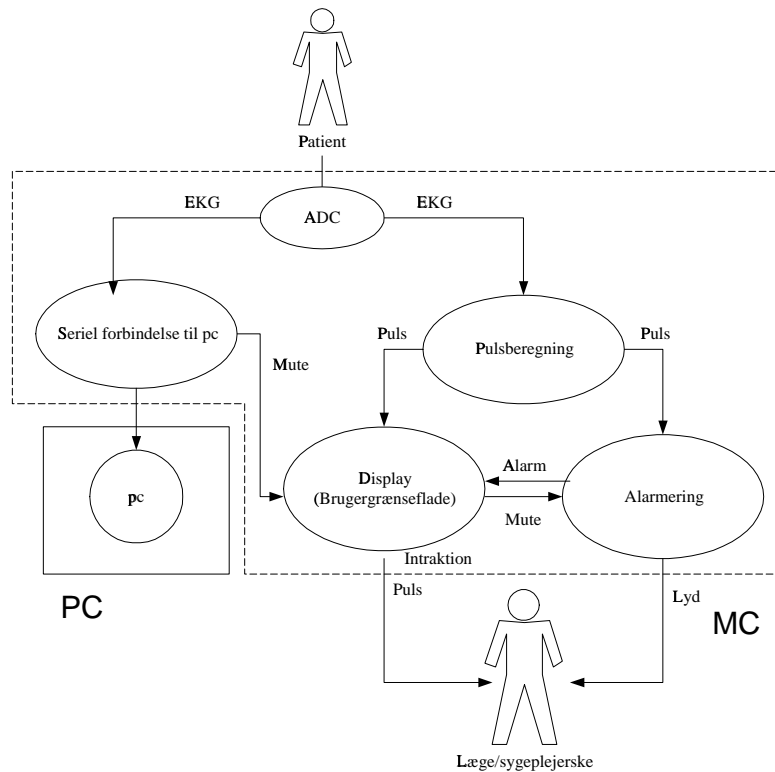
Produktspecifikationen i kapitel 3 beskriver et system, hvor mc-delsystemet i hovedtræk AD-konverterer, beregner pulsen, detekterer og alarmerer ved arytmier, styrer brugerinterface i form af indtastningsmuligheder og visning af puls/EKG samt varetager kommunikation med pc.

Udfra disse 5 hovedopgaver følger opdelingen i nødvendige processer naturligt:

- En proces, der varetager AD-konvertering og gør resultatet tilgængeligt for pulsregning og pc-kommunikation
- En proces, der beregner pulsen ud fra det konverterede signal

- En proces der undersøger, hvorvidt der skal alarmeres for arytmierne ventrikelflimmer, asystoli, tachy- og bradykardi
- En proces til brugerinterface, der viser puls/EKG og giver adgang til et menu-system, som brugeren kan manøvrere rundt i for at ændre indstillinger som pulsgrænser og lydindstilling
- En proces, der via UART kommunikerer med pc.

Figur 4.7 viser sammenhængen mellem processerne. Det er naturligvis nødvendigt at kontrollere data-flowet, så den rigtige funktionalitet opnås.



Figur 4.7: Sammenhængen mellem processer i delsystemet mc. Flow af informationer mellem processerne samt ydre grænseflader er angivet.

### 4.5.1 Struktur af eksekverbart program - main

Hovedprogrammets struktur er af afgørende betydning for designet af det enkelte modul. Det vælges at lade hovedprogrammet, main, kontrollere dataflow og kalde de enkelte moduler gennem almindelige funktionskald. Grænsefladen mellem modulerne udgøres dels af globale variable og variable, der transporteres ind og ud af main gennem funktionskald.

I main må initialiseringsprocedurer foregå før eksekvering af egentlige beregningsmoduler, der derefter skal foregå kontinuerligt, indtil systemets slukkes.

Af hensyn til mc'ens ydeevne er det nødvendigt, at kontrolstrukturene i main sørger for, at der ikke bruges unødigt processortid.

I det følgende designes processerne, så produktspecifikationen kan opfyldes, og således at det overvejes, hvilke variable der er input og output for modulerne.

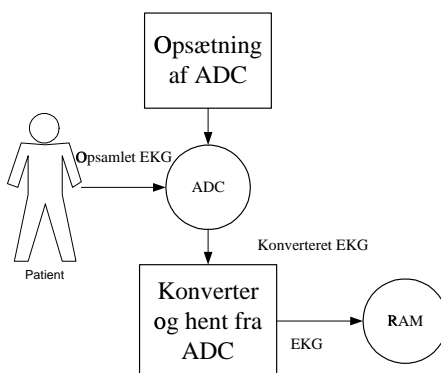
## 4.5.2 Proces: Konverter og hent fra ADC

Processen *Konverter og hent fra ADC* er nødvendig for at konvertere de analoge inputsignaler, der modtages på mc'ens ADC. Desuden skal disse data kopieres fra ADC'en til RAM-registre, så data er tilgængeligt for andre processer, og processen kan derfor naturligt bestå af to moduler som vist på fig. 4.8:

- Opsætning af ADC
- Lagring af konverterede værdier

*Opsætning af ADC* skal initialisere ADC'en, så AD-konvertering i følge 4.3.2 bliver mulig, og *Lagring af konverterede værdier* er nødvendig for at kopiere data fra ADC'en ud i RAM-registre. Dette er vist på fig. 4.8.

For at begrænse antallet af operationer skal data for EKG'et først kopieres til RAM, når der er konverteret på alle 3 kanaler, dvs efter samplingsserier à 3 samplinger.



**Figur 4.8:** Figuren viser sammenhængen mellem modulerne i *Konverter og hent fra ADC*.

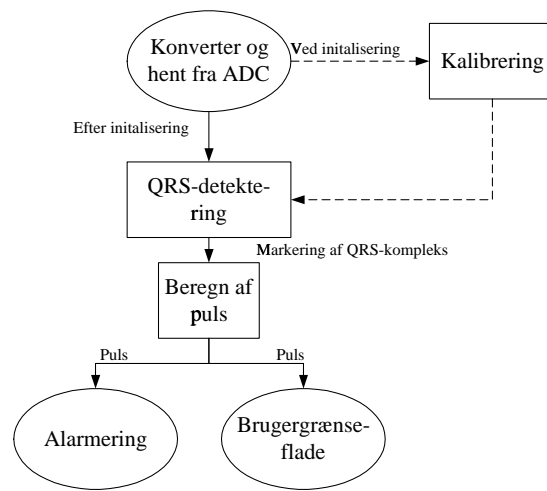
## 4.5.3 Proces: Pulsberegning

Processen *Pulsberegning* har til opgave at beregne pulsen ud fra det konverterede EKG-signal. EKG-signalet skal derfor hentes fra RAM, og pulsslæg-i form af QRS-komplekser skal detekteres ud fra signalet. Ud fra antallet af detekterede QRS-komplekser skal pulsen beregnes. Pulsen bruges i processerne *Brugergrænseflade* og *Alarmering*. Processen *Beregn puls* kan derfor logisk opdeles i 2 moduler: *QRS-detektering* og *Beregn puls* som vist på figur 4.9. Modulen *Beregn puls* foretager beregningen af pulsen ud fra hyppigheden af QRS-komplekser. Derfor skal *QRS-detektering* sætte et flag i det øjeblik, et nyt QRS-kompleks er detekteret og derved danne grundlag for pulsberegningen. QRS-detekteringen skal i følge produktspecifikationen i kap. 3 fungere på alle patienter uanset udseende af EKG. Det er derfor nødvendigt at tilføje modulet, *Kalibrering*. Kalibreringsmodulet skal ved starten af hver brugssituation bestemme størrelsen af de parametre, det vælges at detektere QRS-komplekser ud fra, inden den egentlige QRS-detektering foregår. Størrelsen af parametrene skal gøres tilgængelig for QRS-detekteringsmodulet.

## 4.5.4 Proces: Alarmering

Processen *Alarmering* skal bringe en alarmkrævende situation til personalets kendskab, når de brugerdefinerede grænser for puls overskrides. Når der er detekteret en alarm, skal en lydindikation startes - med forskellige lyde ved henholdsvis livstruende alarm (asystoli og ventrikelflimmer) og pulsalarmer (bradykardi og tachykardi). Processens funktionaliteter kan derfor naturligt opdeles i modulerne:

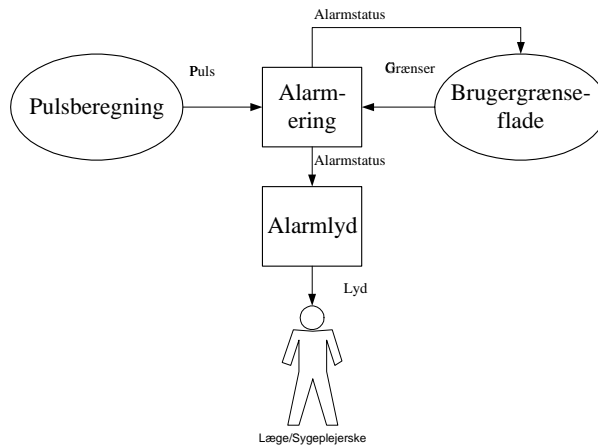




Figur 4.9: Figuren viser sammenhængen mellem modulerne i pulsbergningsprocessen

- Alarmedetktering
- Alarmlyd

Sammenhængen mellem modulerne er vist på fig. 4.10.



Figur 4.10: Figuren viser sammenhængen mellem modulerne i processen Alarmering

Alarmedetktering skal ud fra grænserne samt pulsen undersøge, om pulsen ligger i et område, der kræver alarmering. Alarmlyd skal kontrollere, om alarmedetkterings-modulet har detekteret en alarm. Så længe dette er tilfældet, skal en passende alarmlyd lyde; bip vælges ved pulsalarmer og kontinuerlig lyd ved livstruende alarm, da denne graduering af intensitet vurderes at være i overensstemmelse med alarmernes alvor.

#### 4.5.5 Proces: Brugergrænseflade

For at tillade brugeren at kommunikere med mc-delsystemet som beskrevet i kap. 3 er det nødvendigt at visualisere de opsamlede målinger, indstillinger samt indstillingsmulighederne. Brugergrænsefladen må derfor fysisk bestå af et display, der med tekst og grafik viser puls, grænser og Einthovens afledning II, samt et knappanel, der kan benyttes til navigering i de forskellige menuer og til indstilling af grænser for tilladelig puls. Desuden er en knap nødvendig til midlertidigt eller permanent sluk for lydindikation af alarm, dvs. “mute”.

Brugergrænseflade skal derfor, foruden den grafiske afbildning af Einthovens II afledning, kunne vise 3 forskellige menuer:

- Hovedmenu med pulsangivelse og mutestatus
- Grænsemenu (med indstillingsmulighed)
- Alarmskærm

I menuen *Hovedmenu med pulsangivelse* skal vises pulsen, mutestatus samt genveje til de to andre menuer. I grænsemenuen *Pulsgrænser* skal vises de nuværende grænseværdier for hhv. tachy- og bradykardi, og grænserne skal kunne ændres. Alarmskærmen *Alarmstatus* skal springe frem i de tilfælde, hvor en alarm er detekteret, og der skal stå hvilken arytmi, der alarmeres for. Ved tryk på en knap skal der kunne mute's eller vendes tilbage til hovedmenuen, således at grænseværdierne fx kan ændres i menuen *Pulsgrænser*. Grænsen for bradykardi må ikke være højere end tachykardi-grænsen eller omvendt. Alle menuer skal være tilgængelige fra hovedmenuen, således at det er simpelt at manøvrere rundt mellem forskellige menuer.

Input til *Brugergrænseflade* er pulsen, alarm-status, konverterede EKG-data til grafisk visning samt tryk på knappanelet.

Output skal være de forskellige menuer, alarmskærme, pulsgrænserne samt mute af højttaler.

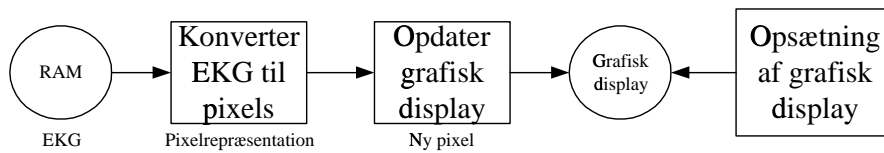
For at opnå ovenstående funktionalitet i *Brugergrænseflade* er det naturligt at inddele processen i følgende 7 moduler, hvis sammenhæng er vist på fig. 4.11 og 4.12. Det er nødvendigt med forskellige moduler til styring af tekstdelen og grafikdelen, da det ikke er muligt i universitetets laboratorium at få ét display til både grafik og tekst:

- Opsætning af tekstdisplay
- Opdater tekstdisplay
- Knapopsætning
- Opsætning af grafisk display
- Konverter EKG til pixels
- Opdater grafisk display
- Mute

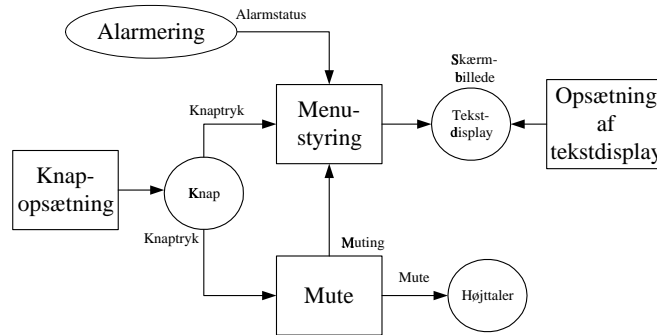
*Opsætning af tekstdisplay* og *Opsætning af grafisk display* skal initialisere hhv. tekstdisplay og grafisk display. For tekst-displayet kræves det, at der opsættes til brug af 4 linier. *Knapopsætning* skal sørge for korrekt påvirkning af delsystemet mc ved tryk på knapperne, således at knaptryk kalder de korrekte menuer frem. For at minimere antallet af knapper er det hensigtsmæssigt, hvis den samme knap kan forårsage forskellige handlinger afhængigt af, hvilken menu der aktuelt vises.

Tryk på muteknappen skal via modulet *Mute* slukke højttaleren og gennem *Opdater tekstdisplay* angive, at der er mutet. Selv om der trykkes mute flere gange, inden minuttet er gået, i tilfælde af mute, må der kun være mute i 1 minut; herefter skal alarmeren gå igang igen. Desuden skal alarmlyden tændes, selvom der er mutet, hvis der opstår asystoli. I tilfælde af permanent mute skal denne funktionalitet aktivt slås fra, før lyden må starte. For at detekteringen af bradykardi og tachykardi kan foregå efter patienttilpassede grænser skal disse grænser være tilgængelige for modulet *Alarmering*. Grænsernes udgangsværdi (hvis ikke brugeren ændrer dem) skal være 40 for bradykardi og 100 for tachykardi [4].

For at *Opdater grafisk display* kan sørge for visning af det EKG, der hentes fra RAM-registrene, skal *Konverter EKG til pixels* konvertere amplituden på det konverterede signal til adresser på det grafiske display.



Figur 4.11: Sammenhængen mellem grafikrelaterede moduler i processen Brugergrænseflade



Figur 4.12: Sammenhængen mellem tekst- og knaprelaterede moduler i processen Brugergrænseflade

#### 4.5.6 Kommuniker med pc

Processen *Kommunikér med pc* skal varetage overførsel af data mellem mc og pc, således at EKG-data overføres fra mc, og at det fra pc er muligt at “mute” alarmer permanent. Kommunikationen skal som nævnt i afsnit 4.4 foregå via den serielle kommunikationsstandard RS-232, hvorfor mc’ens UART benyttes.

*Kommunikér med pc* skal som input have data fra RAM-registrene, så de datapakker, der sendes til UART’en som krævet i kap. 3 indeholder data fra de 3 opsamlede kanaler. Der er behov for følgende 3 moduler:

- Opsætning af UART
- Seriel kommunikation
- Permanent mute

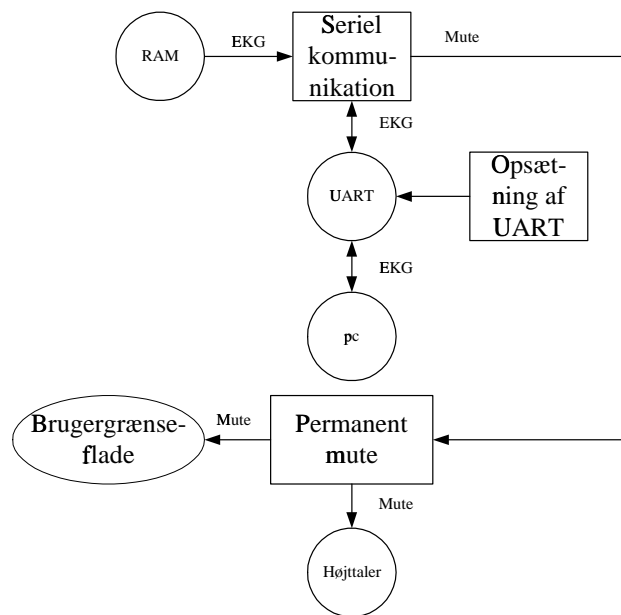
*Opsætning af UART* skal som vist på fig. 4.13 initialisere UART’en, så seriel overførsel af data mellem mc og pc bliver mulig, og *seriel kommunikation* skal sørge for den løbende overførsel af data. *Permanent mute* skal sørge for, at kommunikation kan foregå fra pc til mc, og at mute bliver permanent.

Det er nødvendigt, at UART’en opsættes til at overføre alle 3 kanaler samlet med 500 Hz, så der ikke mistes data. Af samme grund skal der sikres imod fejl i transmissionen, og data fra de 3 afledninger skal sendes i samme rækkefølge hver gang. Hver pakkes start skal markeres, således at det vides, hvilken afledning, der modtages hvornår.

#### 4.6 Pc

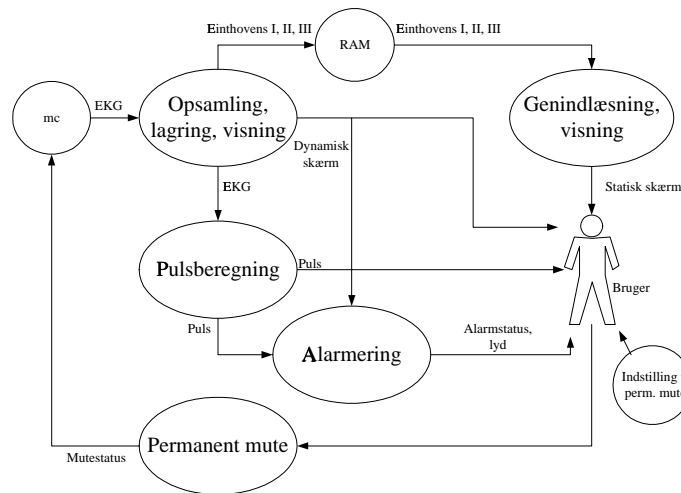
Pc-delsystemet skal varetage alle funktionaliteter forbundet med centralskærmen nævnt i kap. 3: opsamling, lagring, visning af aktuelt og lagret EKG, pulsberregning, alarmeringsfunktionalitet med efterfølgende markering på lagret EKG samt indstilling af permanent mute på mc. Dette giver en naturlig inddeling i 6 processer, der hver har tilknyttet sin egen del af brugergrænsefladen:

- Opstart ny patient



**Figur 4.13:** Figuren viser sammenhæng mellem modulerne i processen *Kommunikation med pc*

- Opsamling, lagring, visning
- Genindlæsning, visning
- Pulsberegning
- Alarmering og markering af alarmer ved visning
- Permanent mute



**Figur 4.14:** Sammenhængen mellem processerne i pc-delsystemet. Processen *Opstart ny patient* er ikke medtaget, da denne proces ikke er afgørende for funktionalitet under brug

Processen *Opstart ny patient* er nødvendig for at opfylde produktspecifikationens krav om mulighed for indtastning af patientdata. Processen *Opsamling, lagring, visning* er som vist på fig. 4.14 nødvendig for at varetage kommunikationen med mc og lagre og vise de 3 modtagne EKG-afledninger fra mc på den dynamiske skærm, og ud fra disse 3 afledninger beregne Goldbergs 3 afledninger. For at begrænse den mængde data, der skal lagres, vælges det kun at gemme Einthovens 3 afledninger, således at de beregnes igen, når de genindlæses. *Genindlæs, visning* skal

hente og vise det gemte EKG på den statiske skærm og tillige foretage beregningen af Goldbergs 3 afledninger. For *Genindlæs, visning* er det krævet, at der kan zoomes på det viste EKG. Pulsens skal beregnes af processen *Pulsberegning* ud fra de opsamlede data., og den skal vises for brugeren. Ud fra pulsen skal *Alarmering* detektere alarmkrævende situationer: asystoli, ventrikelflimmer, tachykardi og bradykardi. Alarmstatus for disse 4 alarmer skal vises for brugeren. I tilfælde af, at brugeren vælger permanent mute, skal *Permanent mute* bringe denne mutestatus til mc's kendskab.

## Kapitel 5

# Afgrænsning af system

Med inddelingen af mc-systemet i processer og moduler og pc-systemet i processer haves et overblik over omfanget af programmeringsopgaverne, der skal løses. I projektperioden er der begrænset tid til rådighed, så det vælges at begrænse antallet af moduler og processer. Afgrænsningen foretages, så der stadig haves et funktionsdygtigt system, men en række funktionaliteter er af tids- og ressourcemæssige grunde ikke blevet implementeret.

Der er for mc-systemet tale om, at modulerne relateret til det grafiske display ikke videre bearbejdes i rapporten. Ud over det grafiske display er modulet *Permanent mute* heller ikke designet og implementeret, og *Brugergrænseflade*-processen behandles således i resten af rapporten som bestående af modulerne *Opsætning af tekstdisplay*, *Knapopsætning*, *Menustyring* og *Mute* med 1 minuts mute.

Da ventrikelflimmer kan have vidt forskellige udseender på EKG'et, er det meget vanskeligt at detektere denne arythmi med sikkerhed [12]. Derfor vælges det at kun asystoli, bradykardi og tachykardi skal detekteres.

På pc er der kun arbejdet med de to mest centrale processer, nemlig *Opsamling*, *lagring*, *visning* og *Genindlæsning*, *visning*, da disse processer vurderes at være de mest centrale for både systemet og projektgruppens læreproces, og det af tidsmæssige årsager har været nødvendigt at afgrænse projektet.

Processerne *Pulsberegning* og *Alarmering og markering af alarmer ved visning* er nært relateret til de tilsvarende processer i mc-systemet, så det blev vurderet, at der ikke skulle lægges flere ressourcer i disse processer.

Med hensyn til processen *Opstart ny patient* er det vurderet, at der ikke er noget læringsmæssigt at hente ved design og implementation, så af hensyn til tidspres er processen ikke videre bearbejdet.

I processen *Opsamling*, *lagring*, *visning* vurderes det at være tilstrækkeligt at gemme data for de seneste 5 minutter; det vigtige er at vise princippet for lagring og ikke hvor meget data, der lagres.

Der er desuden fra projektgruppens side valgt at vægte implementation af et fungerende system højere end en grundig accepttest, og derfor udelades denne. I stedet vil en kort funktionalitetstest i kapitel 15 fremhæve systemets vigtigste egenskaber.



# DEL III KRAVSPECIFIKATION

*I kravspecifikationen vil krav udledt i systemdesignet i analysen blive opstillet på punktform.  
Kravspecifikationens formål er at give systemudvikleren et fast defineret grundlag, hvorudfra det egentlige systemdesign kan begyndes.  
Kravspecifikationen skal afspejle så meget af analysen som muligt, og derved sikre at de væsentligste punkter i produktspecifikationen bliver fulgt*





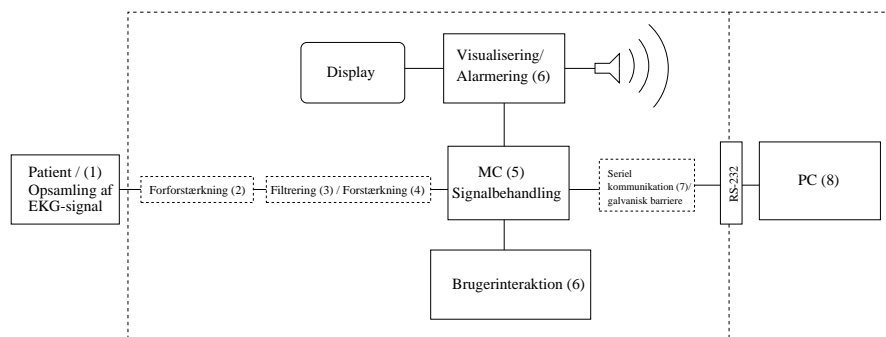
# Kapitel 6

## Kravspecifikation

### 6.1 Indledning

Kravspecifikationen skal sikre, at produktspecifikationen opfyldes, og dermed at den opstillede problemformulering i *Foranalysens* kapitel ?? opfyldes:

*Der ønskes udviklet et to-delt EKG-monitoringsystem til centralovervågning og til bedside monitoring. Systemet skal kunne præsentere EKG og puls for lægen/sygeplejersken.*



**Figur 6.1:** Flowdiagram for systemet. Nummereringen angiver rækkefølgen af delsystemerne og bruges som referenc i kravene

### 6.2 De specifikke krav

#### 6.2.1 Definitioner

Alle krav er primære krav, der skal overholdes i designet af EKG-monitoren.

#### 6.2.2 Krav til delsystemet *instrumentering*

##### Opsamling og Forforstærkning (1)/(2)

- Der skal opsamles i alt 3 kanaler.
- Det skal benyttes Ag-AgCl elektroder, til opsamling af patientens EKG-signal.
- Det opsamlede EKG signal må maksimalt svækkes 1 % 99% inden forforstærker.
- Beskyttelsesdioder og modstande skal anvendes for at sikre patienten mod afladningsstrømme over 0,5 mA.

- Der skal laves en forforstærker for hver kanal - 3 i alt.
- Forforstærkeren skal have en høj CMRR til frasortering af common mode signaler på indgangen.
- EKG signalet skal forstærkes 20 gange.

### Filtrering (3)/(4)

- Der skal laves et filter for hver kanal - 3 i alt.
- Filtrering af EKG-signalet skal bestå af en højpasfiltrering og en lavpasfiltrering.
- Frekvenser under 0,5 Hz skal dæmpes med 20 dB/decade for at undgå baseline drift.
- Frekvenser over 150 Hz skal dæmpes med minimum 59 dB. Lavpasfilteret skal designes således at transitionsbåndforholdet minimerer den nødvendige samplingsfrekvens til 500 Hz.
- Forstærkertrinnet skal forstærke EKG-signalet 50 gange.
- Der skal sikres imod negative spændinger til ADC'en.
- Der skal sikres imod spændinger over 3,3 V til ADC'en.

### Forsyning

- Forforstærker, højpasfilter og lavpasfilter skal forsynes med batterispændinger på højest  $\pm 9$  V.

### 6.2.3 Krav til delsystemet mc (5)/(6)

Ud fra de diskuterede og analyserede krav til delsystemet mc i kap. 4, opstilles i det følgende krav til de moduler, der skal designes og implementeres på mc'en.

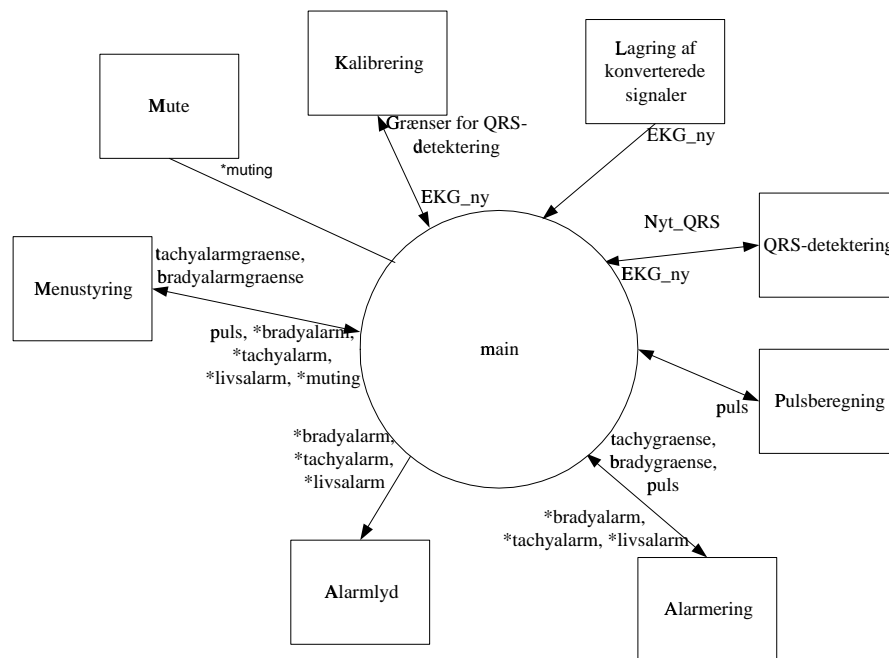
Først kravespecificeres for main-programmet, og her fastlægges variabelnavnene på de variable, der skal transporteres mellem moduler gennem main, således at modulerne kan kodes uden kendskab til øvrige moduler. Alle moduler skal benytte de variabelnavne, der er angivet på fig. 6.2. Det er i afsnit 4.5 fastlagt, hvilke variable, der er brug for.

### Main

- Alle opsætningsmoduler skal kaldes før egentlige beregningsmoduler
- Beregningsmoduler skal kaldes i en rækkefølge og med betingelser for flowkontrol, der sikrer at moduler ikke kaldes unødvendigt
- Variable skal ind og ud af modulerne gennem main efter skabelonen på fig. 6.2.

### Modul: Opsætning af ADC

- Der skal samples på 3 kanaler
- Hver kanal skal samples med 500 Hz
- Bitopløsning skal være mindst 9 bit/sample pr. kanal



**Figur 6.2:** Flowet af variable ind og ud af main. Variable er anført tættest på det modul eller main, de kommer fra, dvs. transporteres ud af; QRS-detekteringsmodulet bruger fx variabelen EKG ny og sender variabelen nyt QRS tilbage til main. Hvor der skal returneres flere variable fra et modul, er det med \* foran variabelnavnene angivet, at der skal anvendes pointere

### Modul: Lagring af konverterede signaler

- Alle konverterede kanaler skal flyttes til RAM
- Der skal flyttes til RAM efter hver samplingsserie
- 2. afledning skal være tilgængelig for main som en global variabel

### Modul: Kalibrering

- I begyndelsen af hver brugssituation skal størrelsen af de parametre, QRS-komplekser skal detekteres ud fra, automatisk fastlægges ud fra det samlede EKG
- Størrelsen skal returneres til main

### Modul: QRS-detektering

- QRS-komplekser skal detekteres, hvis det samlede EKG er i overensstemmelse med de parametre, *Kalibrering* har fundet størrelsen af
- Hvis der er detekteret et QRS-kompleks, skal der returneres en markering til main

### Modul: Beregn puls

- Pulsen skal beregnes efter hvert detekteret QRS-kompleks
- Pulsen skal beregnes så der tages hensyn til både opdateringshastighed og præcision

**Modul: Alarmering**

- Asystoli skal detekteres
- Bradykardi skal detekteres, hvis pulsen er under grænseværdien for bradykardialarm
- Tachykardi skal detekteres, hvis pulsen er over grænseværdien for tachykardialarm
- Alarmer skal nulstilles, når alarmkriteriet efter en alarmkrævende situation ikke er opfyldt længere
- Alarmstatus skal returneres til main

**Modul: Alarmlyde**

- Ved asystoli skal alarmeringen være en gennemtrængende hyletone
- Ved pulsalarm skal alarmeringen bestå af tydelige bip
- Lyden skal slukkes, når alarmstatus angiver, at der ikke er alarm længere

**Modul: Mute**

- Mute skal vare 1 minut
- Trykkes der på muteknappen, imens mute er aktiveret, skal mute-perioden ikke forlænges.
- Mutestatus skal returneres til main, så det kan vises på displayet
- Hvis der opstår asystoli, skal alarmeringen begynde selvom mute er aktiveret
- Der skal kunne mute's på forhånd

**Modul: Opsætning af tekstdisplay**

- Der skal kunne udskrives skærbilleder på 4 linier

**Modul: Opdatering af tekstdisplay**

- Der skal kunne vises skærbillederne: hovedmenu med pulsangivelse, grænsemenu og alarmskærme
- Alle skærbilleder skal kunne nås fra hovedmenuen
- Grænsen for bradykardi må ikke kunne indstilles til at være højere end grænsen for tachykardi og omvendt
- Bradykardigrænsen default: 40
- Tachykardigrænsen default: 100
- Grænserne skal returneres til main
- Skærbillederne for alarmer skal være inaktive i grænsemenuen, så grænseindstilling uden afbrydelse er mulig

**Modul: Knapopsætning**

- Knaptryk skal kunne forårsage: Nyt skærbillede, hævning eller sænkning af alarmgrænser eller muting
- Hver knap kan have forskellig betydning alt efter det aktuelle skærbillede

**Modul: Opsætning af UART**

- UART'en skal opsættes til asynkron kommunikation
- Der skal opsættes til at 3 kanaler, hver samlet med 500 Hz, kan transmitteres
- Der skal sikres imod fejl i transmissionen

**Modul: Seriel kommunikation**

- Hver transmitteret pakke skal indeholde afledningerne I, II og III i den nævnte rækkefølge
- Begyndelsen af hver pakke skal klart markeres

**6.2.4 Seriel kommunikation (7)****Konvertering af spændinger**

- Spændingsniveauer skal konverteres som i tabel 6.1:

Signaloprindelse og størrelse	Konverteres til
mc, 0-0,6 V	pc, 5-15 V
mc, 2,7-3,3 V	pc, -15- -5 V
pc, 5-15 V	mc, 0-0,6 V
pc, -15- -5V	mc, 2,7-3,3 V

*Tabel 6.1: Konvertering mellem spændingsniveauer i kommunikationsenhed*

- Forbindelsen skal indeholde 3 ledninger: transmit, receive og signal ground
- Transmitteren på afsender forbindes til receiveren på modtager og omvendt
- Der skal forsynes med batterispændinger på 0-9 V
- Der skal strømisoleres, således at patientsiden er isoleret fra pc-siden
- Der skal anvendes DB-9-stik i begge ender

**6.2.5 Krav til pc-delsystemet (8)****Proces: Opsamling, lagring, visning**

- Skal kommunikere med mc gennem den serielle RS-232 standard
- Skal lagre Einthovens 3 afledninger for de seneste 5 minutter
- Skal vise Einthovens 3 afledninger samt beregne og vise Goldbergs 3 afledninger

**Proces: Genindlæsning, visning**

- Skal genindlæse Einthovens 3 afledninger fra en fil og vise dem
- Skal beregne Goldbergs 3 afledninger fra Einthovens og vise dem
- Skal muliggøre zoom på EKG'erne

# DEL IV

## DESIGN, IMPLEMENTATION OG TEST

*I denne del af rapporten vil det blive beskrevet, hvorledes processer og moduler er designet og implementeret i det samlede system.*

*Beskrivelser af moduler findes under de respektive processer, hvor dokumentationsstandarder vil blive beskrevet i Introduktionen til design og implementation, kapitel 7.*

*Efterfølgende vil alle processer og moduler blive samlet, og det beskrives hvorledes processer og moduler implementeres som et samlet program på mc'en.*

*Herefter følger design af systemets pc-del.*





# Kapitel 7

## Introduktion til design og implementation

*I dette kapitel beskrives strukturen i dokumentationen af systemudviklingen.*

### 7.1 Designmetoder

Der er designet i følge SPU [13]. Under arbejdet med design, implementation, debugging og test er udviklingsværktøjerne fra IAR Systems til mc'en anvendt: programmer er skrevet i Workbench, og der er med C-Spy debugget vha. breakpoints og watch på variable både ved simulering og programeksekvering på mc.

Med breakpoints menes, at programmet lægges på mc'en, og at det med C-Spy er markeret i koden, hvor eksekveringen af programmet skal stoppe - dvs. hvor der sættes et breakpoint. Watch på variable betyder både ved simulering og eksekvering af program på mc'en, at de variable, der specificeres, overvåges mht. værdi; ved simulering opdateres de hele tiden, på mc angives værdierne først, når programmet stoppes.

#### 7.1.1 Struktur i dokumentationen

De to delsystemer, der involverer softwareudvikling, er dokumenteret i de følgende to kapitler. Dokumentation af de øvrige delsystemer - instrumentering og seriel kommunikation - forefindes i appendiks D og E, da hardwareudvikling ikke er centralt for projektet på 4. semester.

Til dokumentation og test af moduler på mc og af processer på pc, anvendes V-modellen fra SPU. Denne model implementeres i dokumentationen af mc- programmelle, ved at det for hver proces først overvejes, hvordan kravspecifikationen for modulerne kan overholdes. Her lægges der en "slagplan" for moduldesignet, så der kan arbejdes videre ad samme vej for alle moduler.

Hvert modul designes, og implementationen beskrives ud fra flow-diagrammer. Endelig dokumenteres test af modulet, hvorudfra det kan konkluderes, om kravspecifikationen er opfyldt.

Mc-programmet udarbejdes i C-kode, da udviklingssoftwaren til mc'en understøtter dette. Til pc'en vælges det at anvende programmeringssproget LabVIEW.

Som eksempel på implementation af kode, inddrages kildekoden for modulet *Alarmlyd* i dokumentationen.

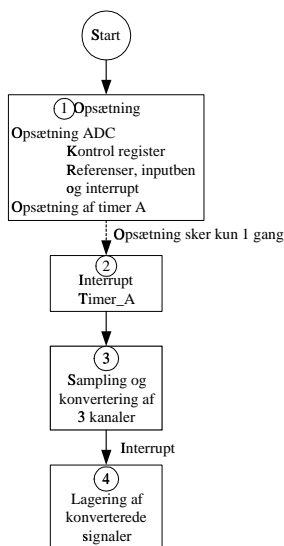


# Kapitel 8

## Konverter og hent fra ADC

Dette kapitel omhandler design, implementation og test af modulerne "Opsætning af ADC" og "Lagring af konverterede signaler", hvilket udgør processen "Konverter og hent fra ADC". Modulerne designes og implementeres på grundlag af de stillede krav. Testen af modulerne er lavet samlet og beskrives sidst i kapitlet.

Opbygningen af ADC'en er en multiplexer, der vælger hvilken kanal, der skal samples på, hvorefter der sidder en 12 bit A/D konverter, som gemmer det konverterede signal i et internt hukommelsesregister. Sammenhængen mellem de 2 moduler ligger i funktionaliteten af ADC'en, hvilket er illustreret på figur 8.1, og vil blive forklaret og refereret til med udgangspunkt i denne figur.



Figur 8.1: Opbygningen ADC modulet

Først opsættes ADC'en, se afsnit 8.1.2, til den ønskede funktionalitet (1), den opsætning sker kun en gang, herefter styres ADC'en af Timer\_A, der i et interrupt for Timer\_A (2), starter ADC'en, så den samler, konverterer og lagrer data fra de ønskede kanaler. Efter lagringen gives et interrupt, der starter lagringen af de konverterede signaler i RAM'en (4).

### 8.1 Modul: Opsætning af ADC

Kravene til *Opsætning af ADC* er:

- Der skal samples på 3 kanaler

- Hver kanal skal samples med 500 Hz.
- Bitopløsning skal være mindst 9 bit/sample pr. kanal

### 8.1.1 Design af Opsætning af ADC modul

Valg af opsætningen af ADC'en udfra ønskede krav, giver ikke de store muligheder. Samplingen med 500 Hz kan vælges at være interruptstyret af en timer eller styret af timer uden interrupt. Valget er faldet på timer, der starter et interrupt til ADC'en.

### 8.1.2 Implementation af Opsætning af ADC modul

Modulet findes implementeret i filen "setup.c".

**Kontrolregister :** I kontrolregistrene er ADC'en sat op til at konvertere en sekvens af kanaler, således at der samples på 3 kanaler så hurtig, at kun begrænsningen af den tid det tager at holde signalet på indgangen, samt den tid det tager at konvertere signalet, har betydning. Formålet med dette er at gøre signalet stabilt og at kondensatoren i A/D konverteren kan blive opladet.

Beregningen af den tid det tager for opladningen af kondensatoren, kan findes ud fra, at signalet konverteres med over 1/2 LSB nøjagtighed, og er givet ved [14]:

$$t_{ch}(1/2LSB) = (R_s + 1K\Omega) \cdot C_i \cdot 9,011 \quad (8.1)$$

Hvor  $R_s$  er modstanden, på indgangen til ADC'en, som er ca. 300  $\Omega$ , se appendiks (D.2) og  $C_i$  er A/D konverterens kondensator på 30 pF. Ud fra formel 8.1 bliver tiden det tager at oplade kondensatoren  $t_{ch}(1/2LSB) = 0,35 \mu s$ . Derfor skal sample-hold tiden for mc'en være større, hvilket er givet ved formelen:

$$t_{sample} = 4 \cdot t_{ADC12CLK} \cdot n \quad (8.2)$$

$n$  vælges til 24, udfra de mulige valg i manualen [14]. Klokken til styringen af hastigheden for konverteringen er valgt til en softwareklokke på ca. 5 MHz (ADC12OSC), og heraf fås en sample-hold tid på:  $t_{sample} = 19,2 \mu s$ , hvilket betyder, at kondensatorerne har nok tid til at blive opladet.[14]

**Referencer, inputben og interrupt:** Som referencespænding for ADC'en anvendes den interne forsyningsspænding på 0 - 3.3 V. Som inputben på ADC vælges ben A0 til afledning 1, A1 til afledning 2, samt A2 til afledning 3, og det er i den rækkefølge afledningerne skal konverteres. A2 sættes op til at være den sidste konvertering i en sekvens.

Efter konvertering skal værdierne gemmes over i de tre første interne hukommelsesregistre på A/D konverteren, hvor der sættes et interruptflag når værdien af den sidste konvertering er gemt.

**Timer A:** Timer A er konfigureret til at styre frekvensen på 500 Hz for samplingen og A/D konverteringen. Derfor er Timer A valgt til at være klokke styres af submaster klokken, der kører med en frekvens på 7,3728 MHz. Timer A skal tælle op til et *delay* ( $7372800/500$ ) svarende til 0.002s ( $500^{-1}$ ), og derefter give et interrupt, som starter samplingen og A/D-konverteringen. Valget af submaster klokken er foretaget for at komme tættest på de 500 Hz, da Timeren tæller op til heltallet for delayet, som giver en frekvens på 500,02 Hz ( $7372800/500 = 14745,6 \rightarrow 14745 = 500,02$  Hz). Timer A er sat til at tælle kontinuerlig, forklaring følge under Interrupt Timer A.

**Interrupt Timer A:** I interrupt service rutinen for Timer A startes en A/D konvertering, samt der lægges den samme værdi for delayet til delayet, således at Timer A forsættermed at tælle op til næste interrupt og starter forfra, når den er kommet til  $2^{16}$ , da timer A er en 16 bit tæller. Det gøres i stedet for at nulstille Timeren hver gang.

**Sampling og konvertering af de 3 kanaler:** A/D konverteringen sker på interrupt fra Timer A, og videresender et interrupt til modulet *Lagring af konverteret signal*, efter ADC'en har lagt data i det tredje interne register, hvilket betyder at ADC'en er færdig med sin konverterings sekvens.

## 8.2 Modulet Lagring af konverterede signaler

Kravene til dette modul er:

- Alle konverterede kanaler skal flyttes til RAM
- Der skal flyttes til RAM efter hver samplingsserie
- 2. afledning skal være tilgængelig for main som en global variabel

### 8.2.1 Design af Lagring af konverterede signaler

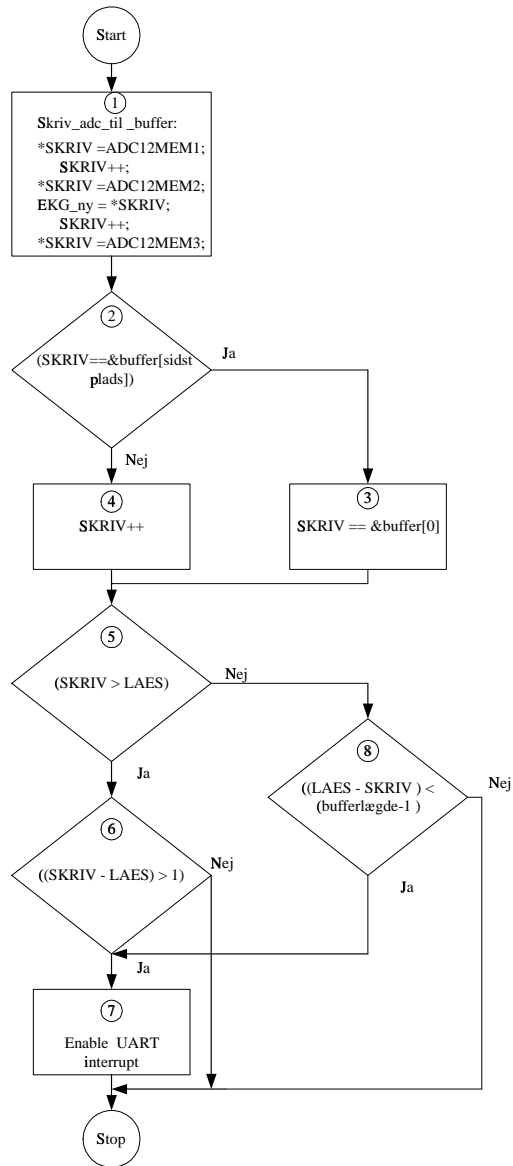
Funktionaliteten af dette modul skal være af flytte data fra ADC til RAM, desuden vil det være praktisk at lagre data, så de er lettilgængelige for de moduler, der skal bruge dem. Da UART'en skal udlæse alle data og sende dem videre, designes lagring mest optimal for UART'en. Derfor bruges en ringbuffer som lagringsprincip, hvor kanaler ligger ..1,2,3,1,2.. af de samlede og konverterede EKG-signal, da dette letter tilgangen for UART'en, og derved optimeres dataflowet. I designet er det ikke praktisk at skulle udlæse den anden afledning fra ringbufferen til puls-beregningsprocessen sammen med UART'en, derfor laves en separat variabel, som den anden afledte gemmes i hvergang. I ringbufferen bruges en pointer (SKRIV) til skrive til bufferen og en pointer (LAES) til at læse udfra bufferen. Desuden testes det på om, der ligger data i buffer, som ikke er udlæst til UART'en, hviket da skal enable interrupt for UART'en.

### 8.2.2 Implementation af Lagring af konverterede signaler

Modulet er implementeret i filen "ADC og seriel transmission.c" som interruptservicerutinerne for ADC\_VECTOR og TIMERA0\_VECTOR.

Implementation af modulet er lavet som en interrupt service rutine og er illustreret i figur 8.2, hvor der bliver refereret udfra nummereringen af de forskellige elementer. Værdierne fra ADC'ens interne hukommelses register, lægges element efter element over i bufferen, som vist i (1). Desuden lægges værdien ADC12MEM1(Afledning 2) over i variabelen (EKG\_ny), som er den værdi pulsen skal beregnes udfra. Til slut tælles pointeren umiddelbart ikke op, da der første testes om bufferen er nået til det sidste element og skal begynde forfra. Grunden til kun at teste her om bufferen er nået til sidste element er, at længden af buffer er lavet som et multiplum af 3, og vil derfor altid nå til det sidste element ved denne betingelse.

(2) Hvis SKRIV pointeren er nået til det sidste element, (3), skal bufferen begynde forfra, ellers skal pointer pege på det næste element i bufferen (4). Herefter testes om SKRIV er foran LAES(5), og om der er mere end et element i bufferen, der er klar til transmission (6) - i så fald enables interrupt for UART-transmit. (7). Hvis SKRIV fra (5) ikke var større end LAES, skal UART-interrupt enabls med mindre SKRIV peger på det første element i bufferen og LAES peger på det sidste (8).



Figur 8.2: Opbygningen af Lagring af konverterede signaler modulet

### 8.2.3 Test af Konverter og hent fra ADC

Test af modulerne *Opsætning af ADC* og *Lagring af konverterede signaler* er fortaget ved hjælp af udviklingssoftwaret til mc'en i form af C-Spy, der giver mulighed for at se variablerne i hukommelsen på mc'en. Testen er udført ved at sætte en signalgenerator på indgangen til ADC'en, og derefter sammenligne de samlede værdier med signalet på indgangen, hvilket viste at modulerne virkede tilfredsstillende. Der blev samlet på 3 kanaler, samplingsfrekvens var korrekt, bitopløsningen på signaler var på 12 bit. Alle konverterede kanaler bliver flyttet til RAM'en efter hver samplingssekvens, og den anden afledning bliver desuden gemt i en separat variabel, EKG\_ny\_f.





# Kapitel 9

## Kommunikation med PC

*I dette kapitel designes, implementeres og testes modulerne til seriel kommunikation via UART'en, bestående af Opsætning af UART modul og Seriel kommunikation modul.*

### 9.1 Modul: Seriel kommunikation og Opsætning af UART

De AD-konverterede data skal overføres serielt til pc'en via mc'ens indbyggede UART. Kommunikation følger RS-232-standarden, hvorfor der er tale om asynkron seriel overførsel.

Da initialiseringsmodul i høj grad afhænger af overførselsmodul, designes overførselsmodul inden initialiseringsmodul. Af hensyn til den kronologiske forståelse beskrives implementationen af initialiseringsmodul imidlertid før implementationen af overførselsmodul.

### 9.2 Design af Seriel kommunikation modul

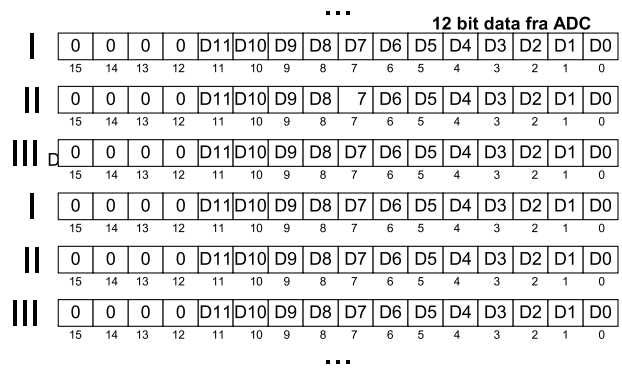
Kravene til *Seriel kommunikations modul* er:

- Hver transmitteret pakke skal indeholde afledningerne I, II og III i den nævnte rækkefølge
- Begyndelsen af hver pakke skal klart markeres

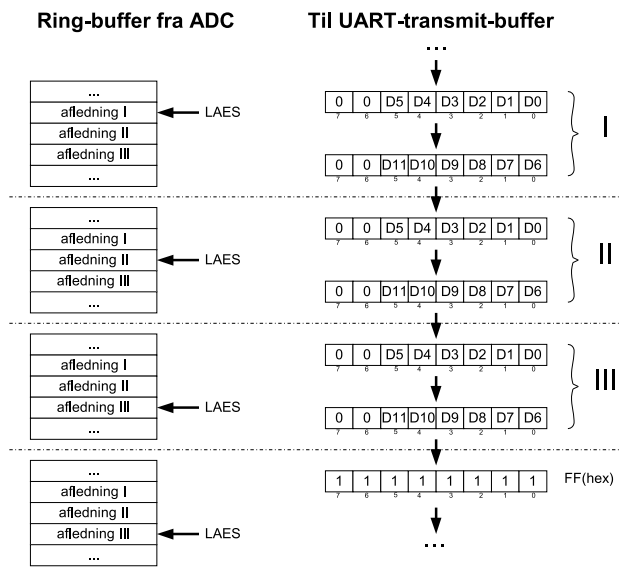
For at tømme ring-bufferen hurtigt nok, skal der sendes data til pc'en, hver gang UART'en er klar, og der er konverteret nye data i ADC'en. UART'en genererer et interrupt hver gang, transmitbufferen er klar til at modtage nye data. Dette interrupt udnyttes til at styre, hvornår der skal tildeles processortid til at overføre data til transmitbufferen. Overførselsmodul designes derfor som en interruptservicerutine for UART-transmit-interruptet. I ringbufferen ligger de konverterede data tilgængelige i formatet som vist på figur 9.1.

En pakke defineres nu som resultaterne af én konvertering fra hver af de 3 afledninger (I+II+III) - en pakke bliver da 3 data-words. Da UART'en imidlertid kun kan sende data i byte-format, må pakken splittes op i 6 portioner af én byte. For at markere afslutningen på en pakke tilføjes yderligere 1 byte, der kan skelnes entydigt fra de øvrige bytes. Til dette formål vælges en byte med værdien FF(hex) - denne byte adskiller sig entydigt fra de 6 bytes i pakken, såfremt pakken splittes op, så hver byte indeholder 2 faste nuller og 6 databits. Læsningen fra ring-bufferen og overførslen til transmit-bufferen foregår samlet som vist på figur 9.2.

Hver gang interrupt-servicerutinen udføres, skal én af de 7 bytes placeres i transmitbufferen og sendes. Når dette er gjort, testes det, om der er flere konverterede data i ringbufferen, der er klar til at blive sendt - det testes altså, om SKRIV-pointeren er foran LAES-pointeren i ringen. Er dette tilfældet, tillades yderligere interrupts fra UART-transmitteren, og der sendes nye data, næste gang transmitteren er klar.



Figur 9.1: Figuren viser, hvordan data ligger i bufferen: de konverterede data (12 bit) ligger lagret i ringbufferen i 16-bit-words. Rækkefølgen er som vist afledning I, afledning II, afledning III, afledning I... osv.



Figur 9.2: Sådan fyldes data og kontrol-byte i transmitbufferen.

Er dette ikke tilfældet, deaktiveres UART-interruptmekanismen. Hermed sendes der ikke flere data, inden der på ny konverteres data i ADC'en - i interrupt-servicerutinen for ADC'en aktiveres UART-transmit-interrupt igen, og der kan på ny overføres data via UART'en. På denne måde sikres det, at der ikke foretages unødigt mange kald af UART-interrupt-servicerutinen.

### 9.3 Design af Opsætning af UART modul

Kravene til *Opsætning af UART* er:

- UART'en skal opsættes til asynkron kommunikation
- Der skal opsættes til at 3 kanaler, hver samplet med 500 Hz, kan transmitteres
- Der skal sikres imod fejl i transmissionen

Designet af *Opsætning af UART modulet* indeholder fire valg: Antallet af data-bits, antallet af stop-bits, valg af paritet, samt valg af baudhastighed. ADC'en konverterer data med en samplingsfrekvens på 500 Hz. Med den valgte overførselsrutine, skal der således overføres i alt 7 bytes fra UART'en til pc'en 500 gange i sekundet. For at få et nøjagtigt billede af den nødvendige båndbredde for transmissionen, er det nødvendigt at inkludere antallet af start-bits, data-bits, paritet-bits og stop-bits i det samlede regnskab.

Det er givet, at UART'en benytter 1 start-bit. Med den valgte overførselsprocedure er der brug for 8 data-bits, og 1 stopbit er tilstrækkelig til formålet, da den maksimale hastighed for det serielle interface er 57.600 baud grundet optokoblerens tidsegenskaber. Der vælges lige paritet for at danne mulighed for fejlkontrol under transmissionen. I alt vil hver af de 7 bytes, der skal sendes pr. pakke indeholde 1 start-bit, 8 data-bits, 1 paritet-bit og 1 stopbit - d.v.s. 11 bits. 7 bytes af 11 bits 500 gange i sekundet giver en krævet baudhastighed på 38.500 bit/s. Der vælges derfor en baud-hastighed på 57.600 baud.

### 9.4 Implementation af Opsætning af UART modul

Modulet er implementeret i filen "AD konvertering og seriel transmission.c".

UART-registrene indstilles til følgende: 8 databits og lige paritet. Submasterclocken (7,3728 MHz) underdeles med 128 for at generere baudclocken på 57.600 baud. UART-transmit-interrupt aktiveres.

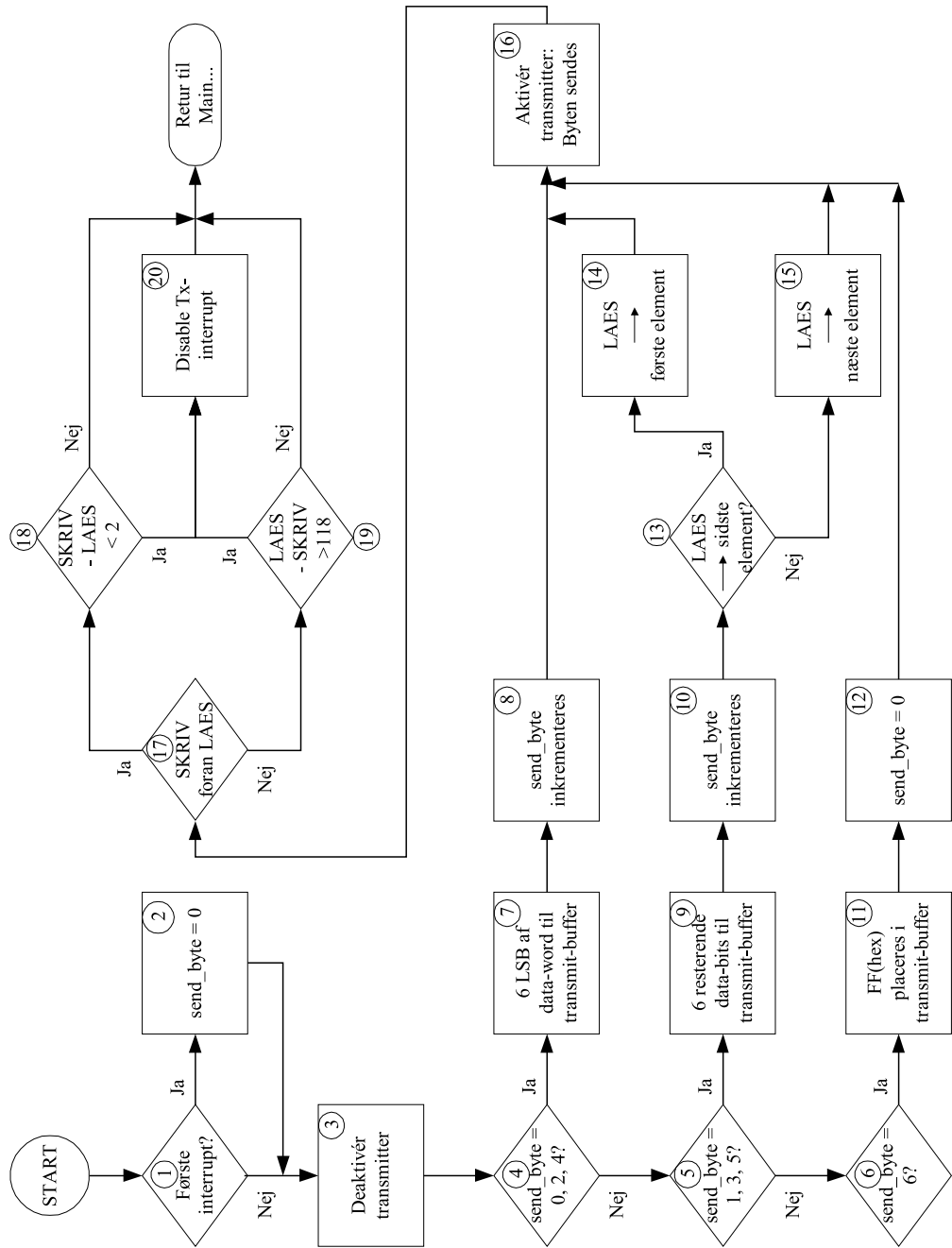
### 9.5 Implementation af Seriel kommunikation modul

Modulet er implementeret i filen "AD konvertering og seriel transmission.c" som interruptservicerutine for UART0TX\_VECTOR.

Implementationen af UART-transmit-interrupt-servicerutinen fremgår af figur 9.3.

I følgende udredning benyttes nummereringen, der fremgår af figurens elementer. Ved første kald (1) af rutinen nulstilles variabelen "send\_byte"(2). Denne variabel angiver, hvilken af de 8 bytes, der skal sendes under dette interrupt.

Ifølge manualen til mc'en, kan der opstå transmissionsfejl, hvis der fyldes data i transmit-bufferen, mens transmitteren er aktiveret. Derfor deaktiveres transmitteren, inden der foretages yderligere (3). Send\_byte kan antage værdier fra 0 til 6. En switch-struktur (4,5,6) afgør udfra værdien af send\_byte, hvad der skal placeres i transmit-bufferen. Når send\_byte antager én af værdierne 0,2,4, placeres de 6 mindst betydende bits af det dataord, LAES peger på, i transmit-bufferen (7) og send\_byte inkrementeres (8).



Figur 9.3: Flowdiagram for Serial kommunikaton modul

Når send\_byte antager én af værdierne 1,3,5, placeres de 6 resterende databits fra det dataword, LAES peger på, i transmit-bufferen (9) og send\_byte inkrementeres (10). Hvis LAES peger på det sidste element i arrayet (13), sættes LAES til at pege på det første element (14) - ellers sættes LAES til at pege på det næste element i arrayet (15), og dermed den næste afledning. Som afslutning på en pakke antager send\_byte værdien 6, FF(hex) placeres i transmit-bufferen (11), og send\_byte nulstilles(12), så der startes forfra ved næste kald af interruptservicerutinen. Der er nu fyldt data i transmit-bufferen, og det er derfor sikkert at aktivere UART-transmitteren (16).

Hermed sendes byten over til PC'en. Det skal nu testes, om SKRIV kun er et element foran LAES i ring-bufferen (17, 18, 19) - i så fald deaktiveres UART-transmit-interrupts(20), indtil der konverteres nye data i ADC'en. Interruptservice-rutinen er overstået, og der returneres til Main.

## 9.6 Test af Kommunikat med pc

Testen af den seriel kommunikation er lavet ved at sætte et hyperterminalprogram på en PC til at modtage det fra UART'en. Hyperterminal programmet blev sat op til den ønskede hasighed på 57.600 baud rate, 8 databits og lige paritet. Testen vektoren blev generet fra mc'en, som satte UART'en op til at sende et bestemt tal over kommunikationen hele tiden. Testvektoren kom korrekt igennem, hvilket betyder at den seriel kommunikation virker efter hensigten.



# Kapitel 10

## Pulsdetektering

*I dette kapitel designes, implementeres og testes modulerne QRS-detektering, kalibrering og beregning af puls, der udgør processen Pulsberegning. Først fastlægges derfor, hvordan QRS-komplekser skal detekteres, da design af både QRS-detekterings- og kalibreringsmodulerne skal foretages herudfra.*

*I kapitlet angives størrelser af EKG-signalet som et tal uden enhed, da de svarer til samplede værdier. Størrelser af den afledede af EKG-signalet angives også som tal uden enhed, men det fremgår af sammenhængen, hvilket signal (differentieret eller ikke-differentieret) der er tale om.*

### 10.1 Valg af detekteringsmetode

Pulsen angiver antallet af gange, hjertet slår pr. minut. Derfor må beregning af puls ud fra et EKG-signal gøres ved at detektere en fællesnævner i de enkelte hjerteslag. Herved opnås viden om, hvor mange gange hjertet har slået over en periode, og pulsen kan beregnes.

En sådan fællesnævner kan være en enkelt tak i eller en del af PQRST-komplekset, fx flere takker i rækkefølge.

I princippet er alle takkerne i PQRST-komplekset detekterbare, men detektering af forskellige takker giver forskellig kompleksitetsgrad.

Det mest udbredte er at detektere QRS-komplekser [4], så denne fremgangsmåde vælges også i dette projekt.

#### 10.1.1 QRS-detektering

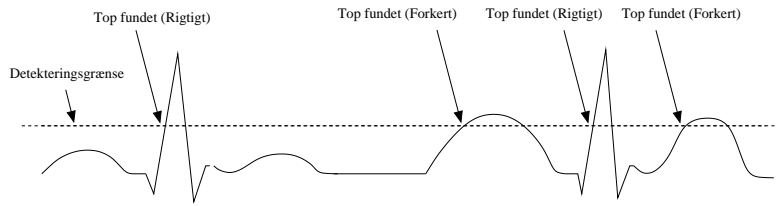
Detektering af QRS-komplekser kan foregå på forskellige måder. Den amplitudemæssigt mest udprægede del af komplekset er R-takken, så alene ud fra amplitude vil den være den letteste at detektere.

QRS-detektering alene ud fra R-takkens amplitude kan dog forstyrres hos visse patienter: hvis patienten har en unormalt høj P- eller T-tak, kan disse fejlagtigt detekteres som R-takker. En sådan situation er vist på fig. 10.1.

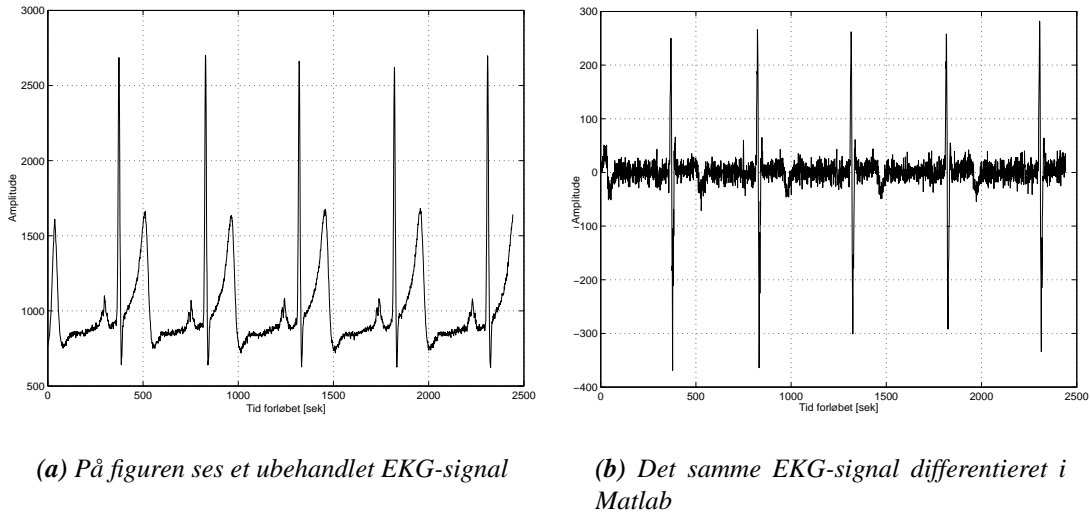
Detektering af R-takker ud fra dens *hældning* vil derimod sikre, at en høj P- eller T-tak ikke detekteres som en R-tak, da hældningen af R- (og S-)takken er meget større end hældningerne af de resterende takker i QRS-komplekset. Takkernes hældninger i PQRST-komplekset findes naturligvis ved differentiation af EKG-signalet.

På fig. 10.2 ses et ikke-differentieret PQRST-kompleks ved siden af et differentieret. Det ses, at R- og S-takkerne tilsammen udgør en tydelig markering, og det vælges derfor at detektere QRS-komplekser og dermed PQRST-komplekser ud fra hældningen af R-takken.





**Figur 10.1:** Figuren illustrerer, hvordan en detekteringsalgoritme fejlagtigt kan detektere flere R-takker i det samme QRS-kompleks, hvis der kun detekteres ud fra en fast amplitudegrænse for R-takken.



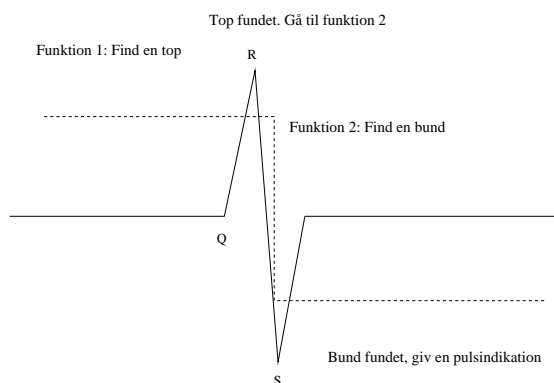
**Figur 10.2:** Figurene viser forskellen mellem et ikke-differentieret og et differentieret EKG. De tydelige udsving i det differentierede signal er R- og S-takkerne

På det differentierede signal kan det vælges kun at detektere R-takke ud fra overskridelse af en grænseværdi for amplituden af den differentierede, således at en overskridelse af grænseværdien betyder, at der er en R-tak. Herved kan der dog stadig opstå problemer med at der detekteres flere R-takke pr. QRS-kompleks. Dette skyldes, at *flere* differentierede værdier i R-takken kan være over grænsen, og at dette trigger detektering af flere R-takke, selvom der kun var tale om én.

Denne type fejl kan undgås ved at indføre en betingelse om en mindste-tid mellem detekterede R-takke. Med en sådan betingelse *skal* der altså gå et vist tidsrum, før en ny R-tak kan detekteres.

En anden måde at undgå flere trigninger på samme kompleks er at udvide detekteringsbetingelsen til ikke blot én overskridelse af grænse for den afledede men i stedet 2. Det ses nemlig af fig. 10.2, at der er en udtalt markering af R- og S-takkerne ved både stor positiv og stor negativ amplitude.

Herved detekteres et QRS-kompleks ved overskridelse af en grænse for positiv amplitude og efterfølgende overskridelse af en grænse for negativ amplitude. Den differentierede af QRS-komplekset består altså af en top og en bund. Ideen er illustreret på fig. 10.3.



**Figur 10.3:** Figuren illustrerer, hvorledes der først søges en top, og derefter en bund, når der er registreret en top

En algoritme, der detekterer QRS-komplekser på denne måde, søger altså først efter overskridelse af topgrænsen, og når denne er overskredet, søges efter overskridelse af bundgrænsen. Når der på denne måde først er fundet en top og derefter en bund, kan dette registreres som et QRS-kompleks.

Grænserne for top og bund, som QRS-komplekset detekteres ud fra, skal som nævnt i kravspecifikationen sættes til et niveau, der passer til EKG-signalet for hver patient. Med algoritmen nævnt ovenfor skal detekteringen altså tage højde for amplituderne af de store positive og negative udsving på det differentierede EKG-signal. Det er derfor valgt at lave en adaptiv kalibreringsrutine, der automatisk finder top- og bundgrænse for den enkelte patient, således at disse grænser anvendes i algoritmen.

Både selve QRS-detekteringen og kalibreringen skal altså benytte den differentierede af EKG-signalet, så det vælges at designe og implementere et hjælpe-modul, der forestår denne differentiation. Design og implementation af dette modul beskrives i det følgende.

## 10.2 Afledet Beregning

Implementation af modulet findes i filen “QRS-detektering.c”, funktionen “differentier”.

Differentiation ses som differensen mellem den  $n$ 'te sample i forhold til en foregående sample antagende værdien  $n - k$ , hvor  $k$  kan antage hvilken som helst værdi. Konstanten  $k$  fortæller over hvor mange samples der differentieres. Dette kommer til udtryk i formel 10.1.

$$y[n] = \frac{x[n] - x[n - k]}{k} \tag{10.1}$$

Ved z-transformation af differensligningen, ses at ligningen gennem overføringsfunktionen  $H(z)$  antager funktionaliteten af et højpasfilter.

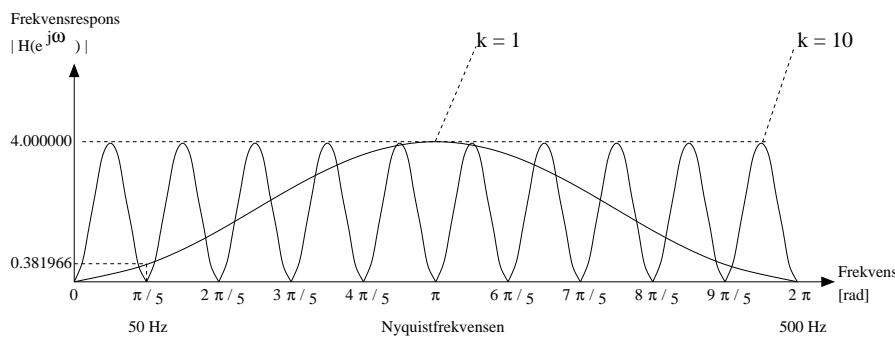
$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 - z^{-k}}{k} \tag{10.2}$$

Dette betyder at signalets lavfrekvente områder dæmpes og højfrekvente områder som QRS-komplekset fremstår klarere.

En større  $k$  værdi sikrer at 50 Hz støj minimeres ved differentiationen.

Ved en optimal differentiation, hvor 50 Hz støj er minimeret, fås en  $k$ -værdi til 10. Dette forklares i forbindelse med figur 10.4.

Grundlaget for at drage en sammenligning mellem en  $k$ -værdi på 10 og 1, skyldes at Matlab's differentiationsfunktion bygger på en  $k$ -værdi på 1. I forbindelse med implementering af c-koden, benyttes differentiationsfunktionen i Matlab i en sammenlignende test.



**Figur 10.4:** Amplitude-karakteristik for  $k$ -værdierne 1 og 10. Herudfra kan ses at valget af en lavere  $k$ -værdi vil resultere i at 50 Hz støj vil passere gennem filteret. Ved  $k = 1$ , vil frekvensresponsen være ca. 0,38, hvilket betyder at ca. 10 % af støjen vil passere gennem filteret.

$k$ -værdien findes gennem først at substituere  $z$  med  $e^{j\omega}$ , hvorved et udtryk for frekvensresponsen findes. Dette ses ved formel 10.3.

$$|H(e^{j\omega})| = \frac{|1 - (e^{j\omega})^{-k}|}{|k|} \tag{10.3}$$

Da  $(e^{j\omega})^{-k}$  er lig  $e^{-j\omega k}$ , samt da  $e^{-j\omega k}$  kan udtrykkes ved  $\cos(-\omega k) - j\sin(-\omega k)$ , kan formel 10.3 udtrykkes ved:

$$|H(e^{j\omega})| = \frac{1}{k} \cdot |1 - e^{-j\omega k}| = \frac{1}{k} \cdot |1 - \cos(-\omega k) - j\sin(-\omega k)|$$

og altså

$$|H(e^{j\omega})| = \frac{1}{k} \cdot \sqrt{2 - 2\cos(\omega \cdot k)} \tag{10.4}$$

Ved at benytte denne ligning for frekvensresponsen kan det beregnes, hvilken værdi  $k$  skal have, for at filteret dæmper 50 Hz-støj mest muligt. Det skal altså beregnes, ved hvilken  $k$ -værdi, filteret har et nulpunkt.

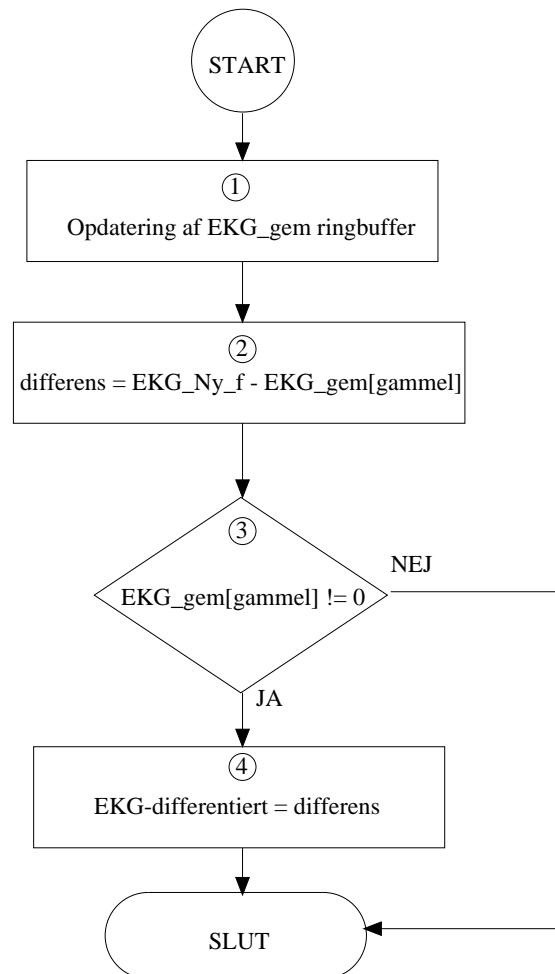
Pga.  $\cos$ -leddet i 10.4 har frekvensresponsen nulpunkter, hvor  $\omega \cdot k$  er 0 (dvs.  $2\pi$ ). Der ønskes altså en  $k$ -værdi, som for  $\omega$  svarende til 50 Hz opfylder  $\omega \cdot k = 2\pi$ .

Da der samples med 500 Hz, er Nyquist-frekvensen 250 Hz, så der findes i signalet ikke frekvenskomponenter over denne frekvens.  $\pi$  svarer dermed til 250 Hz. Dette giver  $k$ -værdien 10.

På fig. 10.4 er dette vist. Det ses, at differentiationen giver et veldefineret og relativt skarpt nulpunkt ved 50 Hz. En endnu mere frekvens-selektiv differentiation kunne opnås ved at anvende en 2- 4- osv. dobbelt  $k$ -værdi. For at undgå aliasering kræver dette dog tilsvarende fordoblinger af samplingsfrekvensen, så 10 er den bedste  $k$ -værdi.

### 10.2.1 Implementation af Afledet beregnings modul

Differentiation af EKG-signalet antager den matematiske udformning vist ved formel 10.1. Algoritmen er implementeret som vist på figur 10.5;



**Figur 10.5:** Figuren illustrerer flowet for differentiationen. Returnering til main sker kun én gang.

Opdatering af EKG\_gem ringbufferen (1) sikrer at der hele tiden tælles op for pladser i bufferen, således at ny værdier lagres og gamle værdier overskrives i de 11 pladser der er afsat i bufferen. Hvis bufferen når sin ende, skal funktionen automatisk begynde at overskrive i starten i bufferen og fortsætte derfra.

I differentiationen (2) tages der forbehold i forhold til formel 10.1. Der vælges ikke at dividere med 10 som foreskrevet i formelen, da en division er en tidskrævende operation der ønskes undgået ved beregning af den afledede for hver enkelt sample.

Dette gør at amplituden af signalet vil være væsentlig større, end hvis divisionen blev gennemført. Dette har dog ingen betydning, da det blot er en forøgelse af amplituden for alle dele af signalet og derfor ikke har nogen betydning for detekteringen af QRS-komplekset.

EKG\_gem[gammel] kan give en usandsynlig høj værdi i det tilfælde, hvor funktionen ikke har været gennemløbet af 10 værdier. Dette gør pga. headroom'et der er inført i forbindelse med filtreringen, at hvis den første værdi der gennemløber differentiationen er toppen af en R-tak, at den

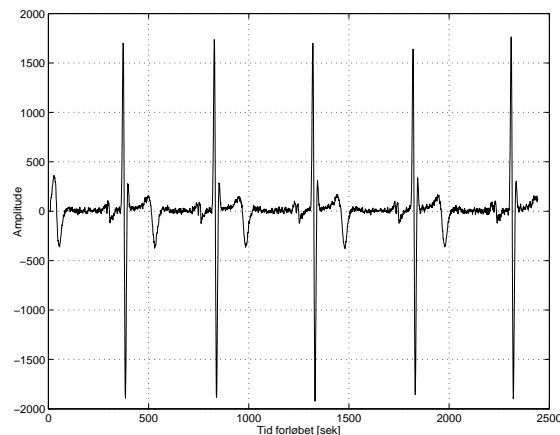
returnerede differentierede værdi er meget større end hvad tilfældet ville være ved differensen mellem den  $n$ 'te sample og den  $n-k$ 'te sample efter at 10 værdier har gennemløbet funktionen. Da en `EKG_gem[gammel]` værdi på nul kan give en usandsynlig høj værdi, ønskes kun at funktionen skal returnere en differentieret værdi i det tilfælde, hvor `EKG_gem[gammel]` er større end nul (3,4).

### 10.2.2 Test af differentiationsalgoritme

Differentiations algoritmen er testet ved at hente det ikke-differentierede signal ind i Matlab, benytte den udviklede algoritme og gemme det differentierede signal i en ny fil.

Der tages udgangspunkt i det ikke-differentierede signal illustreret i figur 10.2, samt det i Matlab differentierede signal.

Differentiationsfunktionens output er illustreret ved figur 10.6.



*Figur 10.6: Figuren illustrerer et EKG-sample differentieret i Matlab vha. den udviklede algoritme.*

Efterfølgende er outputtet sammenlignet med outputtet for det samme signal differentieret i Matlab's egen differentiationsalgoritme, figur 10.2.

Der blev fundet en tilfredsstillende lighed mellem de to signaler.

## 10.3 Kalibrerings modul

Kravene til dette modul er:

- I begyndelsen af hver brugssituation skal størrelsen af de parametre, QRS-komplekser skal detekteres ud fra, fastlægges ud fra det samlede EKG
- Størrelsen skal gøres tilgængelig for QRS-detekteringsmodulet.

### 10.3.1 Design af kalibreringsmodul

Den adaptive kalibrering skal med den valgte QRS-detekteringsmetode finde amplituderne for top og bund i den differentierede af EKG-signalet.

Det er hensigtsmæssigt at lade kalibreringen foregå over et antal pulsslæg, således at grænseværdierne angiver gennemsnit af fundne hhv. top- og bundamplituder. Denne gennemsnitsberegning skal sikre, at der tages højde for mindre variationer i amplituden af signalet under netop kalibreringen. Ofte sættes grænseværdier, der findes ved en gennemsnitsberegning som denne, til et stykke under det fundne gennemsnit [15].

Herved bliver også et QRS-kompleks, hvor amplituderne af top og bund i den differentierede er lidt under og over den midlede værdi, detekteret. I [15] anbefales det at sætte grænsen til 80 % af den midlede grænse, men det er projektgruppens erfaring, at nogle komplekser ikke detekteres med denne grænse. Det er derfor valgt at sætte grænsen til 70 %, da denne grænse virker.

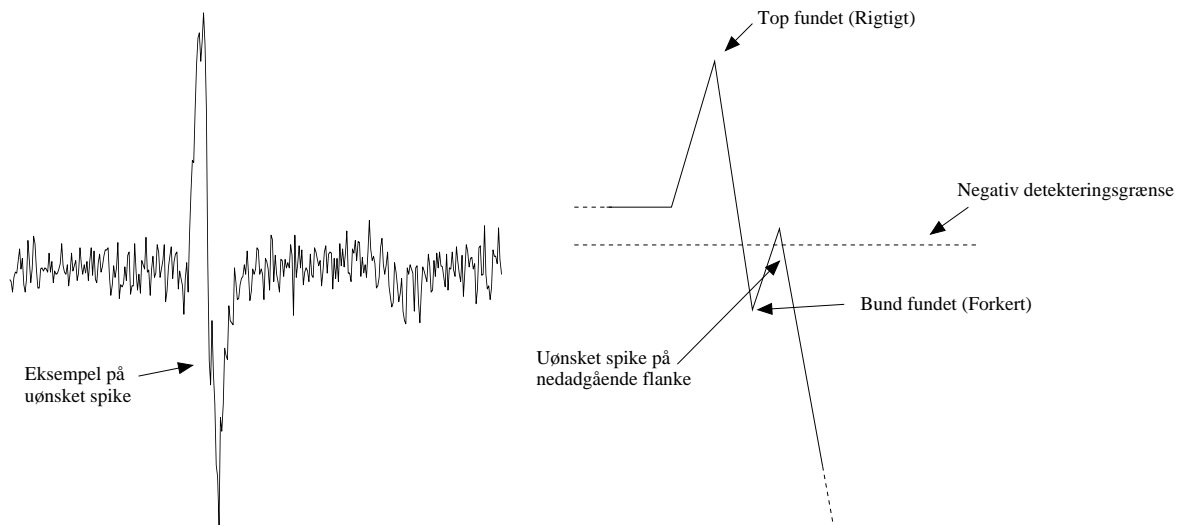
Herved fås en grænseværdi for topdetektering og en grænse for bunddetektering, der passer til den enkelte patient.

Det ses på fig. 10.2 og fig. 10.7, at de isoelektriske stykker mellem PQRST-komplekserne også på det differentierede signal indeholder en del spændingsvariationer og altså ikke er nul. Det er derfor nødvendigt at fastsætte en grænse for, hvornår der må søges efter en top hhv. bund – ellers risikeres det fx, at en af de små toppe i det isoelektriske detekteres som en “rigtig” top og altså giver fejl i den midlede værdi. For at undgå dette vælges en initialiseringsværdi, som angiver minimumamplituden af toppe og bunde.

### Valg af initialiseringsværdi

Initialiseringsværdien for søgning af top er valgt ud fra granskning af det differentierede, optagne EKG; den er sat til -1000. Dette gør, at der skal søges efter top, indtil der er sikkerhed for, det er S-takken, der detekteres som en bund - der skal søges efter en top, indtil signalet går under initialiseringsværdien på -1000.

Selv med en initialiseringsværdi kan der opstå problemer med spikes, se figur 10.7.



(a) Et differentieret EKG-signal med spike på den nedadgående del

(b) Fortegnet skitse af differentieret signal med spike tilsvarende den på (a). Spike'n ses tydeligt sammen med konsekvenserne, hvis detekteringsalgorimen ikke tager højde for spikes

**Figur 10.7:** På figuren ses spike på EKG-signal samt konsekvenserne af spikes

For at undgå, at sådanne spikes kan have indvirkning på detekteringen af bund, hæves grænsen for søgning af bund. Amplituden af de uønskede spikes kan nemlig resultere i, at der detekteres et lokalt minimum, istedet for en bund jvf. figur 10.7. Det er derfor valgt at sætte grænsen for søgning af bund til -500. Derved søges efter bund, indtil -500 overskrides (dvs. indtil værdierne af den afledede er mindre negative end -500), og grænsen sættes til -1000, så der startes forfra med søgning efter top.

### 10.3.2 Implementation af kalibreringsmodul

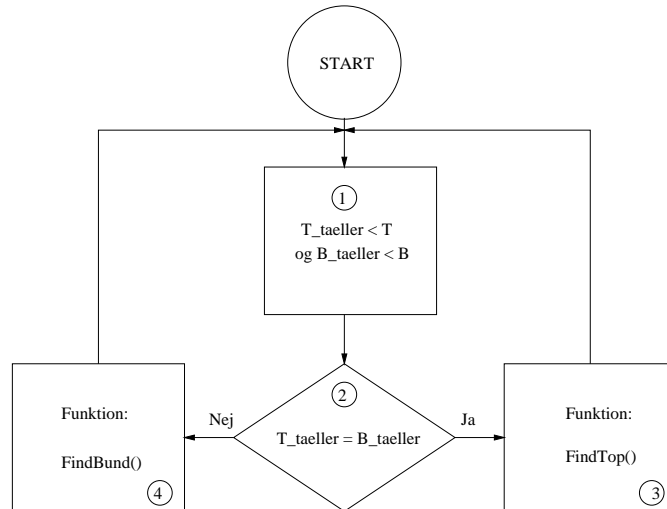
Modulet er implementeret i filen “Kalibreringsmodul.c”.

Kalibreringsrutinen bygger på, at amplituderne for et vist antal toppe og bunde findes for patientens EKG. Ud fra disse værdier findes en middelværdi, der anvendes som faste amplitudegrænser i QRS-detekteringen.

Der henvises til flowdiagrammet 10.8; tal i parentes henviser til denne figur.

Amplituderne for fundne toppe og bunde gemmes i bufferne TopBuffer[] og BundBuffer[], med længderne hhv. T og B. Hvor mange toppe hhv. bunde der er fundet og lagt i bufferne, angives af tællerne T\_tæller og B\_tæller. Variablene opdateres, når en top hhv. bund er fundet.

Kalibreringsrutinen er kun aktiv, så længe T\_tæller er større end T, og B\_tæller større end B, se (1) på figur 10.8, dvs. hvis bufferne for toppe og bunde ikke er fyldt med værdier.



**Figur 10.8:** Figuren viser flowdiagram for kalibreringsrutinen. FindTop og FindBund funktionerne beskrives senere

I while-løkken findes to if-sætninger. Disse skal sikre, at der skiftevis søges top og bund, og at der først søges en top. Den første if-sætning (2) har betingelsen, at T\_tæller er lig B\_tæller. Dette fortæller, at der skal søges en top (gennem FindTop funktionen, se figur 10.8), hvis der er lige mange pladser opbrugt i bufferne.

På figur 10.8 ses flowdiagram for kalibreringsmodulet. Her er betingelsen for søgning af toppe og bunde illustreret.

#### FindTop

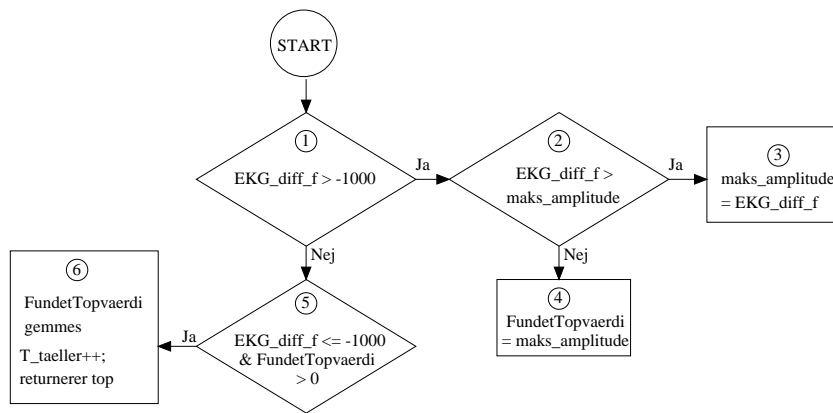
Der refereres til figur 10.9; tal i parentes henviser til denne figur..

En if-betingelse (1) sikrer, at hvis den formelle parameter EKG\_diff\_f er større -1000, skal der søges en top ud fra to betingelser; hvis EKG\_diff\_f er større end den hidtil fundne maksimale positive værdi (2) (gemt i maks\_amplitude), skal EKG\_diff\_f overskrive maks\_amplitude (3), ellers skal maks\_amplitude gemmes som den hidtil største værdi, i FundetTopVaerdi (4).

Derimod, hvis EKG\_diff\_f mindre eller lig -1000 mV og FundetTopvaerdi større end 0 (5), gemmes FundetTopvaerdi i TopBuffer[] (6) og T\_tæller opdateres.

FundetTopvaerdi bliver kun returneret (6), hvis den største positive amplitude er fundet og gemt. Ellers returneres 0.

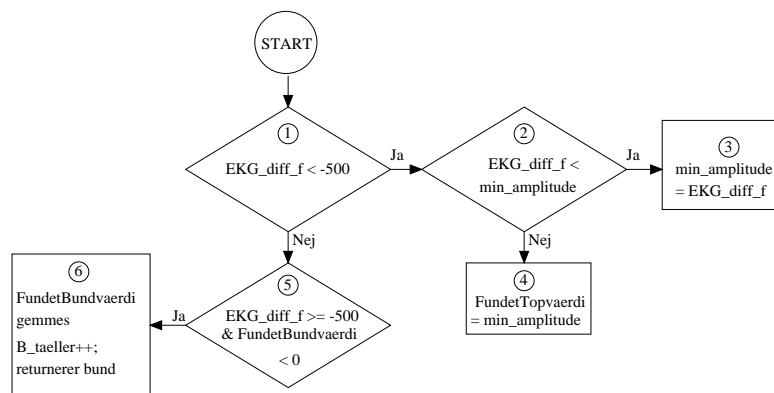
Efter at der er fundet en top, køres while-løkken ((1) på fig. 10.8 igen. Hvis bufferne ikke er fyldt, søges i stedet en bund gennem funktionen FindBund((4), stadig på 10.8).



Figur 10.9: Figuren flowdiagram for funktionen FindTop

## FindBund

Der refereres til figur 10.10; tal i parentes henviser til denne figur.



Figur 10.10: Figuren illustrerer flowdiagram for funktionen FindBund

FindBund har samme funktion som FindTop funktionen, blot med tre ændringer.

Den første ændring er, at der kun søges en bund, hvis EKG\_diff\_f er mindre end -500 (1).

Betingelserne for videre at gemme en fundet bund er, at EKG\_diff\_f skal være mindre (dvs. mere negativ) end den hidtil maksimalt negative gemte værdi (2). I så fald skal EKG\_diff\_f overskrive min\_amplitude (3); ellers skal min\_amplitude gemmes i FundetTopVaerdi (4).

Hvis EKG\_diff\_f er større eller lig -500 og FundetBundvaerdi mindre end 0 (5), gemmes FundetBundvaerdi i BundBuffer[], FundetBundvaerdi gemmes, B\_tae++ opdateres (6).

FundetBundvaerdi bliver kun returneret, hvis den største negative amplitude er fundet og gemt. Ellers returneres 0.

## Grænsedetektering

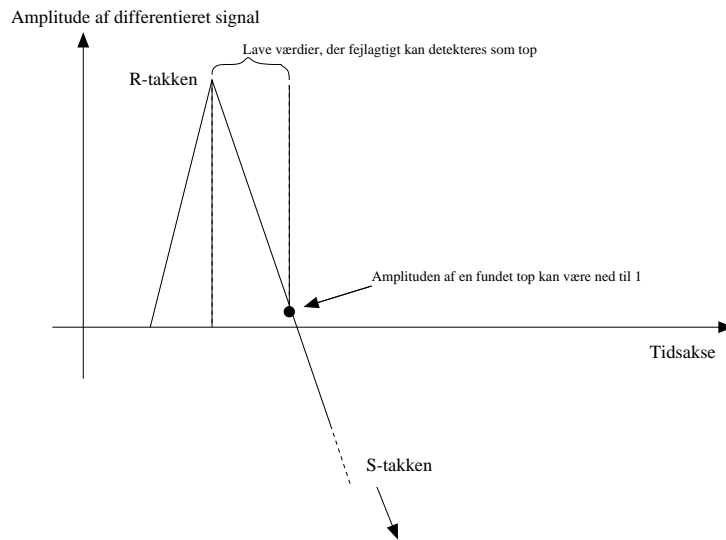
Når betingelsen for while-løkken (1) på figur 10.8 er falsk, findes detekteringsgrænser gennem funktionen GraenseDetect().

I denne funktion beregnes gennemsnittene af værdierne i TopBuffer[] og BundBuffer[].

Ved gennemsnitsberegning for TopBuffer[] vælges det dog ikke at medregne værdien for den første top. Dette gøres, da kalibreringsrutinen måske påbegyndes på et tidspunkt, hvor den differentierede er efter en top og før 0, hvorved den fundne værdi for top bliver fejlagtigt lav. Da betingelsen for lagring af toppe i FindTop funktionen ikke tager højde for dette, kan den første top "worst case" antage værdien 1, men mere sandsynligt en hvilken som helst værdi mellem toppen og 0. Dette giver et opstartsproblem, der kan have stor betydning for fastsættelsen af



grænser. Det samme problem haves ikke for start mellem bund og 0, da der kun påbegyndes søgning efter bund, når der er fundet en top – og der er jo ikke fundet en top, hvis kalibreringen startes mellem bund og 0.



**Figur 10.11:** Figuren viser situationen, hvor en top fejlagtigt detekteres ved en lav værdi, fordi kalibreringen påbegyndes mellem en top og nul

Grænseværdierne sættes til at være 70 % af den midlede værdi som nævnt i afsnit 10.3.1. De to grænseværdier tilskrives variablene GrænseTop og GrænseBund, således at disse er tilgængelige for QRS-detekteringsmodulet.

### 10.3.3 Test af kalibreringsmodul

Testen af kalibreringsmodulet er gennemført i 2 dele: først er evnen til at finde toppe og bunde testet, og herefter er funktionaliteten med midling og beregning af grænser herudfra testet. Begge tests er foretaget som input/output-tests.

#### Test af top- og bunddetektering

Denne funktionalitet er testet vha. en fil, der indeholdt den differentierede af samplede data for 5 QRS-komplekser, dvs. 5 toppe og 5 bunde. Filen blev hentet ind i programmet, og programmet kørt med filen som input. Før dette blev top-værdier og bundværdier for hvert QRS-kompleks manuelt identificeret og nedskrevet som en facit-liste, så det kunne testes, at modulet finder de korrekte toppe og bunde. Efter at filen var kørt igennem modulet, aflæstes værdierne i de to buffers, og dette viste, at alle toppe og bunde var korrekt identificeret.

Værdierne er vist på tabellerne 10.3.3 og 10.3.3, og bruges i testen af midlingen.

T_tæller	Manuelt aflæst værdi	Detekteret top
0	1712	1712
1	1700	1700
2	1736	1736
3	1699	1699
4	1646	1646
5	1764	1764

B_tæller	Manuelt aflæst værdi	Detekteret bund
0	-1893	-1893
1	-1883	-1883
2	-1923	-1923
3	-1857	-1857
4	-1896	-1896
5	-1879	-1879

### Test af midling og grænsefastsættelse

Efter identificeringen af toppe og bunde i den første test, undersøgte værdierne af de beregnede gennemsnit og de beregnede grænser. Værdierne er ganget med 0.7 som foreskrevet af grænse-detekterings-algoritmen. Manuel beregning af gennemsnittene giver hhv. 1196 og -1321, mens resultatet fra modulet var 1196 og -1321. Til grænse-detekteringen er der kun midlet over 5 af de 6 toppe, idet den første top skal frasorteres jvf. 10.3.2

Samlet fungerer modulet altså efter hensigten, da de korrekte værdier for grænseværdier for detektering af toppe og bunde returneres.

## 10.4 QRS-detekteringsmodul

Kravene til dette modul er :

- QRS-komplekser skal detekteres, hvis det samlede EKG er i overensstemmelse med de parametre, *Kalibrering* har fundet størrelsen af
- Hvis der er detekteret et QRS-kompleks, skal der returneres en markering til main

### 10.4.1 Design af QRS-detekteringsmodul

Design af QRS-detekteringsmodulet bygger på overvejelserne i 10.1.1. Derved fås en QRS-detektering, der detekterer toppe hhv. bunde, når grænserne fastsat i kalibreringsmodulet er overskredet af det differentierede signal. I denne QRS-detektering defineres et QRS-kompleks til at bestå af en top efterfulgt af en bund.

Der skal jvf. kravspecifikationen sættes en markering, når der er detekteret et QRS-kompleks.

### 10.4.2 Implementation af QRS-detekteringsmodul

Modulet findes implementeret i filen "QRS-detektering.c".

Signalet EKG\_diff\_f kategoriseres i tre niveauer; værdier over grænsen for detektering af top, GraenseTop, værdier under grænsen for detektering af bund, GraenseBund, samt værdier som findes imellem de to grænser.

Ud fra mønsteret af grænseoverskridelser – beskrevet ud fra variablene NyVaerdi og GammelVaerdi – detekteres QRS-komplekserne:

Når en værdi af det differentierede signal overskrider GraenseTop, tilskrives NyVaerdi værdien 1.

Hvis værdien derimod overskrider GraenseBund, tilskrives NyVaerdi værdien 0.

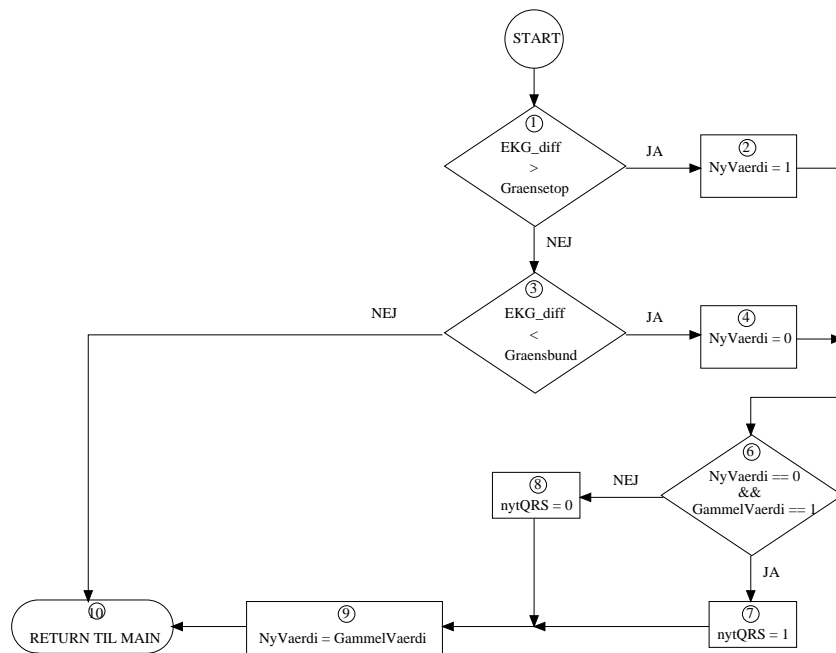
Hvis værdien befinder sig mellem de to grænser, ændres NyVaerdi ikke.

NyVaerdi gemmes i GammelVaerdi efter hver gennemløb.

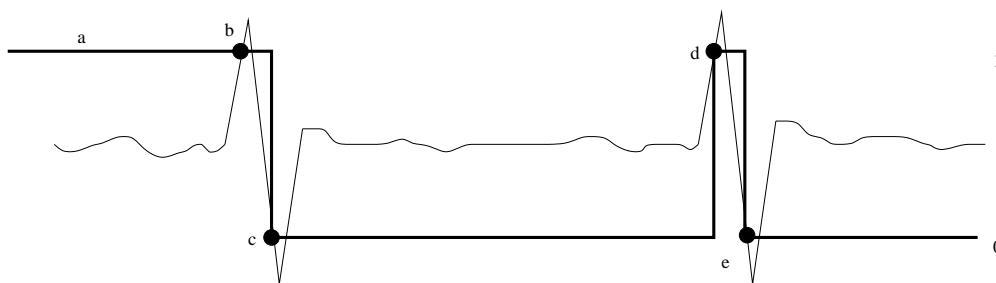
Figur 10.12 illustrerer QRS-detekteringsmodulet.

GraenseTop og GraenseBund refererer til de beregnede grænser fra kalibreringsmodulet.

Ved initialisering af modulet tilskrives GammelVaerdi værdien 1.



Figur 10.12: Figuren viser flowdiagram for QRS-detekteringen. Der returneres kun en markering af fundet QRS-kompleks, hvis en bund er fundet efter en top.



Figur 10.13: Ved (a) er GammelVaerdi initialiseret til 0. Ved (b) detekteres en top, så NyVaerdi sættes til 1. GammelVaerdi er stadig 1. Mellem (b) og (c) er GammelVaerdi=NyVaerdi. Ved (c) detekteres en bund, så NyVaerdi sættes til 0. Da GammelVaerdi er 1, er der detekteret et QRS-kompleks, så en markering gives. Fra (c) til (d) er GammelVaerdi=NyVaerdi. Ved (d) gentages fra (a)

Princippet for QRSdetekteringen illustreres på figur 10.13.

Et QRS-kompleks detekteres altså kun, hvis NyVaerdi er 0 (der er lige detekteret en bund) og GammelVaerdi er 1 (Sidste grænseoverskridelse var en top). Dette sker kun det øjeblik, EKG\_diff\_f er mindre end GraenseBund. Ved det efterfølgende gennemløb vil både NyVaerdi og GammelVaerdi være 0, så der detekteres kun ét QRS-kompleks.

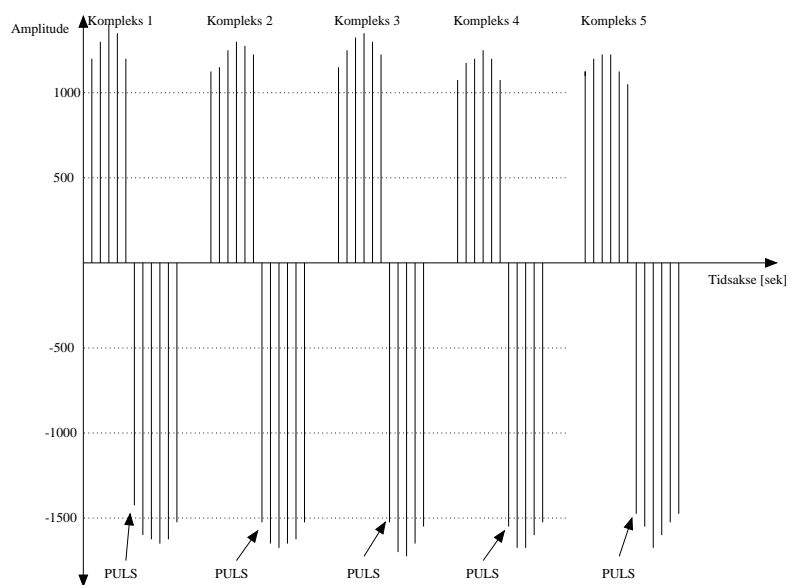
### 10.4.3 Test af QRSdetekteringsmodul

Testen er foretaget ved at undersøge, hvornår der detekteres QRS-komplekser.

En fil med differentierede samples af 5 QRS-komplekser er brugt som input til modulet, og alle interessante værdier er udskrevet på skærmen. Det vil sige, at værdier, som er relevante for detekteringen, blev udskrevet i et terminalvindue. Der tænkes her på værdier som overskred detekteringsgrænserne for top og bund. Disse grænser var sat til 1200 for top og -1300 for bund. Alle værdier over den øvre detekteringsgrænse blev udskrevet sammen med en markering, "HØJ", der indikerer, at værdien er over grænsen, mens alle værdier under den nedre detekteringsgrænse er udskrevet sammen med markeringen "LAV", der indikerer, at det er en værdi, som er under grænsen for bund-detekteringen.

Hvis der blev detekteret en "HØJ" som den sidste grænseoverskridelse før en "LAV", blev det på skærmen markeret, at der er detekteret et QRS-kompleks i hht. figur 10.13. Denne markering kom i form af udskrivning af ordet "PULS" på skærmen.

I figur 10.14 ses de høje og lave værdier, der blev udskrevet på skærmen, repræsenteret ved stave.



**Figur 10.14:** Figuren illustrer output for et terminalvindue som stave. Når et QRS-kompleks findes ved at en "HØJ"-værdi efterfølges af en "LAV"-værdi, og komplekset indikeres ved "PULS" Komplekserne er benævnt 1-5

De eksakte værdier for "HØJ" og "LAV" ved de forskellige komplekser angives her:

#### Kompleks 1

Høje værdier; 1395, 1587, 1700, 1674, 1482

Lave værdier; -1492, -1804, -1869, -1893, -1763, -1549

#### Kompleks 2

Høje værdier; 1250, 1450, 1621, 1736, 1687, 1447

Lave værdier; -1557, -1851, 1883, -1861, -1750, -1524

#### Kompleks 3

Høje værdier; 1303, 1505, 1645, 1699, 1611, 1349

Lave værdier; -1670, -1866, -1923, -1818, -1628

#### **Kompleks 4**

Høje værdier; 1267, 1439, 1576, 1640, 1543, 1271

Lave værdier; -1619, -1802, -1857, -1760, -1557

#### **Kompleks 5**

Høje værdier; 1398, 1610, 1743, 1764, 1550, 1245

Lave værdier; -1462, -1775, -1896, -1857, -1739, -1567

Efter den første grænseoverskridelse af bund-grænsen kom markeringen “PULS” én og kun én gang, så det kan konkluderes, at QRS-detekteringsmodulet virker.

## **10.5 Design af Beregn puls modul**

Kravene til dette modul er følgende:

- Pulsen skal beregnes efter hvert detekteret QRS-kompleks
- Pulsen skal beregnes så der tages hensyn til både opdateringshastighed og præcision

Der er forskellige metoder til beregning af puls. Da pulsen er slag/minut, er en metode optælling af pulsslag over et antal sekunder, eksempelvis 6 sekunder.

Der vil dog være et unøjagtighedsproblem forbundet med dette. Da eksemplet med beregning ud fra slag i løbet 6 sek. er slag i løbet af en tiendedel af 1 minut, vil der kun kunne vises pulsværdier delelige med 10, altså 60, 70, 80 osv. Perioden hvor over pulsslagene tælles kunne derfor sættes op, men her vil der gå længere tid før en pulsændring opdages, da der nu vil blive midlet over en længere periode.

Disse problemer undgås ved, at anvende en metode hvor der ses på hvor lang tid der går mellem hvert QRS-kompleks. Tiden mellem komplekserne skal gemmes i en ringbuffer med plads til  $x$  tider. Herover skal gennemsnittet tages og pulsen bestemmes. Opdateringen af pulsen afhænger af hvor mange tider der gemmes i ringbufferen. Derfor vælges kun at gemme 3 tider i ringbufferen, hvorved der fås en hurtig opdatering af pulsen. Da tiden bestemmes meget præcist, kunne det vælges kun, at tage tiden mellem to QRS-komplekser. Derved vil pulsen opdateres med det samme der sker ændring. Grunden til at midle over tre værdier er at gøre systemet mere stabilt for f.eks ekstrasystoler, uden at gå på kompromis med opdateringshastigheden.

Tiden mellem to QRS-komplekser findes ved at anvende en af mc'ens timer-funktioner. Timeren ser på hvad “tiden” er når der modtages en markering fra QRS-detekteringen og sammenholder dette med “tidspunktet” for sidste detekterede kompleks.

Denne pulsregningsmetode giver dog et problem, hvis en patient eksempelvis får hjertestop. I dette tilfælde vil der aldrig komme en ny værdi i ringbufferen, da timeren vil fortsætte med at tælle, pga manglende markering fra QRS-detektion. Pulsen vil derved forblive den sidste pulsværdi, inden hjertestoppet forekom, og hjertestoppet vil derved ikke blive opdaget.

Dette kan løses ved at lade timeren have en tidsbetingelse på 3 sekunder. Hvis der går mere end 3 sekunder vil patienten have en puls på under 20, hvilket ikke ville være en normal puls, hverken for en patient på en kardiologisk afdeling eller en rask person. Der skal derfor sendes en alarm om asystoli i det tilfælde hvor der er mere end 3 sekunder mellem to QRS-komplekser.

### 10.5.1 Implementation af tidstæller

Modulet er implementeret i filen "Pulsberegningsmodul.c".

Funktionens to hoved egenskaber er at tælle tiden, der går mellem QRS-komplekserne, samt at give besked om hvis der ikke kommer et nyt QRS-kompleks inden 3 sekunder efter det foregående.

Timer B er anvendt til at tælle tiden mellem komplekserne, som er klokke forsynet af A-klokken på 32768 Hz. Timer B underdeler klokken med 8, hvilket gør at den tæller 4096 trin pr. sekund. Da det er en 16 bit tæller tæller den dermed til 65536, før den begynder forfra.

Når der kommer et QRS-kompleks toggler vi bitten, CCISO, mellem høj og lav. Der laves et capture både på den opadgående flanke og den nedadgående. Registret TBCCR0 indeholder det aktuelle "tidspunkt" hver gang der captures, hvilken kan bruges i pulsberegningen.

Dette capture giver en tid, der kan bruges til at beregne tiden mellem de to QRS-komplekser. Der trækkes den sidst gemte tid sidste\_tid, fra den nye tid i capture-registret TBCCR0, og dette ganges med en konstant for at få tiden i millisekunder. Denne konstant er givet ved at der som sagt bliver talt til 4096 på 1 sekund eller 1000 ms, derved  $1000/4096 = 0,244140625$ . Disse tider bliver gemt i en ringbuffer med plads til tre tider til brug ved pulsberegningen.

### 10.5.2 Implementation af Beregn puls modul

I forlængelse af at *Pulstæller* måler tiden mellem hvert QRS-kompleks, og lægger tiderne i et array på 3 elementer, skal pulsen ud fra disse værdier beregnes.

Dette er en simpel gennemsnitsberegning, hvor gennemsnittet findes ved:

$$Puls = \left( \frac{3}{Tid_1 + Tid_2 + Tid_3} \right) \cdot 60000 \quad (10.5)$$

Da værdierne i arrayet er gemt i enheden milisekunder, og puls er defineret som slag pr. minut, forlænges gennemsnitsværdien med 60000.

Dette resultat er en float, men da det er ønskeligt, at pulsen gemmes som en integer, benyttes .quot, der returnerer kvotienten af en division. Ved at dele resultatet med 1 og tage kvotienten af denne division, fås pulsen gemt som en integer. Pulsen er beregnet, og der returneres til main. Flowet for pulsberegning kan ses på figur 10.15.

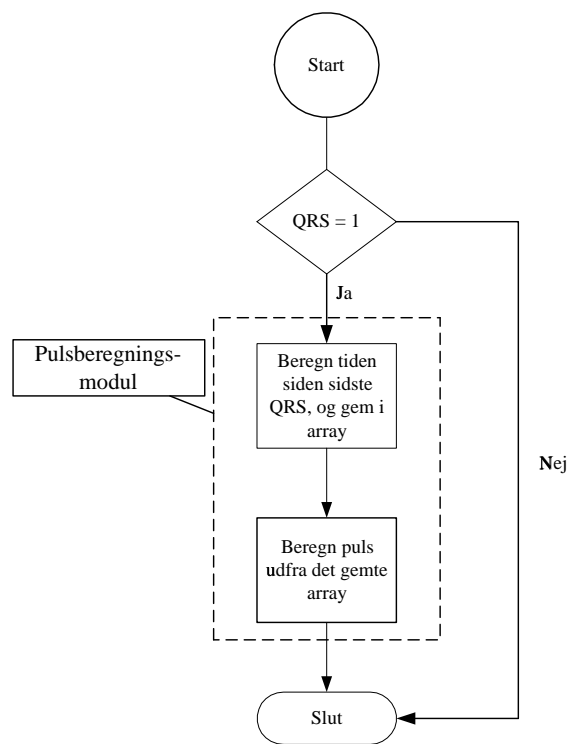
Når man benytter funktionen .quot, risikerer man afrundingsfejl, idet en reel puls på 89,7, vil blive gemt som integeren 89, men det er fra projektgruppens side vurderet at denne afrundingsfejl, i realiteten ikke har indflydelse på produktets funktionalitet.

### 10.5.3 Test af pulsberegningsmodul

For at teste om pulsberegning virkede, blev der i første omgang benyttet et datasæt af kendte EKG-samples, hvorefter en simulering blev foretaget, og pulsen beregnet. Da det returnerede resultat var som forventet, blev beregningsmodulet testet med et dynamisk EKG fra en testperson.

I sidstnævnte tilfælde blev den beregnede puls sammenholdt med målinger fra en eksisterende pulsmåler, Propaq, og pulsen viste sig at være omtrentligt den samme på begge apparater. Desuden blev pulsen målt på den "gammeldags" metode, hvor pulsslæg tælles over 15 eller 30 sekunder, og igen var der overensstemmelse mellem de to beregninger.

Samlet må det derfor konkluderes at pulsberegningen virker tilfredsstillende og som beskrevet i kravspecifikationen.



Figur 10.15: Flow-diagram over pulsberegningsmodulet

# Kapitel 11

## Alarmering

*I dette kapitel designes, implementeres og testes de moduler, der dækker processen Alarmering. Dette er modulerne Alarmdetekteringsmodul og Alarmlyd, hvis hovedfunktion er at detektere arytmier og gøre sygehuspersonalet opmærksom på arytmierne. Modulerne Alarmlyd og Mute designes og implementeres under ét, da disse er meget nært relaterede.*

### 11.1 Alarmdetekteringsmodul

*Alarmdetektering* undersøger, om pulsen ligger i et område, der kræver alarmering, og kravene til dette modul er følgende:

- Asystoli skal detekteres
- Bradykardi skal detekteres, hvis pulsen er under grænseværdien for bradykardialarm
- Tachykardi skal detekteres, hvis pulsen er over grænseværdien for tachykardialarm
- Alarmer skal nulstilles, når alarmkriteriet efter en alarmkrævende situation ikke er opfyldt længere
- Alarmstatus skal returneres til main

#### 11.1.1 Design af Alarmdetekteringsmodul

Alarmdetekteringsmodulet har til opgave at detektere om der er tachykardi, bradykardi eller asystoli. Dette gøres ud fra pulsen og de indtastede grænseværdier.

Da resultatet skal være tilgængeligt for en lang række andre moduler, skal det overvejes, hvordan det, at der er detekteret en alarm, er tilgængeligt for de andre moduler.

#### 11.1.2 Implementation af Alarmdetekteringsmodul

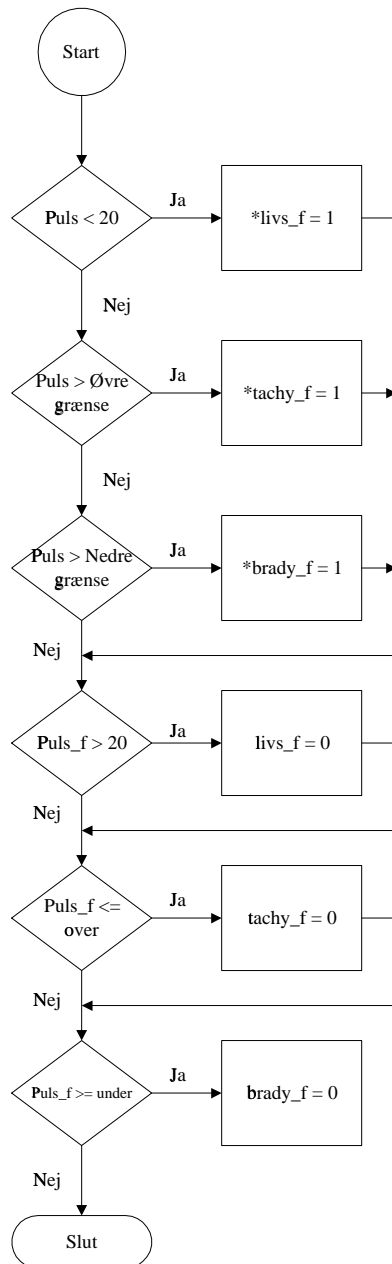
Modulet er implementeret i filen "Alarmeringsmodul.c".

Alarmdetekteringsmodulet er opbygget således, at det først kontrolleres om der er livstruende alarm. Hvis dette er tilfældet, sættes den variabel, som pointeren *livs\_f* peger på, til 1. Der bliver i dette tilfælde ikke undersøgt, hvorvidt der er pulsalarm, da livsalarm har højere prioritet end pulsalarm. Detekteres der derimod ikke en livsalarm, kontrolleres det, om der er tachy- eller bradykardi, og hvis det er tilfældet sættes den variabel, som den respektive pointer peger på, til 1.



For at sikre, at en alarm bliver slukket igen, når pulsen returnerer til normalområdet, bliver dette herefter kontrolleret ved hvert gennemløb.

Det samlede flow i modulet kan ses på figur 11.1



*Figur 11.1: Flow-diagram over Alarmedetekteringsmodulet*

### 11.1.3 Test af Alarmedetekteringsmodul

I de første indledende tests blev pulsen sat lig en konstant, og grænseværdierne for de forskellige arytmier blev på passende måde valgt, så en diode på mc'ens P1.0 skulle lyse, hvis pulsen var uden for grænserne. Da resultatet var som forventet, blev der i en senere test fastsat grænser for arytmier, mens pulsen løbende blev talt op. Modulet blev koblet sammen med Alarmlyd-modulet, så det var muligt at høre, hvilken alarm der eventuelt blev detekteret.

En sidste test blev foretaget ved at lade en af projektgruppens medlemmer cykle på en kondicykel, og ad den vej hæve og sænke pulsen. Grænseværdierne for arytmier blev passende indstillet,

og resultatet var som forventet, hvorfor det må konkluderes at modulet Alarmering opfylder de opstillede krav i kravspecifikationen.

## 11.2 Alarmlyd- og mutemodul

Alarmlyd-modulet sørger for den korrekte akustiske angivelse af alarm; bip ved tachykardi- og bradykardi-alarmer og kontinuerlig lyd ved livstruende alarm. Mutemodulet sørger for afbrydelse af lyden ved tryk på en mute-knap. Disse to moduler er så tæt forbundne, at det gennem design og implementation er blevet klart, at det ikke er hensigtsmæssigt at adskille dem.

Kravene til alarmlydmodulet er følgende:

- Ved asystoli skal alarmeringen være en gennemtrængende hyletone
- Ved pulsalarm skal alarmeringen bestå af tydelige bip
- Lyden skal slukkes, når alarmstatus angiver, at der ikke er alarm længere

Kravene til mutemodulet er:

- Mute skal vare 1 minut
- Trykkes der på mute-knappen, imens mute er aktiveret, skal mute-perioden ikke forlænges.
- Mute-status skal returneres til main, så det kan vises på displayet
- Hvis der opstår asystoli, skal alarmen begynde, selvom mute er aktiveret
- Der skal kunne mute's på forhånd

### 11.2.1 Design af Alarmlyd- og mutemodul

En form for højttaler er nødvendig for at give lyd. Denne kan styres gennem et af benene på mc'en og tændes og slukkes ved pulsalarm og være tændt ved livstruende alarm. Som højttaler vælges en summer.

Status for de tre alarmtyper angives af de 3 alarmstatusvariable, hvis værdier ændres gennem pointere. Det er hensigtsmæssigt ud fra statusvariablene at afgøre, hvordan summer-benet skal tændes/slukkes. Watchdog-timeren (herefter WDT) kan opsættes til at give interrupt med et bestemt interval, så bip kan let implementeres ved at toggle outputbenet til summeren.

En mekanisme til angivelse af hvor længe summeren skal forblive slukket, er nødvendig for at mute kan have den korrekte varighed. En umiddelbar løsning er en timer, der tæller i ét minut og herefter giver et interrupt, der stopper lydundertrykkelsen. Imidlertid findes der ingen timer, der kan tælle i et minut, så det vælges at angive muteperioden som den tid, et *antal* timer-interrupts er om at blive kaldt.

Med disse overvejelser er det naturligt at implementere pulsalarm og mutevarighed i den samme interruptservicerutine, da de to nødvendigvis ikke kan komme til udtryk på samme tid.

For at opnå alarmlyd-funktionaliteten er det nødvendigt at teste, hvilken type alarm der er. Hvis der er tale om en pulsalarm, skal timerinterrupt enables, så summerbenet toggles. Det samme er tilfældet, hvis der er trykket på mute-knappen, men her skal timerinterruptet sørge for lydundertrykkelse og optælling til muteperiodens slutning.

Når der ikke længere er alarmer, skal lyden stoppe og først tænde, næste gang der er alarm. Muteperioden skal stoppe, når minuttet er gået.

## 11.2.2 Implementation af alarmlyd- og mutemodul

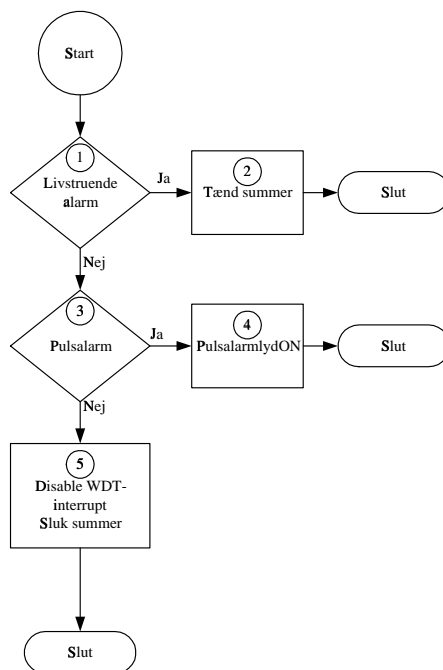
Modulet er implementeret i filen “Alarmlydsmodul.c”.

Det vælges at benytte watchdog-timeren, da det så ikke er nødvendigt at tage højde for at flere processer anvender den samme timer. Interrupt-service-rutinen for WDT skal altså sørge for toggle af summer og opmåling af muteperioden.

For at spare processortid under pulsalarm og mute skal hyppigheden af WDT-interrupt være så lav som muligt. Det er muligt at opsætte WDT til at toggle summeren 4 gange i sekundet, og det vurderes ikke, at disse interrupts under mute er afgørende for systemets ydeevne.

Alarmlyd-funktionaliteten består som vist på fig. 11.2 og fig. 11.3 i hovedtræk af if-sætninger med alarmpointerne som betingelser. På fig. 11.4 er flow i WDT-interrupt-service-rutinen vist. Som eksempel på sammenhængen mellem flowdiagrammer og kildekode er koden vist efter flowdiagrammerne, og indeholder markering af pendant på flowdiagram i kommentarerne.

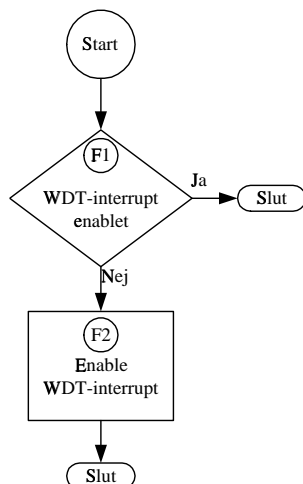
Alarmlyd-funktionaliteten tester, om én af alarmstatuspointerne angiver alarm: hvis livsalarmpointeren peger på en variabel, der er 1, hvilket testes i (1), tændes summeren i (2). Hvis der ikke er livstruende alarm, testes det i (3), om der er en pulsalarm, der kræver bip (dvs. om der er tachykardi-alarm eller bradykardi-alarm). I bekræftende fald kaldes i (4) funktionen pulsalarmlydON.



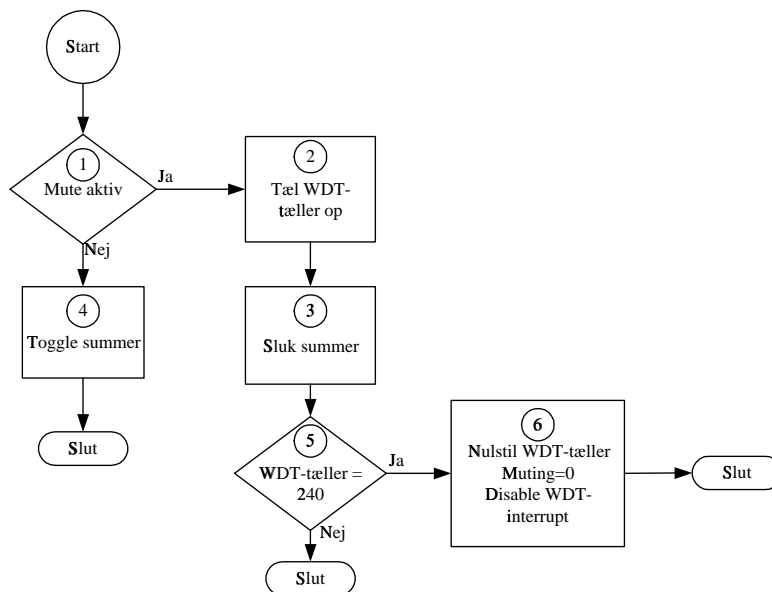
Figur 11.2: Flow-diagram over alarmlydsmodulet

Denne funktion tester i (F1), om WDT-interrupt er enableret, da dette er forudsætningen for, at WDT interrupter og altså toggler summeren (i 4 på fig. 11.4). Hvis WDT-interrupt ikke er enableret, enables det i (F2). Hvis ingen af alarmpointerne peger på en variabel, der er 1, så både (1) og (3) er falsk, disables i (6) WDT-interrupt og summeren slukkes. Herved stoppes lydindikation af alarm, når alarmbetingelsen ophører.

I WDT-interrupt-service-rutinen på figur 11.4 testes det først (1), om der er mutet eller ej. Hvis det er tilfældet, tælles i (2) tællevariablen for antallet af gennemførte interrupts under mute op. Denne er static, så værdien huskes fra kald til kald. I (3) slukkes summeren. Hvis der er gennemført det antal WDT-interrupts, der svarer til 1 minut (5), nulstilles interrupttælleren, mutestatus sættes til 0 og WDT-interrupt disables i (6). Hvis der ikke er mutet, toggles summeren blot i (4).



Figur 11.3: Flow-diagram over funktionen pulsalarmslydON i alarmlydsmodulet



Figur 11.4: Flow-diagram for funktionen WDTinterrupt, der kaldes i interruptservicerutinen for WDT-interrupt

**Kildekode for alarmlyd- og mutemodulet**

I filen "Alarmlydsmodul.c" (tal i parentes referer til fig. 11.2 og 11.3 indtil definitionen af WDT-interruptservicerutinen; herefter refererer tal til fig. 11.4):

```

1 #include "standardheader" //Standardheaderen inkluderes

void pulsalarmlydON(void)
{
5  if ((IE1 & BIT0) == 0) //Hvis WDT-interrupt ikke er aktiveret,... (F1)
    {
        IE1 |= WDTIE ; //...aktiveres WDT-interrupt (F2), der sørger for
            //toggling af summeren
    }
10 }

void alarmlyd_modul(int *bradyalarm_ff, int *tachyalarm_ff,
    *livsalarm_ff)
15 {
    if(*livsalarm_ff==1) //Hvis der er livstruende alarm (1),...
        {
            P5OUT|=BIT7; //...tændes summeren med en konstant tone (2)
        }
20 else
    {
        if((*tachyalarm_ff==1)||(*bradyalarm_ff==1)) //Hvis der er en
            //pulsalarm (3),...
            {
25     pulsalarmlydON(); //...kaldes funktionen til start af pulsalarmlyd (4)
            }
        else
            {
                P5OUT&=~BIT7; //Hvis der ikke er alarm på nuværende tidspunkt
30                 //disables WDT-interrupt og summeren slukkes (5)
                IE1&=~WDTIE;
            }
    }

35 extern int mutestatus;
    interrupt[WDT_VECTOR] void WDT_etter_1/4_sek(void)
        //Interruptservicerutinen for WDT-interrupt
    {
        static int n=0; //Antal gennemførte interrupts

```

```

40     if (mutestatus==1) //Hvis der er mutet (1)...
        {
            n++;          //...inkrementeres antallet af gennemførte interrupts (2)
            P50UT&=~BIT7; //og lydsignalet slukkes/forbliver slukket (3)
45     if (n==240)      //Hvis interruptet er gennemført 240 gange (der er gået
                        //1 minut siden første kald) (5)...
            {
                mutestatus=0; //...ophæves mute,
                IE1&=~WDTIE; //...WDT-interrupt disables,
50     n=0;            //...og antallet af gennemførte interrupts nulstilles (6)
            }
        }
        else           //Hvis der ikke er mutet...
            P50UT^=BIT7; //...toggles summeren (4)
55 }

```

### 11.2.3 Test af alarmlyd- og mutemodul

Modulet er testet efter at der er implementeret en muteknap jvf. kap. 12.

Alarmlyd- og mutemodulet er testet vha. testvektorer for alarmpointerne og mutestatus og med `alarmlyd_modul()` kaldt i en uendelig løkke. En tællevariabel inkrementeredes kontinuerligt i løkken med `mutestatus=0`, og med if-sætninger med tællevariablens værdi som betingelse defineredes forskellige kombinationer af alarmpointerne værdi, så højst én ad gangen pegede på en variabel lig 1. Desuden er dioden anvendt til at angive ændring i alarmstatus, så den skiftevis tændte og slukkede, når en ny betingelse for tællevariablen var opfyldt. Herved opnåedes skift i alarmstatus, hvilket kunne følges vha. dioden, og det kunne observeres, om ændring i alarmstatus afspejledes korrekt i ændring af lydindikationen.

Det er vha. testvektorer for alarmpointerne undersøgt, at alle typer af alarm kan mutes. Det er desuden testet, at alarmstatuspointerne værdi blev korrekt opdateret under mute: en test-tællevariabel blev talt op ved hvert WDT-interrupt under mute, og i main blev der ændret på alarmstatuspointerne værdi.

Testene viste, at alarmlyd- og mutemodulet fungerer som forventet og at kravspecifikationen altså er opfyldt.



# Kapitel 12

## Brugergrænseflade

I dette kapitel designes, implementeres og testes de moduler der dækker processen brugergrænseflade. Processen var oprindeligt tænkt opdelt i modulerne opsætning af tekstdisplay, opdater tekstdisplay og knapopsætning. Sidstnævnte modul integreres her i opsætning af tekstdisplay således, at der kun designes to moduler.

Modulet “opsætning af tekstdisplay” varetager funktioner som opsætning af diverse porte, der benyttes på mc'en, interrupterklæringer og initialisering af displayet.

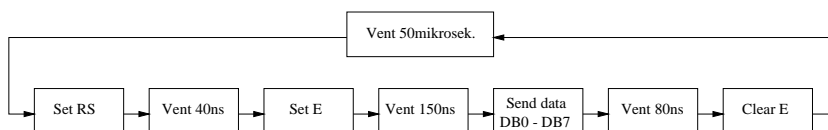
Det andet modul varetager opdatering af displayet med puls, grænser og alarmer.

Hvis displayet skal opdateres skal den anvendte processortid minimeres. Der er 64 tegn på displayet og en kontinuelig udskrivning af alle tegn varer ca. 3 ms. Det er ikke hensigtsmæssigt at stoppe alle andre processer i 3 ms under udskrivning, og derfor skal der vælges en løsning, der sikrer, at dette ikke er nødvendigt. Den første sektion, optimering af udskrivning på display, behandler en løsningsstrategi for dette. Inden gennemgang af den følgende sektion kan appendiks F med fordel læses. Dette behandler displayets benkonfiguration og funktion.

### 12.1 Optimering af udskrivning på displayet

Der er et meget strengt timingdiagram som skal overholdes for at displayet viser korrekt data og for at data som eksempelvis *busyflaget* (BF) kan læses fra displayet. Displayet er langsomt til at opdatere karakterer på skærmen, og visse instruktioner som “display clear” kan tage helt op til 1,64 ms. Enkeltkarakterer kan opdateres med 50  $\mu$ s mellemrum. Mc'en kan skrive til displayet langt hurtigere, end det kan modtage data, og derfor læses BF i mange applikationer med henblik på at generere et interrupt når displayet er klar til at modtage ny data. På denne måde sikres der, at data ikke går tabt fordi displayet ikke nåede at processere data, før der igen sendes en kommando og dataoverførsel bliver mere stabil.

Timingen for en typisk skriveoperation ses på fig. 12.1.



**Figur 12.1:** Displayets timingdiagram skal overholdes, ellers udskrives det ønskede ikke korrekt

På figuren ses det tydeligt, at den væsentligste faktor, der gør, at displayet er langsomt til at processere data, er den delaytid, der kræves til fortolkning af latched data.

Det var det gruppens ønske at udnytte BF funktionaliteten for at optimere stabilitet under skrivning til displayet. Tanken var at udnytte tiden mellem interrupts fra displayet til at udføre andre

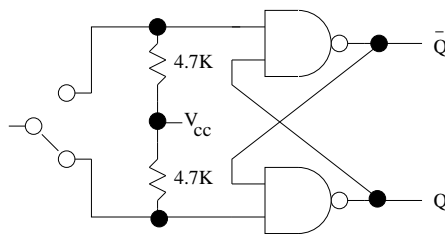


opgaver og sikre, at data blev modtaget korrekt af displayet. Imidlertid krævede dette en hardwaremæssig sikkerhedsforanstaltning, idet mc'en forsynes med 3,3 V og displayet forsynes med 5 V. Busyflaget kan derfor ikke direkte læses fra displayet uden at mc'en tager skade. En idé som blev testet for at undgå denne inkompatibilitet var at indsætte en line driver i kommunikationsleddet mellem enhederne. En passende line-driver som kunne forsynes med 3,3 V og acceptere 5 V- inputs blev konfigureret til at styre datastrøm begge veje mellem display og mc'en og virke som buffer. Indledende resultater var utilfredsstillende. Line-driveren sikrede godt nok mc'en, men den stabilitet, som blev søgt, var til gengæld mistet under switchning af linedriveren til at modtage data fra en anden retning. Der var en forholdsvis stor misvisning af data.

Af den grund blev en anden idé testet. I stedet for skifte retning for line-driveren under læsning fra displayet, blev det udnyttet at line-driveren var en tristate type. Det blev forsøgt, under læsning fra displayet, at sætte line-driveren i en højimpedans tilstand på display siden og don't care tilstand på mc-siden. BF blev så ledt udenom linedriveren og igennem simpel logik, der fortolkede om der blev læst fra displayet (R/W høj) og om BF'et var sat. De logiske kredse blev forsynet med 3,3 V og hensigten var at output fra disse skulle generere interrupts.

Resultatet af dette forsøg havde heller ikke den ønskede stabilitet som udfald. Igen var der misvisning af data og uforventede 4,5 V spikes på mc'ens ben var resultatet efter de logiske kredse. Endeligt blev det besluttet ikke at læse busyflaget, og en tredje -men ligeså brugbar løsning- blev fulgt. De krævede delaytider kan i stedet styres af mc'ens timer således at udskrivning er timerinterruptstyret. Dette minimerer den tid som displayet optager mc'en og interfererer derfor minimalt med andre processer.

En anden hardwaremæssig nødvendig foranstaltning var nødvendig for at sikre et korrekt data-output på displayet. Idet knapper skal implementeres for at kunne styre rundt mellem menuer og indstille grænser m.m, skal prel fra knappernes switch minimeres. Til det formål er der implementeret en antiprel-konfiguration mellem knap og input til mc'ens ben. Denne er illustreret på fig. 12.2.



Figur 12.2: Figuren viser den anvendte antiprel-konfiguration

Der benyttes i alt 5 knapper for at opnå de ønskede funktionaliteter og det er nødvendigt at anvende interruptporte på mikrokontrolleren til disse.

## 12.2 Modul: opsætning af tekstdisplay

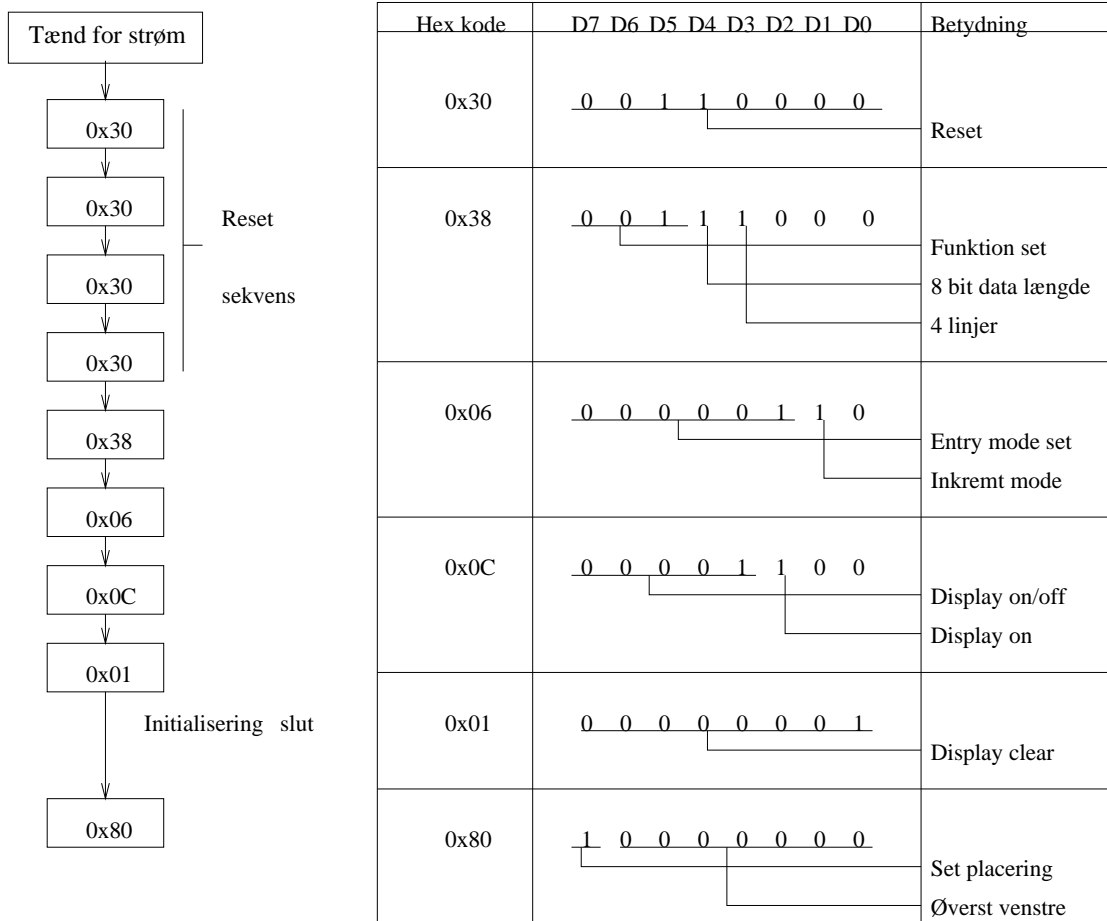
### 12.2.1 Design af opsætnings-modul

Hver gang displayet tændes, skal det initialiseres. Initialiseringen kan være forskellig alt efter behov. Der er en intern initialiseringsprocedure, som sætter displayet i en default mode, men denne er ikke i overensstemmelse med den ønskede anvendelse, så derfor benyttes en brugerdefineret initialisering.

### 12.2.2 Implementation af opsætnings-modul

Modulet er implementeret i filen "Opsætning af tekstdisplay.c".

Den benyttede brugerdefinerede initialiseringsprocedure på fig. 12.3 tænder skærmen, sikrer at 4 linier benyttes (2 linier er standard), slukker cursoren, benytter autoinkrementer mode, clearer displayet og går til startadressen i øverste venstre hjørne af skærmen. Initialiseringsmodul kaldes en gang fra hovedprogrammets main() under betingelsen RS=0 og R/W=0. Delays af forskellig varighed er krævet mellem hver sendt hex kode, og disse er implementeret med forløkker. Dette er muligt under opstart af det samlede system, inden andre moduler kræver processortid.



**Figur 12.3:** LCD initialiseringsproceduren. Først resettes den interne initialisering. Derefter klargøres til 8-bit datapakker og 4-liniers skærmtekst. Displayet sættes til at flytte cursoren en plads efter karakterudskrivning og displayets pixels tændes. Displayet ryddes, og cursoren placeres i øverste højre hjørne, klar til udskrivning.

Der sendes først 30(hex) fire gange for at resette den interne initialisering. Herefter følger datapakker, således at den ønskede funktionalitet opnås.

### 12.2.3 Test af opsætnings-modul

Initialiserings-proceduren er testet med mindre programmer, der placerede tekst på udvalgte steder på skærmen. Alle 4 linier kan benyttes, og der inkrementeres automatisk for hver karakter der udskrives. Når displayet tændes, clears skærmen som forventet. Initialisering-proceduren virker som den skal.

## 12.3 Modul: Opdatering af tekstdisplay

Kravene til dette modul er (med kravene til knapopsætning inkluderet til sidst):

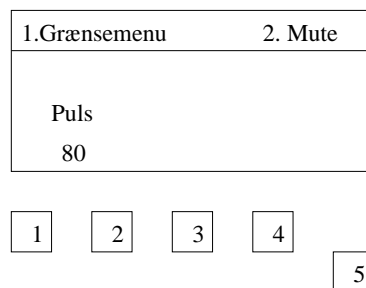
- Der skal kunne vises skærbillederne: hovedmenu med pulsangivelse, grænsemenu og alarmskærme
- Alle skærbilleder skal kunne nås fra hovedmenuen
- Grænsen for bradykardi må ikke kunne indstilles til at være højere end grænsen for takykardi og omvendt
- Bradykardigrænsen default: 40
- Takykardigrænsen default: 100
- Grænserne skal returneres til main
- Skærbillederne for alarmer skal være inaktive i grænsemenuen, så grænseindstilling uden afbrydelse er mulig
- Knaptryk skal kunne forårsage: Nyt skærbillede, hævning eller sænkning af alarmgrænser eller muting
- Hver knap kan have forskellig betydning alt efter det aktuelle skærbillede

### 12.3.1 Design af opdaterings modul

Modulet er implementeret i filen “Opdatér tekstdisplay.c”.

Displayets tegnsæt er ikke ASCII baserede hextal, så det er nødvendigt med en konverteringsalgoritme så korrekte karakterer udskrives. En sådan konvertering tager tid, så der vælges kun at benytte konverteringsalgoritmen til mindre udskrivningsprocedurer som udskrivning af grænser og til konvertering af pulsen som er en nødvendighed.

Der designes 5 menuer på displayet. Hovedmenuen er vist på fig. 12.4



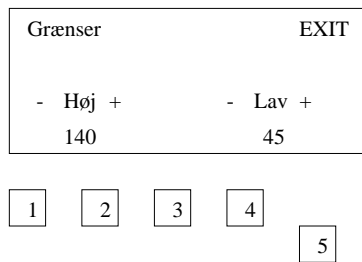
*Figur 12.4: Hovedmenuen*

Her vises pulsen, der er mulighed for at mute en alarmlyd i et fastsat tidsrum og der er adgang til Menu 2. Menu 2 er illustreret på fig. 12.5

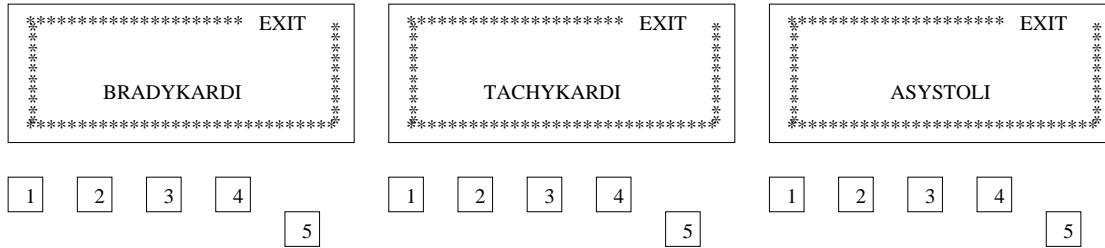
Menu 2 er en grænsemenu, hvor indstilling af pulsgrænser kan foretages. Menu 3-5 på fig. 12.6 er alarmmenuer.

Disse visualiserer en patienttilstand baseret på pulsen. Det skal være muligt at stoppe en alarmskærm og bladre i menuen under aktiv alarm. Denne funktion tillader, at man har mulighed for at ændre grænser, hvis disse var forkert indstillede. Knapper skal have forskellige funktioner afhængigt af, hvilket skærbillede der er aktivt, og betjeningen heraf skal være naturlig.

Knappernes betydning skal styres af to overordnede variable. En variabel, der holder rede på hvilken knap der aktiveres og en anden variabel, der holder rede på i hvilken menu man befinder



Figur 12.5: Grænsemenuen



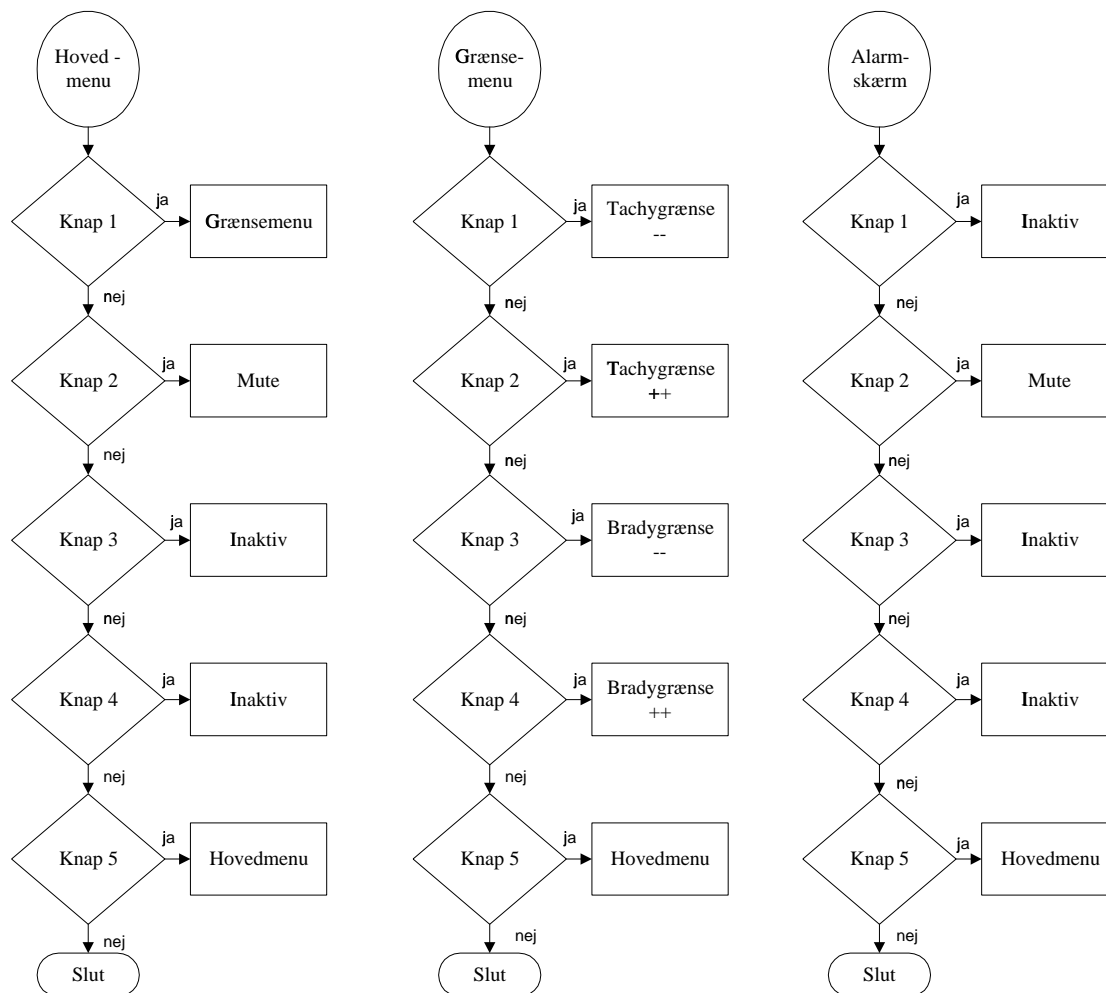
Figur 12.6: Alarmmenuer

sig. Tilsammen skal disse to variable sikre, at de korrekte karakterer udskrives.

Sammenhængen mellem menuerne og knapfunktioner fig. 12.7, illustrerer grafisk de forskellige knappers funktion og kan lette gennemgangen af flowdiagrammerne.

Der er på fig. 12.7 angivet, hvilken funktion hver af de 5 knapper har i forhold til hvilken menu, der er aktiv. Fra hovedmenuen er der adgang til grænsemenuen med knap 1, og det er muligt at mute en lydalarm med knap 2. I grænsemenuen kan der indstilles en øvre og nedre tilladelig pulsgrænse. Dette gøres med knapperne 1-4. Grænsemenuen kan ikke undertrykkes af en visuel alarm. Dette sikrer, at man ikke bliver afbrudt under indstilling af grænser. Knap 5 aktiverer hovedmenuen fra grænsemenuen.

Når pulsen overskrider de indstillede grænser, vises en af tre alarmskærme. Disse er tachykardi, bradykardi og asystoli. Hvis en alarmskærm er aktiv, deaktiveres alle knapper med undtagelse af knap 2, der giver brugeren mulighed for at mute og knap 5, som giver brugeren den netop omtalte manøvreringsmulighed. Knap 5 deaktiverer kun visuelle alarmer, så der er nødvendigt efterfølgende at mute, hvis dette ønskes. I den følgende implementation gennemgås flowdiagrammerne for koden, som styrer menusystemet. Først beskrives knap 5, som er tilgængelig fra alle menuer og udskrives hovedmenuen, derefter gennemgås de andre knapper, som er tilgængelige, når hovedmenuen er vist. Efterfølgende beskrives grænsemenuens knapper, og der afsluttes med at gennemgå, hvordan en alarmskærm aktiveres. Kun én alarmskærm gennemgås, da de to andre stort set har samme flowdiagram.

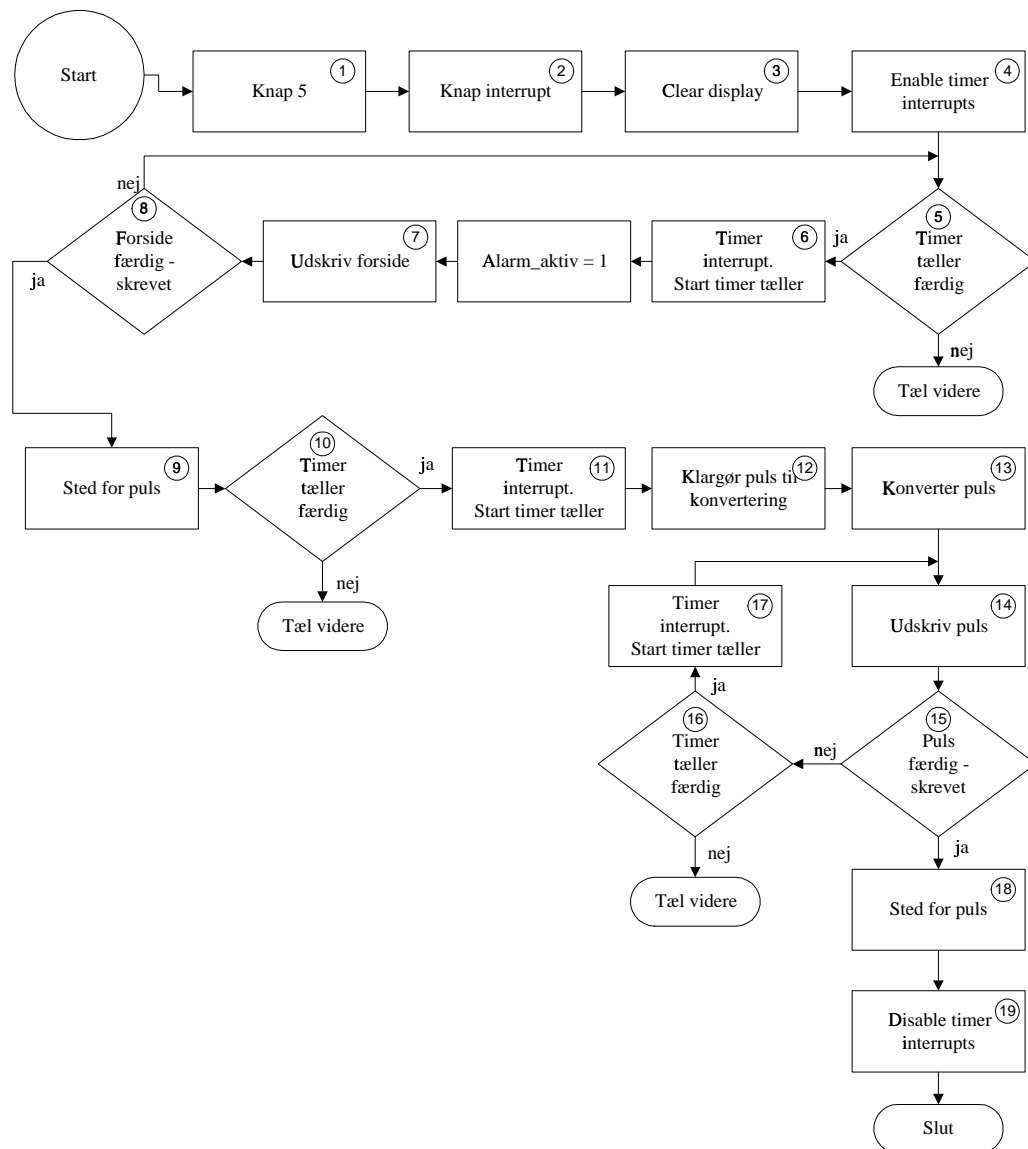


**Figur 12.7:** Figuren viser sammenhængen mellem aktiv side og knapfunktion. Der er ialt implementeret 25 forskellige knapfunktioner. Her er medtaget blokering af knapper, så de er inaktive

### 12.3.2 Implementation af visning af hovedmenu

Når knap 5 trykkes, vises hovedmenuen uanset hvilken menu, der er aktiv. Knappen genererer et interrupt som det ses af fig. 12.8, (1). I interruptservicerutinen switches på det aktive flag i forhold til hvilken knap, der er trykket på. Dette er ikke vist på flowdiagrammet, da kun knap 5 diskuteres. I (3) cleares displayet. Dette er nødvendigt, fordi funktionen sætter startadressen til øverste venstre hjørne, så der kan skrives herfra. Kaldes denne funktion ikke, påbegyndes udskrivning af hovedmenuen fra det, sted hvor den foregående side blev afsluttet. Det aktive interruptflag lægges ned, så der igen kan switches på det ved næste interrupt. Inden interruptet afsluttes, aktiveres timer A på microcontrolleren, (4). Denne benyttes til at generere repetitive interrupts, så der kan udskrives en karakter ad gangen som ønsket, (7). Hex værdier som displayet genkender som karaktererne for hovedmenuen er initialiseret i et integer array. Det sidste tal i dette array er et 0, og dette afgør, at der ikke er mere tekst at udskrive i hovedmenuen. Når hovedmenuteksten (8) er udskrevet, benyttes en karakter-placeringsfunktion, (9) således at der springes til det sted, hvor pulsen skal udskrives. Det er nødvendigt at isolere hvert ciffer i pulsen for at konvertere værdien for pulsen til en hexværdi, som displayet skal modtage for at vise pulsen. Dette klares i (12), hvor hvert ciffer for pulsværdien lægges i et array, som kan klare en puls op til 999. Pulsværdien er en global variabel, som er modtaget fra pulsberegningsmodulet. I (13) konverteres disse værdier og udskrivning af pulsen påbegyndes i (14). Der er taget hensyn til, at der ikke udskrives nuller foran pulsværdier under 100. Som afslutning på udskrivning af

hovedmenuen gøres der klar til at skrive en ny puls på det rigtige sted hvis pulsen ændres (18). Siden er færdigskrevet, og der er ikke brug for flere timer interrupts. Derfor disables disse i (19)



**Figur 12.8:** Flowdiagrammet illustrerer funktionskald hvis hovedmenuknappen trykkes. For yderligere forklaring, se afsnit implementering af visning af hovedmenu

### 12.3.3 Test af visning af hovedmenuen

Hovedmenuen blev designet som den første udskrivnings-skærm og er blevet testet isoleret og efter andre skærbilleder var designet. Det er blevet testet om tryk på knap 5 clearer skærmen og udskriver hovedmenuen, uanset aktivt skærbillede. Dette er tilfældet hver gang. Alarm-skærbilleder blev genereret ved at sætte pulsvariablen både over og under undstillede grænser. Det blev testet at udskrivningen af hovedmenuen ikke gentages når den er færdig-udskrevet. Udskrivningen stopper som forventet. For at teste om nuller forekom for pulsværdier under 100 eller under 10, blev knapper opsat til at hæve og sænke variabelen puls til forskellige værdier. Nye pulsværdier udskrives på det korrekte sted og uden nuller foran når pulsen var under 100 og under 10. Konverteringen af pulsværdien var rigtig. Udskrivning af hovedmenuen fungerer, som den skal.

### 12.3.4 Implementation af visning af grænsemenuen

Det eneste sted hvorfra grænsemenuen er tilgængelig er fra forsiden. Flowdiagrammet for udskrivningen af grænsemenuen er illustreret på fig. 12.9

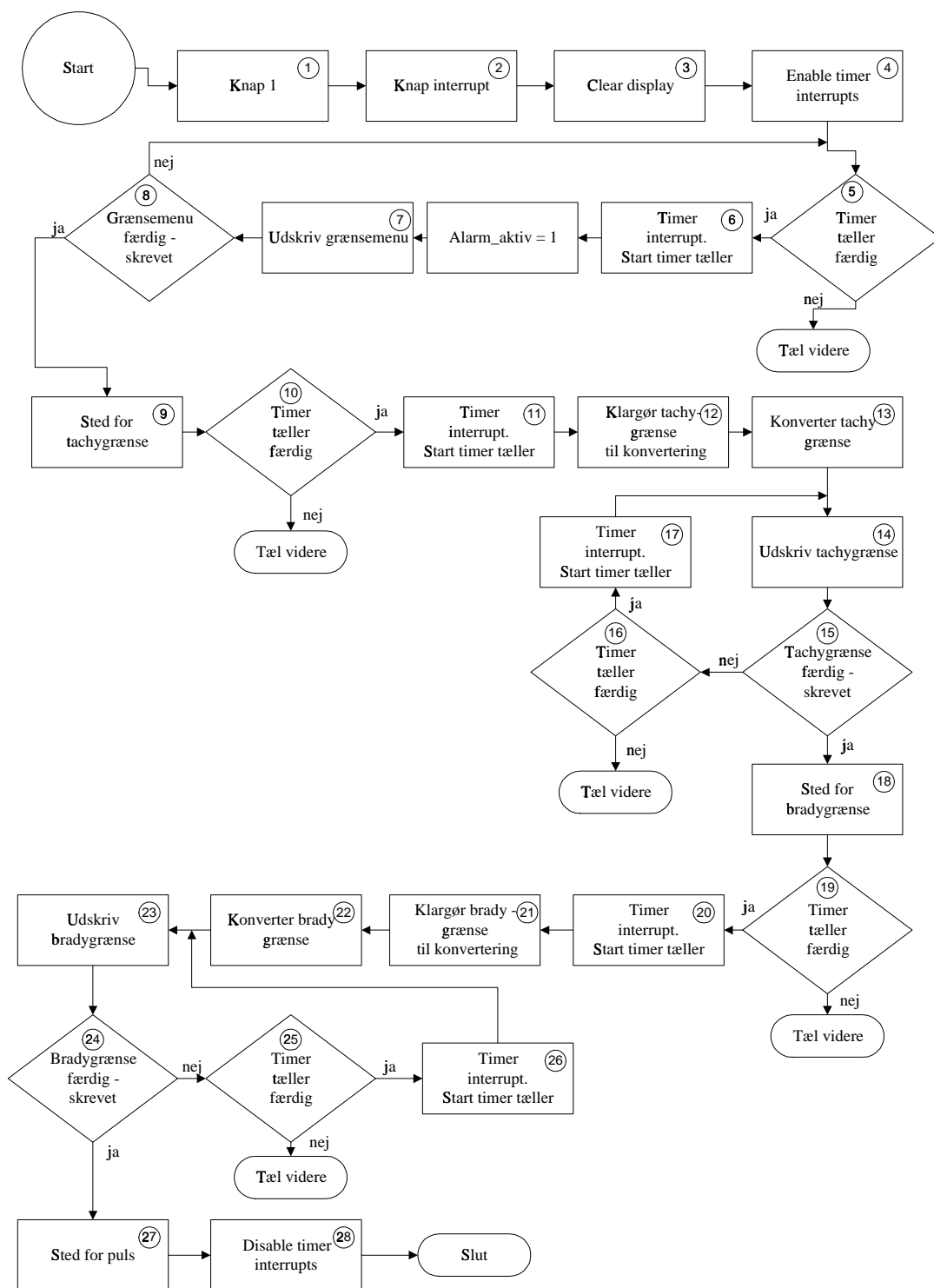
Fra (1-8) følges samme princip som udskrivning af forsiden. I (7) genbruges udskrivningsfunktionen, som blev brugt til at udskrive hovedmenuen. Der switches i denne funktion på en variabel, som blev sat i interruptservicerutinen, da det blev identificeret, at knap 1 var trykket. Den nedre grænse for pulsen placeres samme sted som pulsen udskrives, (9). På denne måde genbruges endnu en funktion. Der skal udskrives to grænseindstillinger, en nedre og en øvre. På samme måde som pulsen skal konverteres til display hex-kode, sker dette også for grænseværdierne, (12,13) og (21,22). Der er lavet et samlet array for grænserne, og den enkelte grænse identificeres ved kald af den korrekte plads i arrayet. Den nedre grænse har sin egen placeringsfunktion, (18).

Afslutning af grænsemenu udskrivning forløber identisk med udskrivning af hovedmenuen.

### 12.3.5 Test af visning af grænsemenuen

Udskrivning af grænsemenuen forløber næsten identisk med udskrivning af hovedmenuen og er blevet testet på lignende måde. Grænsemenuen udskrives kun, hvis forsiden er aktiv og udskrives ikke ellers. Dette er testet ved at trykke på knap 2 fra alarmskærm-billederne. Det er testet, om skærm-billeder for alarmer kan undertrykke grænsemenuen. Der blev induceret alarmtilstande med knapper der var sat op til at ændre pulsværdien. I ingen tilfælde blev grænsemenuen undertrykt af en alarmskærm.

Default værdierne for bradygrænsen og tachygrænsen placeres rigtigt og er korrekt konverteret. Udskrivning af grænsemenuen forløber, som den skal.

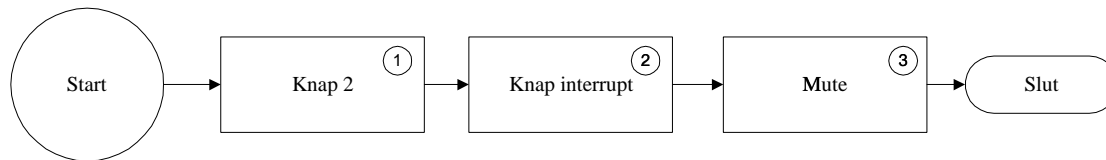


**Figur 12.9:** Når knap 1 trykkes fra hovedmenuen, udskrives grænsemenuen. Til dette formål genbruges flere funktioner som anvendes til at udskrive hovedmenuen med. Dette letter kodningsprocessen.



### 12.3.6 Implementation af Muting

Knap 2 skal trykkes, hvis der ønskes at mute. Fig. 12.10 viser det simple flowdiagram for denne funktion. Trykkes knap 2, genereres et knapinduceret interrupt. I interruptet switches på flaget for interruptet, WDT startes og mutestatus sættes til 1. Displayet viser ikke om der er mutet, men dette kan implementeres ved en mindre udbygning af koden for displayet.



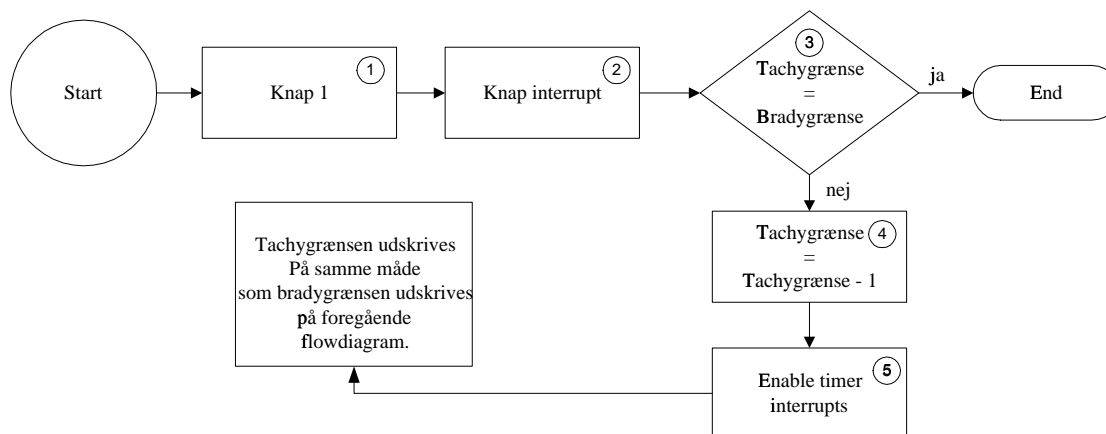
**Figur 12.10:** Når knap 2 trykkes, mutes lydalarmer i en fastindstillet tid. Det visualiseres ikke, om mute er aktiv, men der er vist, hvilken knap der skal trykkes på for at mute

### 12.3.7 Test af muting

Det er blevet testet at WDT-interrupt enables og mutestatus sættes til 1 ved tryk på knap 2. Dette er tilfældet, og der mutes korrekt. Testen af mute-modulet er beskrevet i det foregående kapitel. Det er muligt at mute fra alle skærbilleder undtagen fra grænsemenuen. Denne funktion virker som den skal.

### 12.3.8 Implementation af grænseindstilling

Grænseværdierne bestemmer, hvilket interval man tillader pulsen at ligge i. Hvis den aktive side er grænsemenuen sættes 2 variable ved tryk på knapperne 1-4. Disse variable bruges af alarmeringsmodulen til at sammenligne med pulsen og herudfra bestemme, om der skal være en visuel alarm. På fig. 12.11 er flowdiagrammet for indstilling af den øvre grænseværdi til en lavere værdi vist. Princippet er det samme for indstilling til en højere værdi og for indstilling af den nedre grænse. I (3) laves en begrænsning af indstillingen for høj grænseværdi i forhold til den lave grænseværdi. Lignende begrænsninger er implementeret for indstilling af de andre grænseværdier.



**Figur 12.11:** I grænsemenuen kan der indstilles værdier for høj og lav tilladelig puls. Overskrides disse, visualiseres det med en passende alarmskærm

### 12.3.9 Test af grænseindstilling

I grænsemenuen blev det testet, om knapperne 1-4 ændrer den viste grænseindstilling for bradykardi og tachykardi. Dette var tilfældet. Ligeledes blev det bekræftet, at de variable som benyttes af alarmerings-modulet, antog korrekte værdier i forhold til knaptryk. Der kan ikke indstilles grænseværdier under 0, tachykardigrænsen kan ikke indstilles lavere end bradykardigrænsen og omvendt. Der forekommer ikke nuller foran grænseværdierne, hvis disse er under 100 eller under 10. Med grænserne indstillet med forskellige værdier er det testet, at der kommer alarmskærme, hvis pulsen er under grænseværdierne, og den aktive side er hovedmenuen. Grænseindstilling virker efter hensigten.

### 12.3.10 Implementation af aktivering af alarmskærm

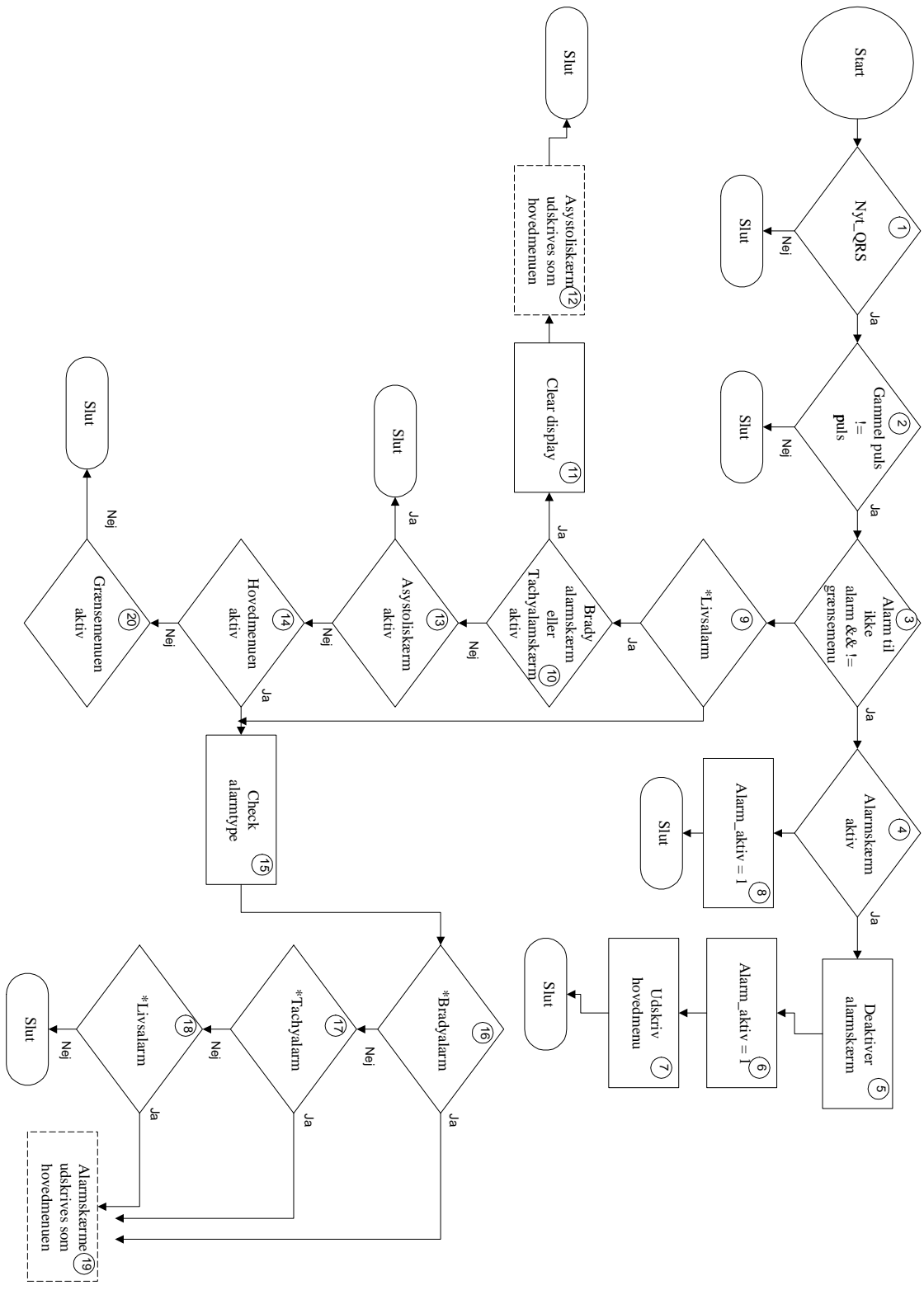
På fig. 12.12 illustreres, hvorledes alarmskærme aktiveres og opdateres. Som betingelser for at alarmskærme kan udskrives eller cleares, skal der være detekteret et QRS-kompleks, (1) og pulsen skal have ændret sig, (2). Det første der undersøges for i (3) er, om en alarmskærm skal cleares eller udskrives. Er der ikke længere alarm, (3) og er en alarmskærm aktiv, (4) skal alarmskærmen cleares, det skal være muligt at kunne vise evt. nye alarmer når hovedmenuen er udskrevet, (6) og hovedmenuen skal udskrives, (7).

Negeres (3) er der 5 muligheder. En af de tre alarmskærme er aktive, hovedmenuen er aktiv eller grænsemenuen er aktiv. I tilfælde af asystoli, (9) og en alarmskærm for bradykardi eller tachykardi var aktiv, (10) skal displayet cleares og alarmskærmen for asystoli udskrives, (12). Hvis der er asystoli, (9) og denne skærm allerede var aktiv, (13) sker der ingenting. I tilfælde af asystoli hvor hovedmenuen er aktiv, (14) undersøges alarmtypen, (15-18) og asystoliskærmen udskrives, (19).

Er der ikke asystoli, (9) skal alarmskærmen bradykardi eller tachykardi udskrives. Dette sker i (15-19).

### 12.3.11 Test af aktivering af alarmskærm

Aktivering af alarmskærme sker korrekt, hvis den aktive menu ikke er grænsemenuen. Dette er testet med knapinducerede pulsændringer.



Figur 12.12: Alarm flow

# Kapitel 13

## Samling af moduler

*I dette kapitel beskrives det, hvordan de implementerede moduler samles til et helt system, hvor alle moduler virker sammen.*

### 13.1 Main

Main er implementeret i filen “main.c”

Alle moduler er implementeret hver for sig som beskrevet, så hvert modul efter implementation er en selvstændig enhed, der opfylder kravspecifikationen.

Projektgruppen vælger at linke modulerne, da denne løsning anbefales i SPU-bogen og klart er lettest at overskue, når der er produceret meget kode - især hvis koden er udviklet i moduler. Herved kan funktioner kaldes fra en main-funktion i en fil, men findes i en anden fil.

Linkningsprocessen består efter oversættelsen i at gøre de forskellige moduler i stand til at kalde funktioner defineret i andre moduler [16].

#### 13.1.1 Struktur af headerfiler

I linkningsprocessen er headerfiler (.h-filer), der indeholder forskellige inkluderinger af andre headerfiler, definitioner samt for-erklæringer af funktioner nødvendige.

De filer, hvori en bestemt headerfil inkluderes, får så kendskab til funktioner erklæret i en anden fil, hvis headerfilen indeholder for-erklæring af prototyperne. Dermed benyttes headerfiler til at lade filer/moduler dele funktioner.

For main er der derfor lavet en headerfil, mainheader.h, der inkluderer standardheaderen og for-erklæringer af prototyper af alle moduler/funktioner, der kaldes fra main.

Der er lavet en headerfil, pulsheader.h, der gør differentiationsmodulet tilgængeligt for både kalibrerings- og QRS-detekteringsmodulerne. Headerfilen LCDheader.h gør diverse display-funktioner tilgængelige fra både “Opsætning af tekst-display” og “Opdatering af tekstdisplay”.

Uanset hvilke funktioner, der skal være kendte filer/moduler imellem, er det nødvendigt i alle filer at inkludere “MSP430x14x.h”, der associerer sigende navne for registre og bits i mc’en til de hexadecimale adresser, mc’en kan forstå. Dette gøres i en standardheader, “standardheader.h”, som desuden inkluderes i headerne for main og pulsregningen. Standardheader.h inkluderes i alle moduler, der ikke har enten mainheader.h eller pulsheader.h inkluderet. I Pulsregningsmodul.c er det desuden en nødvendighed at benytte standardbiblioteket stdlib.h, så funktionen .qout kan anvendes.

### 13.1.2 Variable

Alle variable, der skal udveksles mellem moduler jvf. fig. 6.2, er erklæret i begyndelsen af main.c. Nogle af værdierne initialiseres til en værdi, der ikke er 0; fx grænser for tachy- og bradykardi, der har en standardværdi, indtil de ændres vha. knapperne. Udveksling af disse variable foregår gennem kald af modulerne, hvor variablene bruges som aktuelle parametre.

I interrupts er det imidlertid ikke muligt at udveksle variable på denne måde; interruptservice-rutiner kan ikke tage eller returnere parametre. I filerne, hvor interruptservicerutiner er erklæret, er de variable, interruptservicerutinen skal manipulere, derfor erklæret eksterne; fx er variabelen mutestatus erklæret eksternt i filen Alarmlydsmodul.c, da WDT-interruptservicerutinen er erklæret her, og den skal manipulere mutestatus.

### 13.1.3 Struktur i main

Main indeholder forskellige strukturer til flow-kontrol i forbindelse med kald af modulerne. Tal i parentes henviser til fig. 13.1.

Først foregår al initialisering af mc og display, hvor mc'en opsættes til at konvertere, bruge clocks, timere, UART og porte korrekt. Displayet initialiseres, og kalibreringsskærmen anmodes udskrevet. Som afslutning på initialiseringen enables General Interrupts, så fx konvertering begynder. Initialiseringsmoduler og -funktioner kaldes kun én gang (1).

Herefter kalibreres (2), så der findes grænser til brug i QRS-detekteringen. Kalibreringen kører kun afbrudt af interrupts, indtil der er kalibreret færdigt. Under kalibreringen er dioden tændt.

Når kalibreringen er afsluttet, slukkes lysdioden, hovedmenuen udskrives, og hovedløkken kører, indtil mc'en slukkes. Hovedløkken gennemføres kun, når der er konverteret nye data (3), så der ikke foretages beregninger på samme input flere gange.

I hovedløkken kaldes QRS-detekteringsmodul (4).

Hvis der er gået 3 sekunder siden detektering af sidste QRS-kompleks, hvilket testes i (5), sættes pulsen til 0 (6).

Hvis QRS-detekteringsmodul markerer, at der er detekteret et nyt QRS-kompleks(7), kaldes pulsbergningsmodul og dioden tændes som indikation af et pulsslag (8).

Hvis pulsen er ændret siden sidste pulsbergningsmodul (9), kaldes alarmeringsmodul, da der så kan være ændret alarmstatus. Desuden kaldes den funktion, der tester, om der skal udskrives en alarmskærm, samt den funktion, der udskriver den nye puls og eventuelle alarmskærme (10).

Hvis der netop er opstået og detekteret asystoli (11), deaktiveres WDT-interrupt og mutestatus sættes til 0 (12), således at mute ophører, hvis det var mutet før asystoli.

Hvis der ikke er mutet (13), kaldes alarmlydmodul (14). I princippet er det tilstrækkeligt at kalde alarmlydmodul sammen med alarmeringsmodul, da der kun med ændret alarmstatus skal ændres lydangivelse. Med det valgte design er det imidlertid nødvendigt at betinge kald af alarmlydmodul af, at der ikke er mutet – ellers forbliver der ikke mutet ved livstruende alarm indtil mute-perioden er overstået.

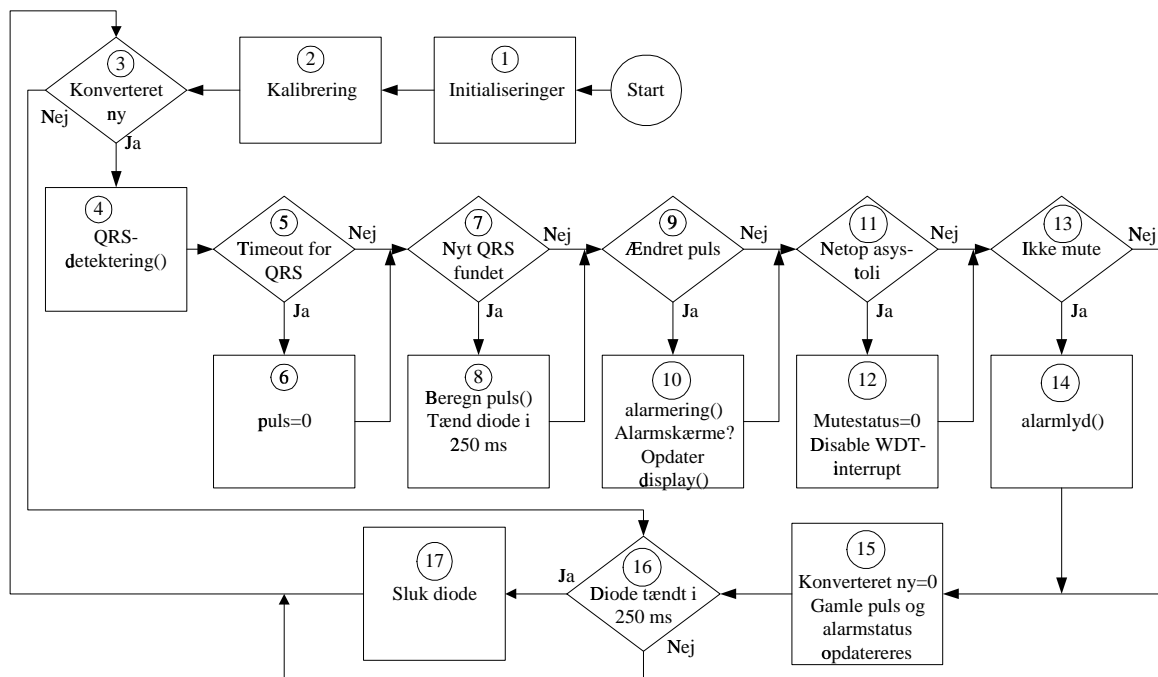
Dette er enden på de egentlige kald af moduler i hovedløkken, så den variabel, der angiver, at der er konverteret en ny værdi og at hovedløkken derved skal køre, nulstilles.

Desuden opdateres værdierne for gammel puls og gamle alarmstatus (15), således at nye værdier beregnet i ét gennemløb bruges som gamle værdier i det næste gennemløb.

Endelig slukkes dioden (17), hvis den har været tændt i mere end 250 ms (16), så den med blink af godt 250 ms varighed angiver pulsen.

### 13.1.4 Test af main

Da main samler alle moduler, er implementationen i realiteten en integrationstest modulerne imellem. Først er modulerne i hver proces integreret, således at de virkede sammen. Herefter er



Figur 13.1: Figuren viser flow-diagram for main-funktionen

de forskellige processers moduler ét for et sat sammen, og for hvert er det testet, at input/output var korrekt. Først når et nyligt integreret modul virkede sammen med resten, er et nyt modul integreret. Da alle moduler var integreret, blev koden optimeret. Test af den endelige, optimerede kode er identisk med funktionalitetstest.



## Kapitel 14

# Design, implementation og test af brugerflade på PC'en

### 14.1 Design

#### 14.1.1 Hvad er LabVIEW?

LabVIEW er et multitrådet, grafisk programmeringssprog, der tillader simpel udvikling af grafiske brugerflader til data-opsamling og -manipulation. Programmer lavet i LabVIEW benævnes VIs (Virtual Instruments), og underprogrammer benævnes subVIs. I dette projekt bruges LabVIEW til at implementere den grafiske brugergrænseflade på pc'en, der giver brugeren mulighed for at betragte real-time EKG-signaler samt analysere minut-gamle optagelser af patientens EKG. Der skal altså designes såvel et delsystem til opsamling, lagring og visning af de serielt overførte data fra mc'en som et delsystem til genindlæsning og visning af de lagrede data.

#### 14.1.2 Brugerflade og programdesign

For at udnytte LabVIEWs multitrådede opbygning bedst muligt, må designet af et LabVIEW-program tænkes som en sammensætning af parallelle processer - beregningsopgaver, der kan foregå pseudo-parallelt. Beregningsrækkefølgen er data-flow-styret, og det er derfor væsentligt anderledes at designe et LabVIEW-program end et program i et imperativt sprog som C. Det foreliggende programdesign tager sit udgangspunkt i ovenstående beskrivelse af de to ønskede processer: "Opsamling, lagring, visning" og "Genindlæsning, visning".

#### Obygning af brugerfladen

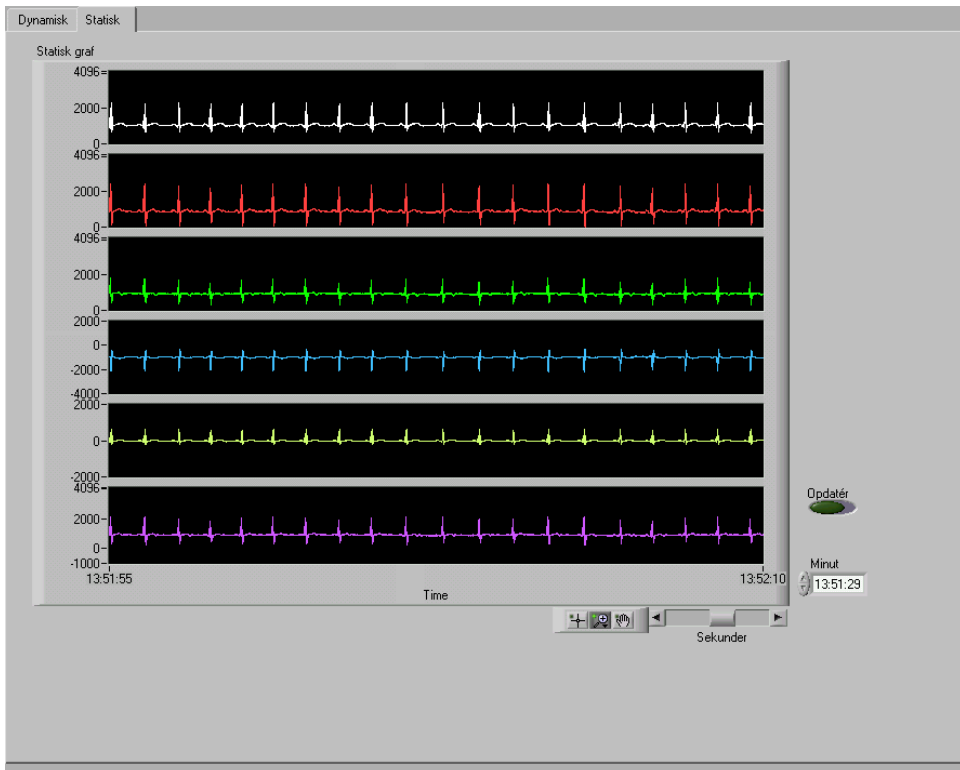
Præcis som opgave-beskrivelsen to-deles brugerfladen. Der designes ét skærmbillede til visning af dynamiske data (direkte fra porten) og et andet skærmbillede til visning af statiske data (indlæst fra fil).

Som vist på figur 14.1 består den dynamiske skærm af en waveform-chart, hvorpå data kan præsenteres, og en knap til at vælge hvilken afledning, der skal vises. På den statiske skærm kan alle 6 afledninger vises på en stacked waveform-chart, som det fremgår af figur 14.2. Der findes en minutvælger til valg af det minuts data, brugeren ønsker at se på grafen samt en graf-palette til zoom og scroll på grafen, så detaljer i EKG'et kan undersøges nærmere.





Figur 14.1: Figuren viser skærbilledet for visning af dynamiske data direkte fra microcontrolleren.



Figur 14.2: Figuren viser skærbilledet for visning af statiske data fra harddisken.

### 14.1.3 Programstruktur

Selve programmet ligger i baggrunden og knytter brugerfladens elementer sammen på den rigtige måde. Når der vælges en anden afledning på den dynamiske skærm, skal programmet sørge for at vise data for en anden afledning; når den statiske skærm vælges, skal data indlæses og vises på den statiske graf o.s.v. Også programmet er to-delt: Det består af en tråd til opsamling, der opsamler, lagrer og viser samt en tråd til filhåndtering, der genindlæser og viser. Programmets struktur fremgår af figur 14.3.

Inden selve hovedprogrammet gennemføres en række initialiseringer (1). Serielporten (com 1) initialiseres, så kommunikationshastighed m.v. stemmer overens med mc'ens specifikation, og for at undgå ressourcekrævende, dynamisk allokering af hukommelse under programafviklingen, initialiseres de nødvendige arrays fra start. Tillige indstilles den dynamiske graf, så tidsaksen stemmer overens med den aktuelle tid og samplingshastigheden. Tråden til opsamling af signalet fra serielporten kører hele tiden. Transmissionen synkroniseres, så der modtages en hel datapakke ad gangen (13). Hernæst hentes data (14), der skrives på skift til 6 forskellige filer i en ring(15), og den valgte afledning vises på skærmen (16). Hvis der opstår en transmissionsfejl, og den serielle strøm ikke længere er synkroniseret (17) gennemføres synkroniseringsproceduren på ny.

Tråden til genindlæsning af lagrede data gennemføres kun, når den statiske skærm er valgt (2). For at muliggøre statisk visning af data i mere end 5 minutter uden datatab, kopieres skrivedatafilerne (5), inden der læses data til grafen. På den måde undgås det, at de datafiler, der ønskes vist på den statiske graf overskrives, mens brugeren betragter graferne. Da lagring og læsning fra fil principielt kan foregå samtidig, er det hensigtsmæssigt at undgå at læse fra den fil, der skrives til lige nu. Derfor kopieres kun de 5 øvrige filer - svarende til 5 minutters data.

Opsamlingstidspunktet for data læses fra fil(6), og den statiske graf samt minutvælgeren indstilles med de korrekte tidspunkter (7). Den valgte data-fil læses (8) og vises på den statiske graf (9). Hvis brugeren vælger et nyt minuts data til præsentation på grafen (10), indlæses en anden data-fil til visning. Hvis brugeren ønsker at opdatere grafen (11), gennemføres hele tråden på ny, så de sidste nye data kopieres og indlæses. Når der bladres væk fra den statiske skærm (12) ophører afviklingen af tråden.

## 14.2 Implementation

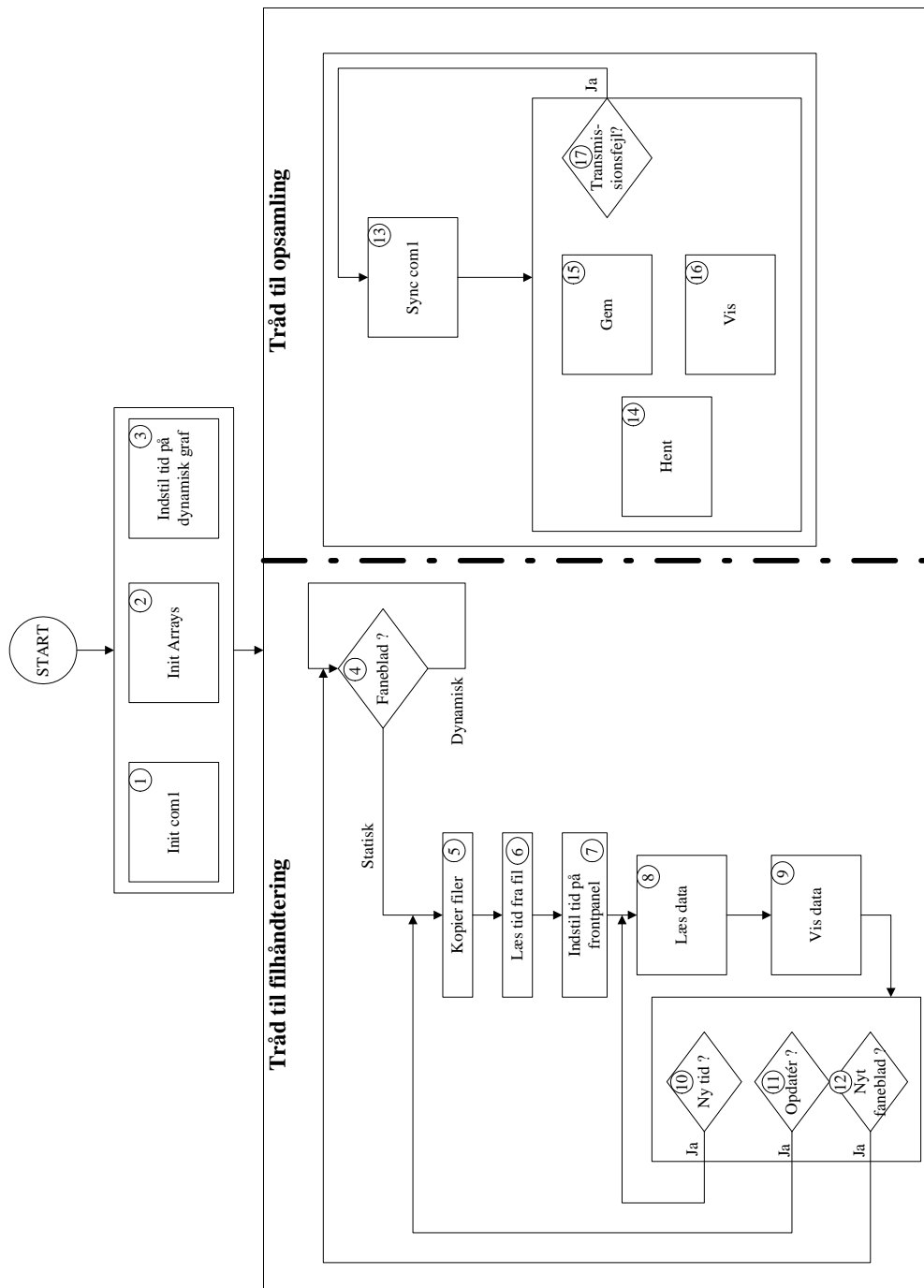
### 14.2.1 Tråd til opsamling

Tråden til opsamling, lagring og visning af real-time EKG-signaler fra mc'en er opdelt i tre moduler som vist på figur 14.4.

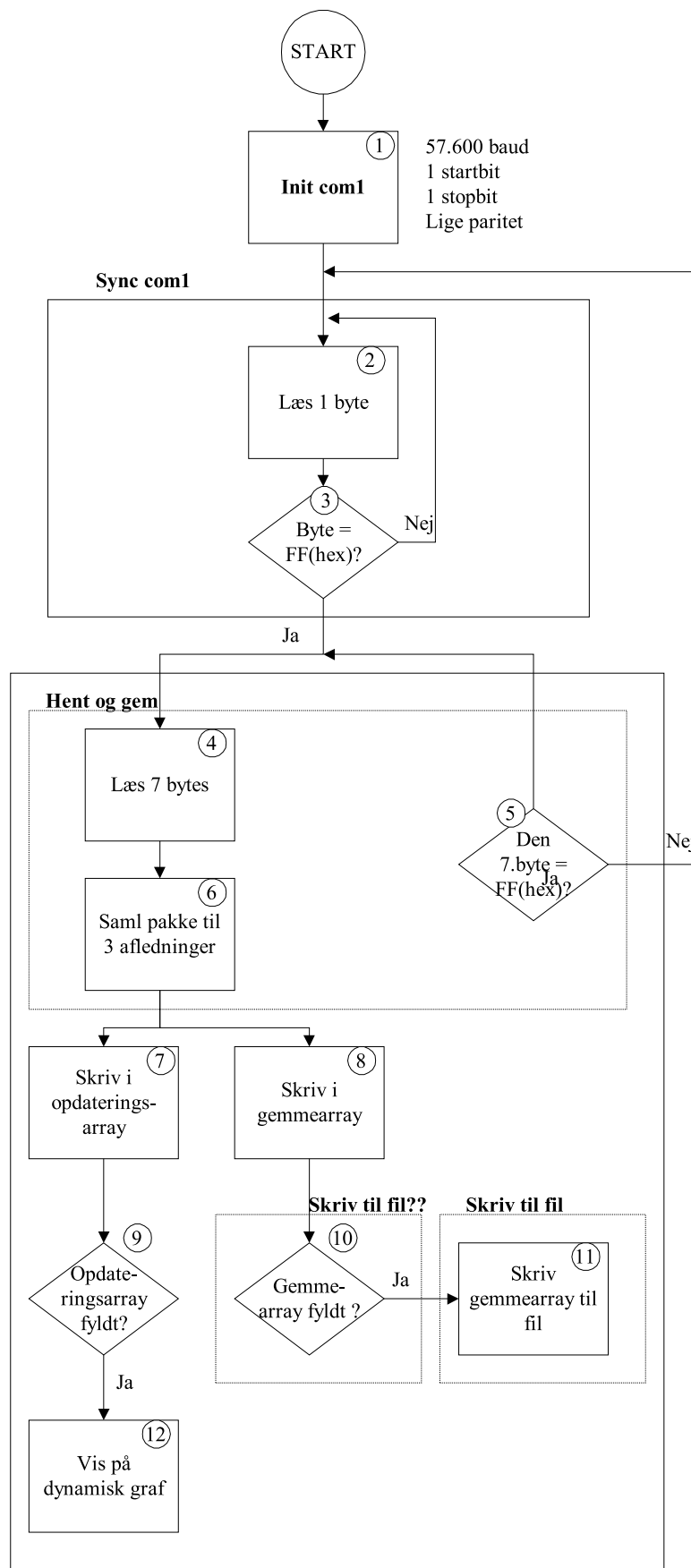
Seriel-forbindelsen initialiseres, hvorefter kommunikationen synkroniseres, så der læses fra starten af en data-pakke, og endelig hentes data én pakke af gangen, vises og lagres. I det følgende benyttes nummereringen af elementerne på figur 14.4 i parentes. Fra start initialiseres seriel-forbindelsen (1) til 57.600 baud, 1 startbit, 1 stopbit og lige paritet, så specifikationen stemmer overens med mc-UART'ens indstillinger.

Under synkroniseringen læses én byte af gangen (2), indtil en byte med værdien FF(hex), der angiver afslutningen på en pakke, modtages (3). Synkroniseringen er fuldført. Det er nu givet, at de næste 7 bytes på porten udgør en data-pakke. Derfor hentes nu 7 bytes (4), og de 6 data-bytes samles til 3 afledninger (6). Hvis der opstår paritetsfejl under transmissionen, meddeles dette på frontpanelet. Den 7. byte kontrolleres (5) - den skal have værdien FF(hex) for at data-strømmen fortsat er korrekt synkroniseret. Hvis dette ikke er tilfældet, gennemføres synkroniseringsproceduren på ny, og der meldes om "Synkroniseringsfejl" på frontpanelet.

De opsamlede data for de tre afledninger lagres i to arrays - et opdateringsarray (7) med længden 20 og et gemme-array (8) med længden 30.000. Hver gang opdateringsarrayet er fyldt (9),



Figur 14.3: Figuren viser programmets opbygning og opdeling i to tråde - én tråd til opsamling af signalet fra microcontrolleren og én tråd til læsning af data fra fil.



**Figur 14.4:** Figuren viser struktur for tråden til opsamling af signalet. Benævnelserne med fed skrift angiver navnet på en sub-VI, der kan findes i kildekoden. Elementernes nummerering anvendes i teksten.

skrives de 20 datapunkter til den dynamiske graf på frontpanelet (12). På denne måde opdateres grafen kun med frekvensen 25 Hz - det sparer systemressourcer at undgå hyppigere grafisk opdatering, og 25 Hz er tilstrækkeligt til at give det menneskelige øje fornemmelsen af en kontinuert opdatering. Gemme-arrayet rummer data, der er opsamlet i det foregående minut. Når arrayet er fyldt (10), lagres hele arrayet i en fil (11), og påfyldningen starter fra toppen igen.

SubVI'en "Skriv til fil??"(10) sørger desuden for, at der hele tiden skrives til den korrekte plads i gemme-arrayet, selvom der er foretaget synkronisering midt i en opfyldning og, at der skrives til fil, når der er lagret nøjagtig ét minuts nye data i gemme-arrayet. Til lagring af data benyttes LabVIEW-filtypen datalog-file. I denne filtype kan LabVIEWs datatyper (clusters, arrays m.v.) lagres direkte, og det er let at læse et enkelt element fra filen uden særlig hensyntagen til elementernes rækkefølge.

### 14.2.2 Tråd til filhåndtering

Som beskrevet ovenfor lagres ét minuts data af gangen. Data lagres som et array af clusters af 3 unsigned integers (det omtalte "gemme-array"), da waveform-chart'en, der skal bruges til visning af data kun accepterer et cluster som input. Hvert cluster udgør da én måling på 3 afledninger. Sammen med data lagres desuden lagringstidspunktet, så det er muligt at benytte dette til korrekt indstilling af tidsaksen på den statiske graf, når der læses fra filen. Strukturen for tråden til filhåndtering fremgår af figur 14.5.

Som vist på figuren tilpasses fil-kopieringen, så den ældste skrive-fil altid kopieres over i læs1, den næstældste i læs2 og så fremdeles (1). Efter endt kopiering læses lagringstidspunktet for "læs1"(2), der fratrækkes 60 sekunder, så start-tidspunktet for den lagrede fil fremkommer, og dette tidspunkt tilskrives derpå minutvælgeren og tidsaksen på den statiske graf som minimum-værdi (3). Det bliver dermed muligt at vælge fem forskellige tidspunkter med et minuts interval svarende til starten af hver af de fem lagrede filer.

Programmet betragter minutvælgerens værdi (der styres af brugeren) og læser data fra den tilsvarende fil (4). Ud fra de tre Einthoven-afledninger beregnes de tre Goldberg-afledninger (5), og alle seks afledninger vises på den statiske graf (6). Programmet venter på brugerens valg af nyt minut eller anmodning om opdatering (7). Hvis der vælges et nyt minut, indlæses den tilsvarende data-fil og denne vises. Hvis der anmodes om opdatering, kopieres skrive-filerne igen, så de fem læse-filer svarer til de nyeste, lagrede filer, inden den valgte data-fil indlæses og vises.

## 14.3 Test

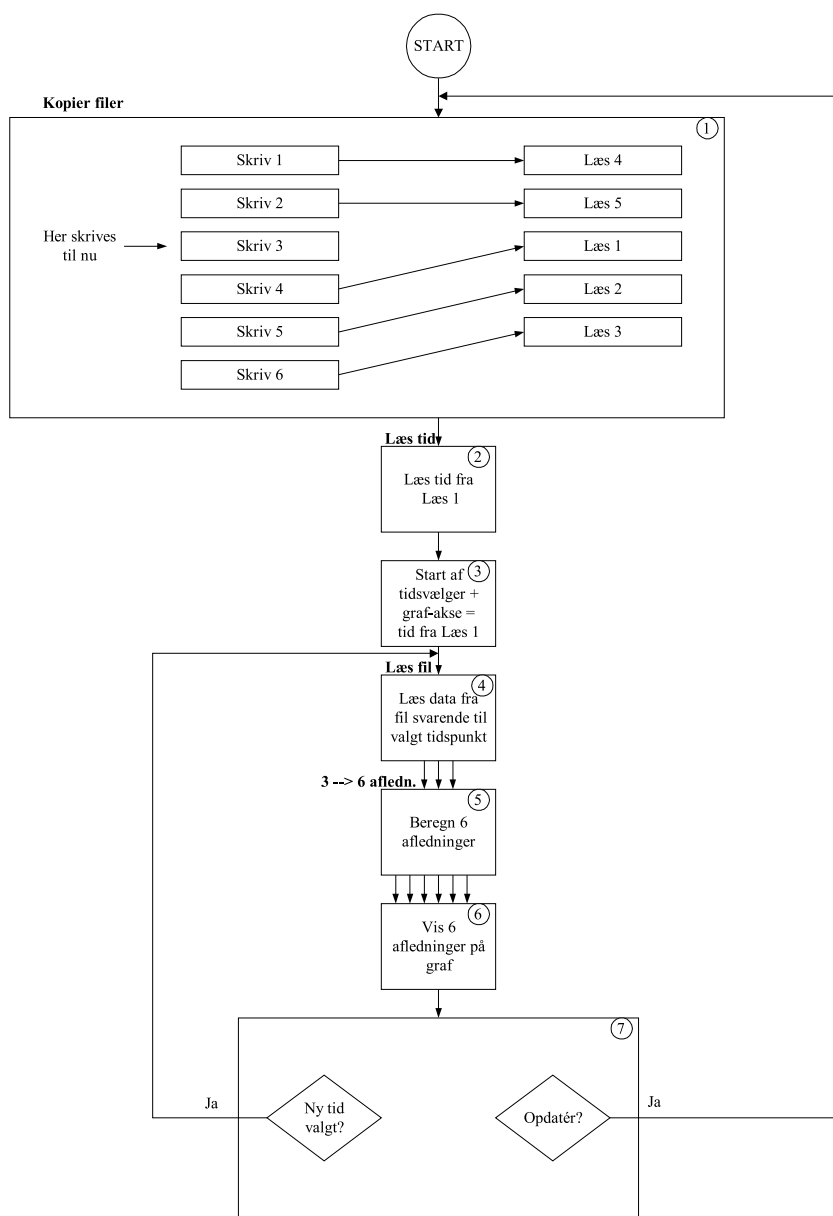
Brugerfladen på pc er testet ved først at gennemføre test af subVI's, og derefter teste processerne, dvs. trådene. Desuden er der foretaget integrationstest. Testene beskrives ikke i detalje, men resultatet og en kort beskrivelse af testopstilling angives.

### 14.3.1 Test af subVI's

Der er foretaget input/outputtest og korrigerende af programmerne, indtil det forventede output kom i alle realistiske input-tilfælde. Output er testet ved brug af "highlight execution", hvor det følges, hvordan og i hvilken rækkefølge programdelene eksekveres, samt ved brug af "prober", hvor man kan følge værdien af en variabel.

### 14.3.2 Test af processer

Der er foretaget funktionstest af processerne hver for sig. Synkroniseringen er testet ved at fremprovokere mætning af serielbufferen, så frontpanelet viste "Synkroniseringsfejl" og derefter undersøge, om signalet vist på frontpanelet vendte tilbage til et korrekt EKG. Testen viste, at synkroniseringen fungerer efter hensigten.



**Figur 14.5:** Figuren viser struktur for tråden til filhåndtering. Benævnelserne med fed skrift angiver navnet på en sub-VI, der kan findes i kildekoden. Elementernes nummerering anvendes i teksten.

Opsamling og dynamisk visning på frontpanelet er testet ved at benytte et oscilloskop som "facitliste" til EKG'ets udseende og undersøge, om EKG vises korrekt på den dynamiske skærm. Resultatet var, at de to billeder af EKG'et stemte overens - der vises efter hensigten.

Lagring af EKG i forskellige filer er testet med sinussignaler med varierende frekvens på indgangen til AD-konverteren på mc. Frekvensen blev løbende ændret, så frekvensændringerne kunne bruges til at identificere hvor i den 6 minutters periode, testen forløb over, en given datafil stammede fra. Efter de 6 minutter blev filerne åbnet i LabVIEW og det undersøgt, om filerne viste frekvensændringerne på rette tidspunkt. Testen viste, at det seneste minut blev gemt i "Læs 1", næstsensete i "Læs 2" osv., og at det var det første minut, der blev overskrevet. Lagringen sker altså som ønsket.

Processen til filhåndtering er testet ved på den dynamiske skærm at åbne data gemt i den dynamiske skærm. Minuttet, data ønskedes vist fra, blev valgt og vist på skærmen. Desuden blev én fil vist over en periode på flere minutter, så nye data lå i alle filer, så det kunne testes, at den ældste fil altid ligger i læs1. Testen viste, at filhåndteringen sker som forventet.

### Integration af processer

Det er testet, om der kan hentes/gemmes og læses fra filer samtidig. Denne test viste, at programmet fungerer som forventet, men at der er problemer med serielbufferen, der mætter under skiftet mellem de forskellige skærbilleder. En diskussion af dette problem findes nedenfor.

## 14.4 Problemer med opsamling af serielt signal

Under implementationen af ovenstående program blev det tydeligt, at opsamlingen af 3500 bytes i sekundet fra seriel-porten var en ganske ressourcekrævende opgave for LabVIEW/Windows. Serielportens buffer kan kun rumme 640 bytes af gangen, og det er derfor afgørende, at programmet tager fra hele tiden, så bufferen ikke mættes. Dette fænomen var et markant problem i starten af implementationsfasen - når bufferen mættes, vises/lagres der korrupte data og data går tabt. For at undgå dette problem blev opsamlingstråden optimeret mest muligt, så der ikke fandtes unødigt ressourcekrævende beregningsopgaver indenfor løkken, der skulle gennemføres for hver opsamlet data-pakke. Fra start var det hensigten at skrive data direkte til filen for hver opsamling, men da filadgang tager meget tid, blev denne procedure re-designet, så der kun kræves filadgang en gang i minuttet - data gemmes nu midlertidigt i et array (uden dynamisk allokering af hukommelse).

På samme måde opdateres skærmen (den dynamiske graf) nu kun 25 gange i sekundet og ikke for hver ny sample. Adgang til globale variable i LabVIEW tager også tid, da variablene lagres i en selvstændig fil, og opsamlings-løkken blev derfor re-designet, så globale variable kun tilgås uden for loopet. En anden optimeringsmulighed blev tillige forsøgt: I stedet for at læse én data-pakke af gangen, blev programmet indstillet til at læse, gemme og vise så mange data-pakker, som var tilgængelig på porten. Denne optimeringsprocedure viste sig imidlertid ufrugtbar, da den besværlige håndtering af et "ukendt" antal pakker var ganske ressourcekrævende. Med den nævnte optimering lykkedes det at få programmet til at køre hurtigt nok til at forhindre mætning af serielports-bufferen ved visning af den dynamiske graf. Der skal dog ikke mange "forstyrrelser" til, før problemet opstår på ny, og det er derfor vigtigt, at Windows ikke håndterer andre programmer i baggrunden, mens programmet kører.

Desværre opstår problemet stadig i nogen grad, når der skiftes til den statiske graf, og computeren i ca. et halvt sekund skal bruge ressourcer både på opsamling af data og kopiering/indlæsning af filer. I denne situation ville det være hensigtsmæssigt at kunne tildele seriel-porten højere prioritet eller have en større buffer på seriel-porten. Det kan konkluderes, at seriel overførsel af store datamængder til LabVIEW er en uheldig omstændighed, hvis programmet skal udføre andre opgaver en den isolerede opsamling.

# Kapitel 15

## Funktionalitetstest

*I dette kapitel vil en funktionalitetstest blive foretaget af det implementerede system.*

### 15.1 Afgrænsning

Som beskrevet i afgrænsningen kapitel 5, er det fra projektgruppens side valgt at bruge tiden på implementation frem for accepttest.

Dette valg begrundes med, at der kræves en stor mængde ressourcer, for at lave en forsvarlig accepttest. I stedet for test er tiden blevet brugt på at finpudse systemet, så det lever op til de krav, der er blevet stillet.

Systemet er under integrationen af processerne blevet afprøvet på kryds og tværs, og det er eftervist at de ønskede funktioner fungerer - også som en samlet enhed. Af ressourcemæssige årsager, dokumenteres kun en brøkdel af denne afprøvning i det følgende.

For at demonstrere systemets funktionalitet, vil følgende funktioner blive fremhævet i funktionalitetstesten:

- Pulsberegning
- Visualisering af afledninger

#### 15.1.1 Pulsberegning

Under implementationen af processen *Pulsberegning*, blev det testet og konkluderet at processen var i stand til at beregne en puls. Pulsberegning er et minimumskrav til systemet, så dette ønskes testet på det samlede system.

##### Testmetode

Testpersonen blev koblet til systemet med elektroder, samt til Propaq via en probe på fingeren. I 15 minutter blev pulsen målt på de to systemer. Måleresultaterne kan ses i tabel 15.1. Heraf ses at pulsen beregnet på systemet stemmer nogenlunde overens med Propaq'en. I gennemsnit var der ca. 3 pulsslæg til forskel.

#### 15.1.2 Visualisering af afledninger

I forbindelse med den løbende udvikling af centralskærmen, er det blevet testet om signaler blev opsamlet, konverteret og behandlet rigtigt på både mc'en og pc'en. Et minimumskrav til systemet er at det er i stand til at visualisere EKG-afledningerne på centralskærmen. I testperioden er



signalgeneratorer blevet brugt som input til mc og pc, men systemet skal desuden testes med et rigtigt EKG.

Der testes på Einthovens 3 afledninger, da Goldbergs afledninger, blot udledes ved beregning. En testperson kobles til systemet, og der tages "screen-shots" af afledningerne i LabVIEW og på oscilloskop, Resultaterne er vist på figurene: 15.1 til 15.6,

System	PROPAQ	Forskel
60	63	3
64	67	3
73	70	3
64	73	9
68	68	0
68	69	1
70	68	2
64	66	2
67	71	4
68	68	0
63	63	0
64	63	1
62	62	0
60	61	1
58	60	2
63	63	0
65	63	2
63	67	4
62	63	1
60	61	1
69	71	2
61	70	9
64	69	5
64	64	0
59	60	1
59	63	4
61	63	2
63	64	1
64	70	6
68	59	9
61	66	5
58	61	3
60	59	1
57	61	4
60	60	0
63	64	1
67	69	2
65	69	4
63	68	5
66	68	2
59	66	7
64	63	1
62	64	2
64	62	2
60	61	1
64	64	0

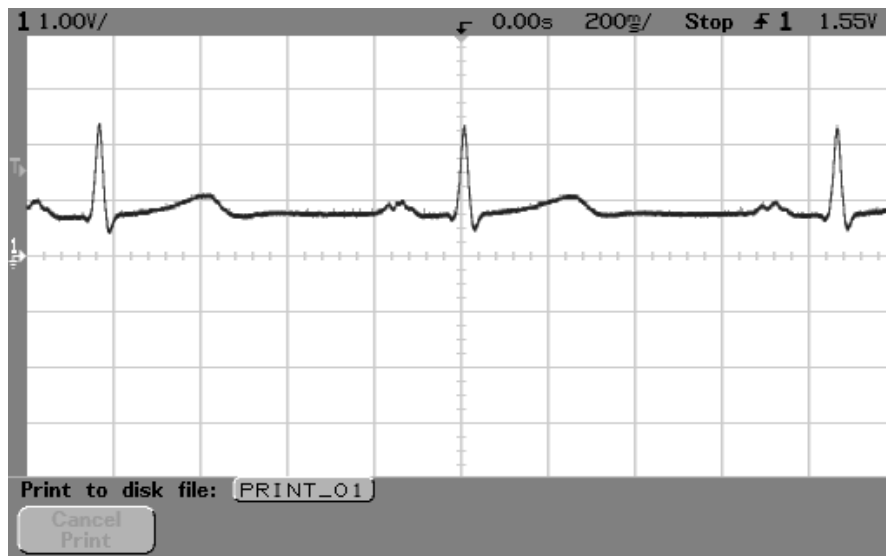
57	59	2
58	60	2
60	62	2
64	63	1
63	63	0
61	67	6
62	66	4
64	68	2
63	68	5
74	67	7
66	70	4
63	69	6
64	65	1
62	60	4
Total sum af forskel	=	164
Forskel i snit	=	2,73

*Tabel 15.1: Pulsværdier målt af system og PROPAQ. Der er omtrent overensstemmelse*

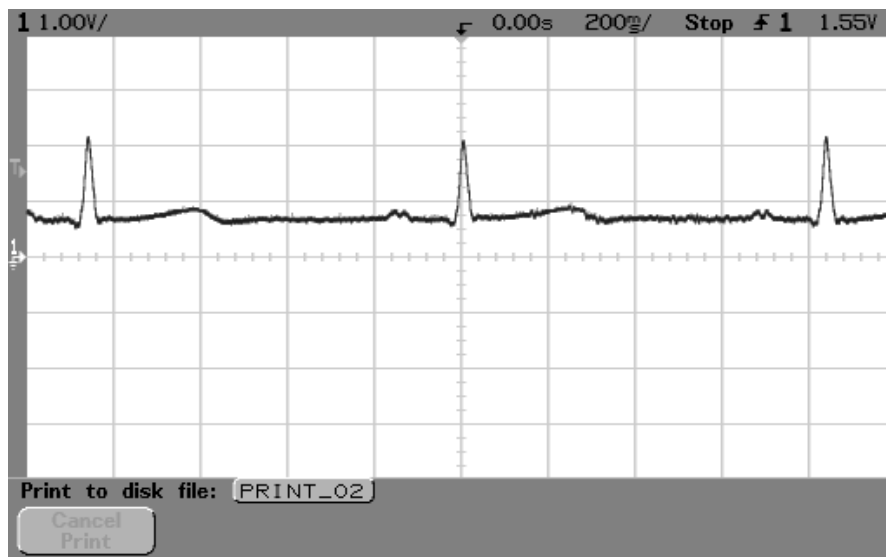
## 15.2 Screenshot fra oscilloskop og Labview



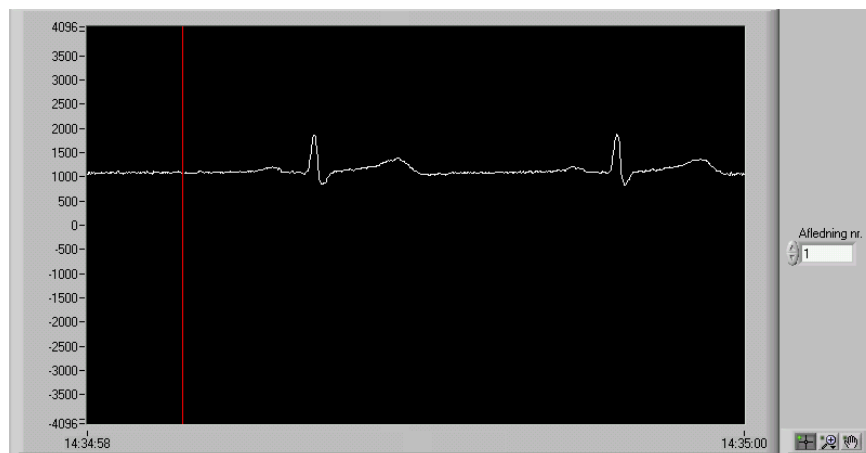
*Figur 15.1: Oscilloskop screenshot fra 1. afdning.*



Figur 15.2: Oscilloskop screenshot fra 2. afledning.



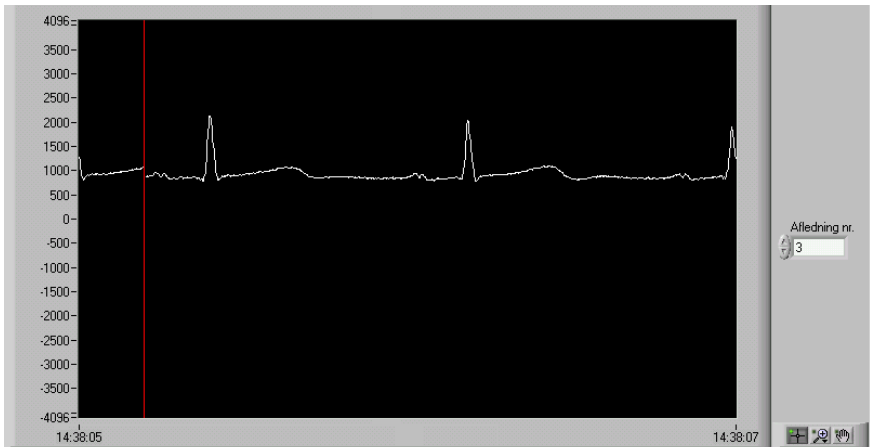
Figur 15.3: Oscilloskop screenshot fra 3. afledning.



Figur 15.4: Labview screenshot fra 1. afledning.



*Figur 15.5: Labview screenshot fra 2. aflødning.*



*Figur 15.6: Labview screenshot fra 3. aflødning.*

**DEL**  
**V**

**KONKLUSION OG PERSPEKTIVERING**



# Kapitel 16

## Konklusion

Gennem analysen af brugernes (læger/sygeplejersker) behov, ønsker og krav til et EKG-system til kontinuerlig overvågning af patienter er det klarlagt, at de vigtigste egenskaber ved et sådant system er visning af EKG-afledninger, pulsberedning samt alarmer for ventrikelflimmer, asystoli og tachy-/bradykardi. Det er nødvendigt med en bedside-del og en central del, så flere patienter kan overvåges fra samme sted. Systemet er designet vha. Struktureret Program Udvikling og inddelt i delsystemer, der for softwares vedkommende desuden er inddelt i processer.

Der er designet et system bestående af instrumentering, microcontroller, serielt kommunikationsled samt pc. Implementation er på microcontroller udført ved programmering i C, på pc ved programmering i LabVIEW.

Instrumenteringsdelen forbereder signalet til sampling ved 500 Hz på hver kanal ved forstærkning til 0,2-2,7 V i forforstærker med driven-right-leg-kredsløb og 8. ordens filter med forstærkning. Filterets knækfrekvens er 163 Hz.

Bedside anvendes en microcontroller til AD-konvertering og til at foretage alle beregninger og operationer nødvendige for at beregne pulsen og detektere alarmer. Pulsen beregnes over 3 pulsslag. Pulsslag i form af QRS-komplekser detekteres ud fra amplituden på den differentierede af EKG-signalet: i begyndelsen af hver brugssession kalibreres systemet over 5 pulsslag, så amplituden af top og bund af den differentierede af EKG-signalet kendes af systemet. Top og bund svarer til den opadgående hhv. nedadgående del af R-takken, der er meget stejl og derfor giver store udslag i den differentierede. Hvis der først sker en overskridelse af grænseværdien for detektering af en top og derefter af grænseværdien for detektering af bund, er der tale om en R-tak og altså et pulsslag. Ud fra pulsen samt grænser for brady- og tachykardi, der kan indstilles vha. knapper i forbindelse med et LCD-display, beregnes det, hvorvidt der er alarm eller ej. Der detekteres asystoli, hvis pulsen er under 20, og tachy- og bradykardi, hvis pulsen er hhv. over og under det indstillede normalområde. Ventrikelflimmer detekteres ikke, da denne arytmi kan have meget forskelligt udtryk og dermed er svær at detektere. Hvis der er alarm, lyder en summer med bip ved pulsalarm og kontinuerlig lyd ved asystoli. Alarmer kan mutes i 2 minutter og går herefter igang igen med den type lyd, der svarer til typen af den alarmkrævende situation. Al interaktion med brugeren sker gennem knapper og LCD-display, der kan vise flere forskellige skærbilleder, alt efter hvordan der trykkes på knapperne.

Gennem en serielforbindelse transmitteres Einthovens 3 afledninger til pc, hvor Goldbergs 3 afledninger beregnes og kan vises sammen med Einthovens 3 på et dynamisk skærbillede, hvor brugeren kan vælge, hvilken afledning der skal vises. Einthovens afledninger gemmes i 5 filer á 1 minut, som hentes, hvis brugeren vælger at se lagret data i et statisk skærbillede. Alle 6 afledninger ses i det statiske skærbillede, så Goldbergs afledninger beregnes igen. Under transmissionen kontrolleres det hele tiden, at datapakkerne er uden fejl, og at de kommer i den rigtige rækkefølge. Når der er opsamlet data for 1 minut, gemmes data i én af de 5 filer, så en hel fil gemmes ad gangen for at spare systemressourcer. Det opleves ved skift mellem dynamisk og sta-



tisk skærm, at seriel-bufferen mættes på grund af bufferens begrænsede størrelse og manglende systemressourcer.

Systemet er implementeret og funktionelt. Funktionalitetstesten viser, at systemet fungerer som forventet, men der er ikke foretaget egentlig accepttest.

## Kapitel 17

# Perspektivering

Projektgruppen anser de udførte tests for udpræget mangelfulde grundet de begrænsede tidsmæssige ressourcer. Det er derfor en nødvendighed, at systemet testes mere dybdegående, inden udviklingsarbejdet fortsættes.

Det designede og implementerede system er et skridt på vejen mod et EKG-system, til anvendelse på en kardiologisk afdeling. Men der er områder i systemet, der kunne optimeres, for at højne systemets værdi og anvendelighed.

Det har gennem test vist sig, at muskelstøj i høj grad opfanges og opfattes af systemet som QRS-komplekser, og derfor medfører beregning af en forkert og unaturligt høj puls. Løsning på dette problem kunne oplagt være implementation af et digitalt båndpas-filter, der i forbindelse med QRS-detektering frafiltrerer de forholdsvis højfrekvente muskelsignaler, der opstår når patienten bevæger sig. Implementation af et digitalt filter vil stabilisere pulsregningen og eliminere en stor procentdel af de fejlalarmer, der opleves. Samtidig vil den diagnostiske nøjagtighed være bevaret, da filtreringen kun sker i forbindelse med pulsregningen.

Det velkendte problem med 50 Hz støj er allerede forsøgt reduceret vha. driven-right-leg kredsløb, men for yderligere at begrænse 50 Hz signaler, kan et digitalt notch-filter med fordel implementeres på mc'en.

I sin nuværende form kan systemets centralskærm kun udføre en lille del af de opgaver, der forventes på en kardiologisk afdeling, og det ville være ønskeligt at parametre som pulsregning, alarmdetektion, lagring af data for flere timer samt trendkurver for puls og events implementeres, hvilket af projektgruppen anses som en overkommelig opgave.

Herefter vil en oplagt mulighed for forbedring være at skabe kommunikation fra pc til mc, så det centralt bliver muligt at vælge hvilken afledning, der vises bed-side, samt mulighed for at mute bed-side.

Bed-side vil næste skridt i udvidelser af systemet være visning af EKG'et, som beskrevet i analysen. Det vil give lægen og sygeplejersken mulighed for at følge med i patientens tilstand når de er på patientstuerne. Men de tilgængelige displays lægger op til en ny diskussion.

Bed-side kan det implementerede systems brugergrænseflade ikke accepteres på kardiologisk afdeling. Det lille display med kun 4 liniers tekst, blev udviklet for at udnytte så mange af mc'ens funktionaliteter som muligt, men det kan ikke tilfredsstille kardiologer og sygeplejersker, der i dag er vant til store monitorer på væggen. Derfor vil en udgave til kardiologisk afdeling ikke indeholde en skærm i direkte forbindelse med mc'en. Her vil mc'en kun blive brugt til dataopsamling, hvorefter al signalbehandling vil foregå på centrale computere. Fra de centrale computere vil det behandlede signal blive sendt ud til centralskærme og bed-side monitorer.

Samlet vurderes det derfor af projektgruppen, at det nuværende system ikke vil få en fremtid på en kardiologisk afdeling, men at processen i arbejdet med at opsamle og behandle data på en

microcontroller har været lærerig og vil kunne udnyttes til at designe et rigtigt EKG-system på et senere tidspunkt.

# Litteratur

- [1] Agamemnon Despopoulos and Stefan Silbernagl. *Color Atlas of Physiology, 4th edition*. Thieme, 1991.
- [2] John G. Webster. *Medical Instrumentation, Application and Design*. John Wiley & Sons, Inc, 1998.
- [3] John G. Webster. *Encyclopedia Of Medical device and Instrumentation, Application and Design, Volume 2*. John Wiley & Sons, Inc, 1988.
- [4] Aalborg Sygehus Syd Lars Hvilsted Rasmussen, Overlæge på Kardiologisk Afdeling, 2002.
- [5] Christian Lange. *Vejledning i EKG mønstre*. Medicotest A/S, 1997.
- [6] Bruce B. Winther and John G. Webster. Driven-right-leg circuit design, iee transactions on biomedical engineering. Technical report, IEEE, 1983.
- [7] Ken Yoshida and Andrei Patriciu. Changes in the skin and electrode impedance spectra due to long-term surface stimulation. Technical report, Center for Sensory Motor Interaction, Aalborg University, Denmark, 2000.
- [8] IEC. Safety requirements for medical electrical systems,, 1992.
- [9] Alan V. Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing*. Prentice Hall, 1998.
- [10] American Heart Association, 2002.
- [11] Electronics Industries Association (EIA). Rs-232-c standard. Technical report, Electronics Industries Association (EIA), August, 1969.
- [12] Projektvejleder Kim Dremstrup Nielsen, 2002.
- [13] Stephen Biering-Sørensen et al. *Håndbog i Struktureret Program.Udvikling*. Ingeniøren - Bøger, 2000.
- [14] Texas Instruments. Msp430x1xx family user's guide, 2001.
- [15] P. Laguna et. al. New algorithm for qt interval analysis in 24-hour holter ecg: performance and applications. Technical report, Med. & Biol. Eng. & Comput., 1990.
- [16] Andrew S. Tanenbaum. *Structured Computer Organization, 4.th Edition*. Prentice-Hall International, 1999.
- [17] Texas Instruments. Msp430x13x, msp430x24x mixed signal microcontroller, 2000, rev. 2001.
- [18] Adel S. Sedra and Kenneth C. Smith. *Microelectronic Circuits*. Oxford University Press, Inc, 1998.



# DEL VI APPENDIKS

*I appendiks forefindes kapitler som relaterer sig til meget specifikke områder af rapporten.*

*I appendiks A forefindes et referat for et besøg på Aalborg sygehus Syd, hvor interview med klinikere er foretaget. Disse interviews anvendes i forbindelse med foranalyse samt opstilling af brugerkrav i forbindelse med analysen.*

*I appendiks B forefindes beskrivelse af opsamlingsmetoder for EKG.*

*I de efterfølgende appendiks forfindes hardwaremæssige specifikationer. Blant andet forefindes design og implementation af instrumenteringsdelen samt seriel kommunikation med pc. Herunder findes også en overordnet beskrivelse af den benyttede mc: MSP430F149 I afsnit 15 forefindes et optaget EKG af en af gruppens medlemmer samt kildekode for det endelige system.*



## Appendiks A

# Referat af besøg på Aalborg sygehus Syd

Et besøg på Aalborg Sygehus Syd blev arrangeret i forbindelse med projektstart. Målgruppen lå ikke klar, derfor blev der i første omgang arrangeret et besøg på både på sygehusets Anæstesi-Intensiv- og operationsafdeling, samt Kardiologisk Afdeling.

Besøgets primære funktion var at få et indblik i anvendelsen af EKG som et værktøj på de pågældende afdelinger, samt hvilke ønsker der ville være til et EKG-monitoreringsapparat. Altså, hvad er det egentlig man bruger EKG til?

Første del af besøget foregik på Anæstesi- Intensiv- og operationsafdelingen hvor overlæge på Thoraxintensiv afdeling(TIA), Bodil Steen Rasmussen bød velkommen. Anden del af besøget foregik på Kardiologisk Afdeling ved Lars Hvilsted Rasmussen, administrerende overlæge på afdelingen.

Der blev spurgt til hvordan afdelingernes dagligdag fungerede. Anvendelse af EKG som diagnostisk værktøj på afdelingerne. Funktionaliteter og begrænsninger ved monitorering af EKG. Samt hvilke yderligere ønsker der ville være til en deres eksisterende EKG-apparater.

### A.1 Anæstesi- Intensiv- og operationsafdeling

På afdelingen monitoreres der både i patientstuerne (bed-side), hvilket også benævnes TIA1, og på en hovedskærm i en overvågningsstue (TIA2), hvor optagelser af alle afdelingens patienter er samlet (Centralskærm). Ved morgen- og dagvagter, forefindes der en sygeplejerske for hver patient. Ved nattevagt findes hovedsageligt kun sygeplejersker i overvågningsstuen.

Afdelingen består af 9 sengeafsnit.

Der anvendes 5 elektroder, hvorved 8 standardafledninger fås for EKG.

Den typiske indlæggelsestid for en patient på Anæstesi- Intensiv- og operationsafdelingen, er 24 timer.

Parametre der bliver monitoreret for er: blodtryk, puls, asystoli og ventrikelflimmer, tachy og bradykardi

Alarmtyper: Røde (asystoli og ventrikelflimmer) og grønne (fx tachy- og bradykardi) alarmer.

Alarmgrænser for de grønne alarmer ændres individuelt for hver patient.

Begrundelse for monitorering; hjertepatienter vil efter operation normalt have en meget lav puls omkring 50, hvilket gør dem sårbare. Derfor er monitorering vigtig for at undgå forskellige hjertesvigt, blodpropper m.m.



### A.1.1 Funktionaliteter ved eksisterende system

Mulighed for visning bedside og ved centralskærm. På punktform opridses nedenfor de funktionaliteter systemet varetager, samt de begrænsninger systemet har.

- Bed-side
  - History funktionalitet, med tilhørende visning af trends. Dvs. indikationer af de alarmer der har været. Visning af et lagret EKG over et 72 timers forløb.
  - EKG monitoren bedside er transportabel. Følger patienten fra operationsstuen til Anæstesi- Intensiv- og operationsafdelingen.
  - Mulighed for at slå lydalarm fra på bedside monitoren i to minutter.
  - Mulighed for at slå alle alarmer fra på bedside monitoren, når patienten er så syg, at reaktion på alarmer ikke længere vil kunne redde patienten.
  
- Centralskærm
  - Fastfrysning af EKG.
  - History funktionalitet. Visning af et lagret EKG over et 72 timers forløb.
  - Trends for de sidste 24 timer. Dvs. indikationer af de alarmer der har været.
  - Zoomfunktionalitet. Episoder af 3 sekunders varighed kan vises.
  - Udprintningsmulighed.
  - Indtastning af personlige informationer; navn, cpr nr.
  - Horisontal zoom.
  - Mulighed for begrundelse af ændrede grænser.
  
- Begrænsninger ved systemet
  - Kun overvågning af et lige antal patienter af pladsmæssige begrænsninger; 2, 4, 8 etc. Ingen større omskiftning når der skiftes mellem visning af patienter. Alarmering for en mulig 9. patient bliver ikke vist ved centralskærmen, da der kun er plads til 8 patienter på centralskærmen.
  - Ingen markeringer af events på udskrift.
  - Gemmer kun de sidste 8 arytmier.
  - Ikke mulighed for at vise trends både som tal og som kurve.

Pris: 100.000 kr for. Siemens monitor alene. 500.000 alt i alt for seng med ophæng m.m.

### A.1.2 Vurdering af systemet

Det er muligt at skabe en naturlig døgnrytme, idet alarmer kan slukkes permanent bedside. Om natten opdages alarmen af sygeplejersken, der overvåger centralskærmen.

Systemet er logisk bygget op og nemt at gå til vha. drejeknap. Man kan slukke alarmen fra begge sider af sengen. Der er desværre mange fejlalarmer, hvilket kan give urolige patienter. Det skønnes, at hver anden alarm er en fejlalarm.

Størrelsen på det visualiserede EKG er tilfredsstillende.

Udskrifter anvendes ved alarm, men først efter at have kontrolleret, hvad der står på skærmen.

## A.2 Kardiologisk afdeling

Afdelingen består af 31 sengepladser, der er fordelt på 2- og 4-mandsstuer. I 2-mandsstuerne er der en enkelt monitor på væggen, mens der i 4-mandsstuerne er 2. Der benyttes Einthovens 3 afledninger, som optages og sendes via telemetri fra patienten til en central overvågningspult.

De tre optagede afledninger gemmes i op til 72 timer, og kan let hentes frem og skrives ud på en laserprinter. Dette benyttes specielt om natten, når der ikke umiddelbart er en læge tilstede. I stedet printes EKG-sekvensen, der gav alarmer, og videregives til lægen næste morgen.

På centralskærmen vises en af de optagede 3 afledninger sammen med pulsen. Da de kardiologiske læger for det meste er interesserede i P-takkens størrelse og udseende, vælger sygeplejersken ved centralskærmen, den af de 3 afledninger, der tydeligst viser netop denne tak. Den valgte afledning vises så på centralskærmen og bed-side monitoren.

Hovedprincippet for lægens diagnosticering på kardiologisk afdeling er, at lægen tager en udskrift af EKG'et og sammenligner de forskellige afledninger med et tidligere optaget EKG fra patienten. Her er det mønstergenkendelse, der studeres og diagnosticeres ud fra.

Diagnosticerende EKG-monitoren, som fx findes på skadestuen er ikke brugbare på afdelingen, da de er ukorrekte, og de kardiologiske læger er så kompetente, at maskinens diagnose ikke er til nogen hjælp.

Som en hovedregel anses det som en nødvendighed, at alarmering og behandling af livstruende arytmier som ventrikelflimmer og asystoli foretages indenfor 5 minutter. Derefter risikeres permanent hjerneskade hos patienten - hvis han overlever.

Én gang dagligt tages et EKG af hver patient, der lægges i patientens journal. 3 gange dagligt noteres de arytmier der måtte have været for henholdsvis morgen-, dag- og nattevagten i journalen.

Det antages at ca. 85 % af de pulsalarmer der forekommer på afdelingen, er fejlalarmer. Dette skyldes hovedsageligt at patienten ligger uroligt, eller på anden måde er fysisk aktiv, hvoraf der vil forekomme muskelstøj på det signal man måler. Dette forklarer også at der "kun" er 50 % fejlalarmer på TIA, da patienterne på denne afdeling udelukkende er sengeliggende.

---

Anæstesi- Intensiv- og operationsafdeling	Kardiologisk afdeling
Bodil Steen Rasmussen, Overlæge	Lars Hvilsted Rasmussen, Adm. overlæge Ph.D.
Thoraxintensivt afsnit (TIA)	Klinisk kardiologi.

---



## Appendiks B

# Principper for EKG opsamling

Opsamlingen af EKG'et foretages ved hjælp af elektroder placeret på kroppens overflade. Der findes forskellige modeller til at opsamle EKG på, men overordnet findes der to typer af afledninger; bipolar afledning og unipolar afledning. Bipolar (to-polet) afledning er, som navnet antyder, en afledning, hvor der foretages måling af spændingsforskellen mellem to elektroder. Ved unipolar (en-polet) afledning måles spændingsforskellen mellem en konstant reference-spænding, der udgøres af to eller tre elektroder, og en eksploderet elektrode.:-=) [1]

### Afledningsmodeller

Elektroderne kan placeres ud fra forskellige afledningsmodeller. De tre mest anvendte er Einthovens, Goldbergs og Wilsons. I det følgende beskrives de tre modeller.

#### *Einthoven*

Einthovens model for opsamling af EKG er en bipolar afledning, hvor der som resultat fås standardafledningerne I, II og III. Disse tre ekstremitetsafledninger fås ved hjælp af tre elektroder. Én på højre arm, én på venstre arm og én på venstre ben. Derved får man følgende udtryk for afledningerne, hvor det er potentialerne målt med elektroderne, der indgår. Se figur B.1.

$$I = v.arm - h.arm$$

$$II = v.ben - h.arm$$

$$III = v.ben - v.arm$$

#### *Goldberg*

Goldbergs model for opsamling af EKG er en unipolar afledning, og der bruges tre elektroder placeret som ved Einthovens. Med Goldbergs model kan der opnås yderligere 3 afledninger, og derved kan man med kun 3 elektroder opnå 6 afledninger ved at benytte Goldberg og Einthovens modeller.

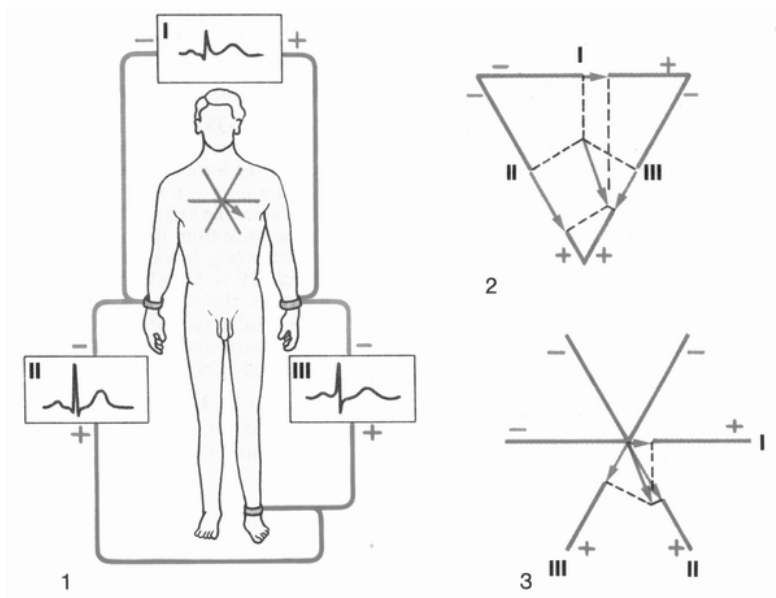
Goldbergs afledningsmodel giver tre afledninger  $aV_F$ ,  $aV_R$  og  $aV_L$ , der findes ved at tage 2 elektroder som referencepunkt i forhold til den sidste elektrode. Se figur B.2

De 3 afledninger er givet ved følgende:

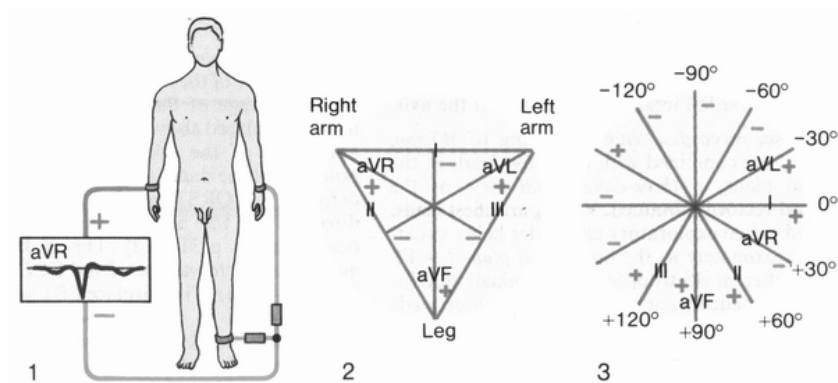
$$aV_R = h.arm - \frac{v.arm + v.ben}{2}$$

$$aV_L = v.arm - \frac{h.arm + v.ben}{2}$$

$$aV_F = v.ben - \frac{h.arm + v.arm}{2}$$



**Figur B.1:** På figuren ses de tre afledninger, der kan opnås ved Einthovens model.

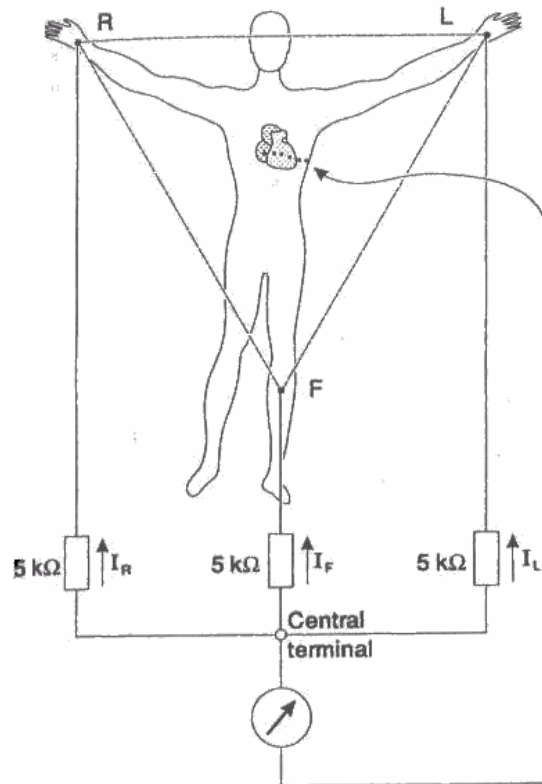


**Figur B.2:** På figuren ses en af de tre afledninger man kan opnå med Goldberg's metode, i dette tilfælde aVR. Dette gøres ved at forbinde to elektroder som reference og bruge den sidste som eksploreret elektrode.

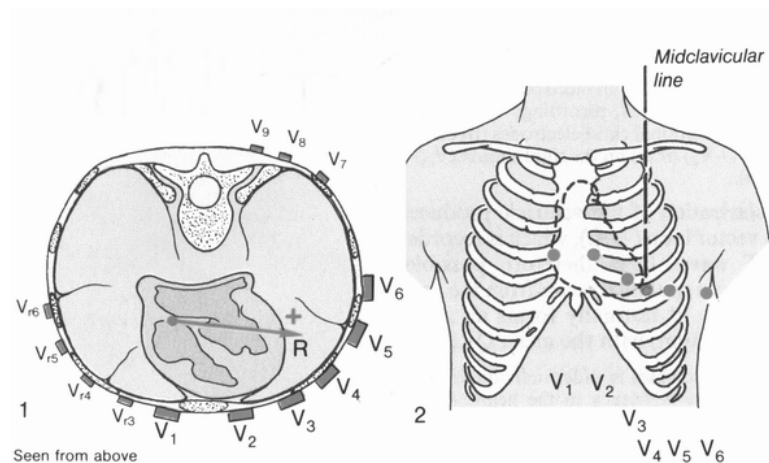
### Wilson

Wilson's model, er som Goldbergs en unipolær afledning, hvor de 3 elektroder på ekstremiteterne er samlet til central terminal (CT). Se figur B.3.

Wilson's afledningsmetode betegnes også prækordial afledning, da de anvendte elektroder - oftest 6 - sidder prækordialt på patienten. De 6 elektroder er placeret således, at den første elektrode er placeret lige til højre for sternum i det 4. interkostalrum, og derefter er der placeret fem elektroder videre på tværs af brystet mod venstre, hvor den 6. elektrode sidder i den venstre midttaksilærline på højde med det 5. interkostalrum. Se figur B.4



**Figur B.3:** Figuren viser hvordan de 3 afledninger samles til CT.



**Figur B.4:** På figuren ses elektrodeplaceringen, når Wilson's metode benyttes

Ved at samle de 3 elektroder på ekstremiteterne i CT, kan der ses ind på hjertet fra flere forskellige vinkler. Wilson's model kan udbygges med yderligere tre elektroder, som placeres på højde

med 5. interkostalrum over til venstre for columna, samt 4 elektroder, der placeres prækordialt mod højre. Se figur B.4

De forskellige afledninger udledes på følgende måde: n angiver afledningen, der kan antage værdierne 1 til 12.

$$V_n = v_n - \frac{h.arm + v.arm + v.ben}{3}$$

Des mere detaljeret billede, der ønskes af hjertets tilstand, des flere afledninger skal der bruges. Derved opnår man at se ind på hjertet fra flere forskellige vinkler. Hvor mange afledninger, der skal bruges til en EKG-måling, og dermed hvilken opsamlingsmodel, man skal bruge, afhænger derfor af anvendelsen af EKG'et.

## Appendiks C

# Hardware specifikation - mc'en (MSP430F149)

*Følgende kapitel omhandler features ved den anvendte mc. Kun features anvendt i forbindelse med udviklingsforløbet er medtaget og beskrevet. For yderligere dokumentation henvises til [14], [17].*

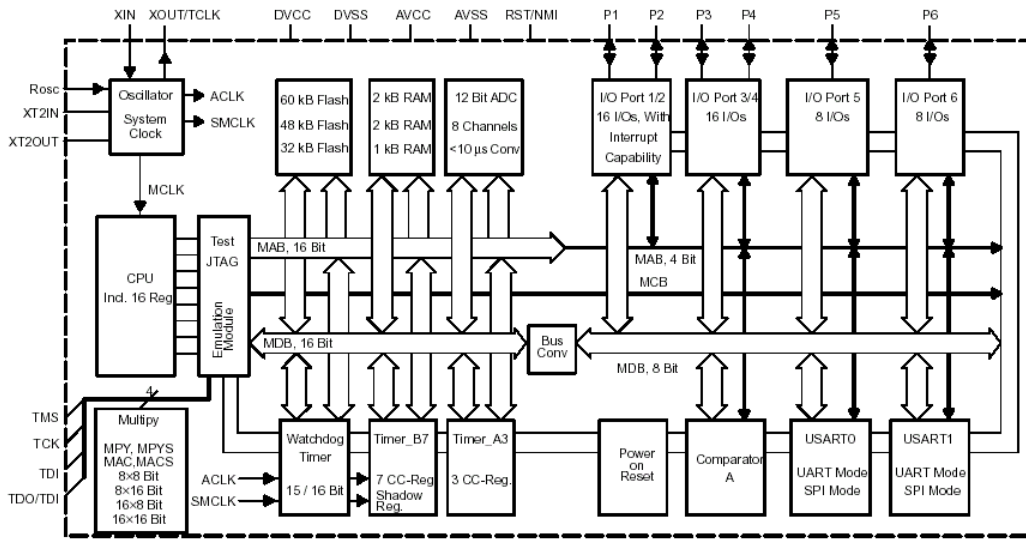
MSP430F149 er den udleverede mc og omfatter følgende features;

- Basic Clock Module
- Watchdog Timer / General Timer
- Timer A ( 3 sammenlignings registre), Timer B ( 7 sammenlignings registre )
- I/O Port: 1,2 - med interrupt
- I/O Port: 3,4,5,6 - uden interrupt
- Comparator\_A
- ADC12 (12 bit A/D)
- USART0, USART1
- Hardware multiplier

MSP430F149 er en 16 bit RISC microcontroller, der bygger på Von Neumann arkitektur med en intern konfiguration som illustreret på figur C.1.



**MSP430x14x**



*Figur C.1: Illustration af den udleverede microcontrollers interne struktur. Figur taget fra [17].*

## C.1 Anvendte mc features

### C.1.1 Hukommelsen

Mc'en indeholder 2Kb RAM, samt 64Kb + 256b Flash Memory.

### C.1.2 Basic Clock module

Mc'en er i gruppens version udstyret med 2 eksterne klokke, Masterklokken (MCLK) på 7,3728 MHz og A-klokken (ACLK) på 32,768 kHz. Desuden bruges mc'ens egen softwareklokke (ADC12OSC) på ca. 5 MHz og submasterklokken, der i software er sat op til det samme som masterklokken. masterklokken bruges til at klokke CPU'en.

### C.1.3 ADC

ADC modulet på mc'en består af en analog multiplexer, A/D-konverter, intern hukommelse register, reference spændinger og sampling/konverterings klokke. Der gøres i dette projekt brug af 3 kanaler (A0 - A2) på multiplexeren, som konverteres og lægges ned i de 3 første interne registre (ADC12MEM0 - ADC12MEM2).

### C.1.4 Timer A

Timer A består af en 16 bit tæller, som klokke forsynes med submaster klokken på 7,3728 MHz. Timeren kan benyttes i capture eller compare mode.

### C.1.5 Timer B

Timer B består af en 16 bit tæller, som klokke forsynes med clock A på 32,768 KHz, der underdeles med 8. Timeren kan benyttes i capture eller compare mode.

### C.1.6 Watchdog Timer

Denne kan vælges at benytte som almindelig timer. WatchdogTimeren består af en 16 bit tæller, som klokke forsynes med 32,768 KHz, der underdeles med 8.

### C.1.7 USART0

USART'en bruges som en UART, der transmitter med 57.600 baud rate, som genereres ud fra submasterklokken (7.3728 MHz.) delt med 128.

### C.1.8 I/O Port 1,2

Portene 1 og 2 består hver af 8 linjer som individuelt kan adresseres og kontrolleres for hver individuel bit. Alle mulige kombinationer er tilladelige. Interrupt processering af eksterne events er mulig. Begge porte kan konfigureres som enten input eller output port ved at sætte retningsregisteret (P1DIR/P2DIR) højt eller lavt. Læsning fra portene sker med kommandoen (P1IN/P2IN), og der kan skrives til portene med kommandoen (P1OUT/P2OUT). Når portene bruges til at styre interrupts, skal der vælges om transition (P1IES/P2IES) fra logisk høj til lav eller omvendt, skal generere et interrupt. Hvis et interrupt genereres fra en linje sættes et tilhørende flag (P1IFG/P2IFG), som indikation for hvilken linje der genererede interruptet. For at benytte portene som interruptporte sættes den generelle interrupt enable funktion `_EINT()`, PIE1 og PIE2.

### C.1.9 I/O Porte 3,4,5

Portene 3,4 og 5 består ligeledes hver af 8 linjer som individuelt kan adresseres og kontrolleres. Disse porte kan ikke generere interrupts, ellers gælder samme kommandoer som for portene 1 og 2.

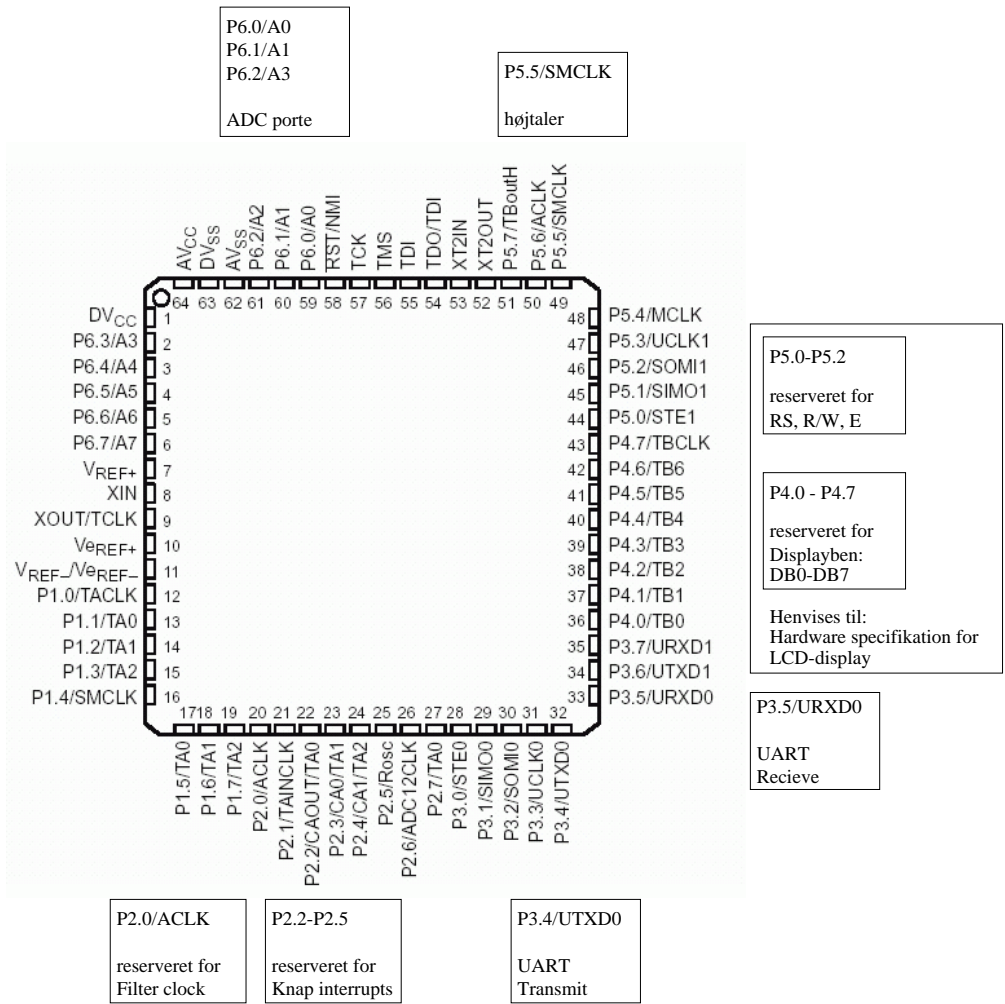
## C.2 Interrupt struktur for MC

MSP430-serien understøtter simpel implementation af interruptstyrede rutiner. Prædefinerede interruptvektorer for hvert interruptflag angiver placeringen af den enkelte interruptservice-rutines placering i hukommelsen.

## C.3 Anvendelsen af eksterne enheder

For instrumentering, seriel-kommunikation og kommunikation til display, anvendes eksterne enheder der benytter I/O-portene på mc'en. Disse forbindelser er illustreret på figur C.2. Figuren henviser til efterfølgende kapitler;

- Appendix D.1 for forforstærkning.
- Appendix D.2 for filtrering.
- Appendix E for seriel kommunikation.
- Appendix F for LCD-display.



**Figur C.2:** MSP430F149 med forklaring til eksterne enheder. Kasserne fortæller først hvilke ben på mc der benyttes, hvorefter til hvilket formål de tjener ved eksterne ben el. indgang / udgangsbene for eksempelvis filter.

## Appendiks D

# Design, implementation og test af instrumentering

*I det følgende designs forforstærkeren, der skal foretage den primære forstærkning af EKG-signalet fra patientens krop.*

### D.1 Design og implementation for forforstærker

#### D.1.1 Krav til forforstærker

Der er stillet følgende krav til forforstærkeren:

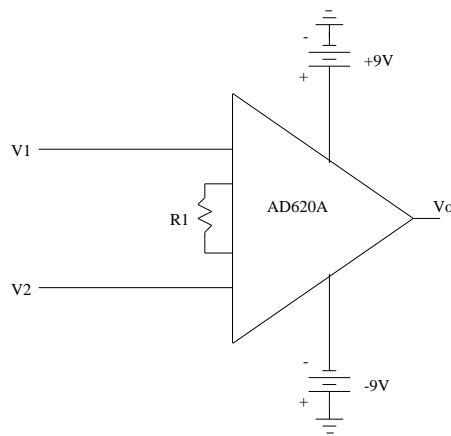
- Der skal opsamles i alt 3 kanaler.
- Det skal benyttes Ag-AgCl elektroder, til opsamling af patientens EKG-signal.
- Det opsamlede EKG signal må maksimalt svækkes 1 % inden forforstærker.
- Beskyttelsesdioder og modstande skal anvendes for at sikre patienten mod afladningsstrømme over 0,5 mA.
- Der skal laves en forforstærker for hver kanal - 3 i alt.
- Forforstærkeren skal have en høj CMRR til frasortering af common mode signaler på indgangen.
- EKG signalet skal forstærkes 20 gange.

#### D.1.2 Valg af forstærkertype

Grundet de store krav til forforstærkerens indgangsimpedans og CMRR, står det klart at en almindelig op-amp konfigureret som differensforstærker ikke er et fornuftigt valg af forstærkertype. Denne konfiguration ville kræve indgangsmodstande på min. 10 M $\Omega$  og en feedbackmodstand på 200 M $\Omega$  for at sikre et gain på 20. Da dette ikke kan implemeteres i praksis, vælges en mere avanceret forstærkertype, instrumentationsforstærkeren, der kombinerer differensforstærkerens egenskaber med et buffertrin, der sikrer meget høje og ens indgangsimpedanser og høj CMRR.

Valget af instrumentationsforstærker er faldet på AD620A fra Analog Devices. Denne forstærker tilbyder de egenskaber der søges, og er desuden velegnet til batteriforsynede applikationer, fordi den ikke bruger særlig meget strøm - blot 1.3 mA ved maksimal forstærkning.

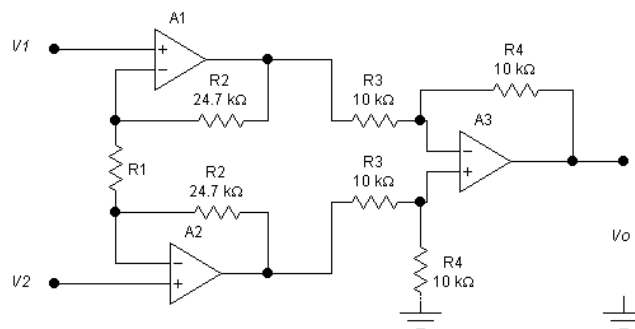
Forstærkeren er i designet konfigureret som det ses af fig. D.1.



**Figur D.1:** Således er den anvendte instrumentationsforstærker konfigureret. Den eksterne modstand  $R_1$  bestemmer forstærkningsgraden.

### D.1.3 Beregning af gain-modstand

En simplificeret skematisk fremstilling af de interne komponenter i forstærkeren er givet på fig. D.2.



**Figur D.2:** Simplificeret layout af interne komponenter i den anvendte instrumentationsforstærker AD620A. Forstærkeren er konfigureret således at det differentielle gain opnås over de inverterende forstærkere A1 og A2.

Output fra den anvendte instrumentationsforstærker er [18],

$$V_o = \frac{R_4}{R_3} \left(1 + \frac{2R_2}{R_1}\right) (V_2 - V_1) = \left(1 + \frac{2 \cdot 24.7K\Omega}{R_1}\right) (V_2 - V_1) \quad (D.1)$$

Det differentielle gain  $G_d$  er således bestemt ved

$$G_d = \left(1 + \frac{2 \cdot 24.7K\Omega}{R_1}\right) \quad (D.2)$$

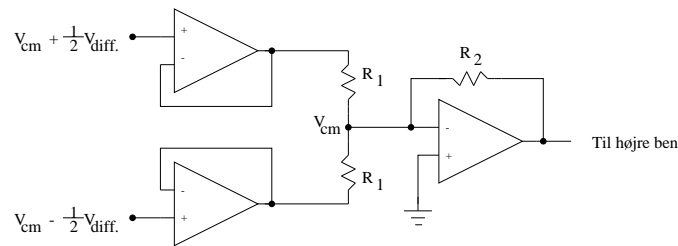
For at opnå et gain på 20 i forforstærkeren skal der altså benyttes en modstand  $R_1$  med størrelsen:

$$R_1 = \frac{49.4K\Omega}{(20 - 1)} = 2,6k\Omega \quad (D.3)$$

### D.1.4 Driven-right-leg kredsløb

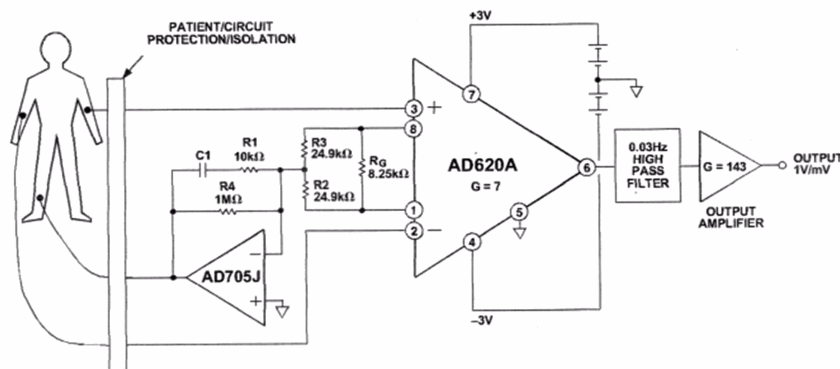
Ifølge [3] og [6] samt diverse applikationsnoter, implementeres et standard driven-right-leg-kredsløb (herefter: DRL) blot som en inverterende forstærker, der forstærker common mode-signalet fra punktet, der opstår ved at forbinde punkterne før de to  $R_3$ -modstande på figur D.2 via to lige store modstande og måle i midtpunktet. Som vist på figur D.3 kan det let vises, at dette

punkt repræsenterer common mode spændingen  $V_{cm}$ , såfremt det antages, at differenssignalet  $V_{dif}$  fordeles sig ligeligt med  $+1/2V_{dif}$  på den ene indgang til forstærkeren og  $-1/2V_{dif}$  på den anden indgang.



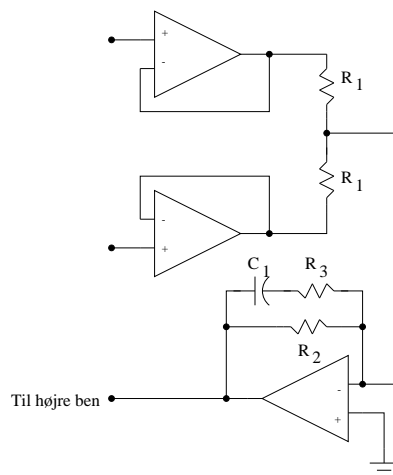
**Figur D.3:** Midtpunktet mellem de to  $R_3$ -modstande repræsenterer  $V_{cm}$ .

Da AD620 er en IC-kreds har vi imidlertid ikke adgang til dette punkt, og på databladet er der derfor angivet en anden konfiguration til implementation af DRL. Denne konfiguration, der er vist på figur D.4, viste sig imidlertid ikke anvendelig i praksis, da punktet mellem  $R_2$  og  $R_3$  på diagrammet ikke repræsenterer  $V_{cm}$ . Såvel ben 1 og 8 på AD620'eren bærer et konstant offset på  $-400$  mV, og punktet mellem  $R_2$  og  $R_3$  antager derfor værdien  $V_{cm} - 400$  mV.



**Figur D.4:** Sådan implementeres et driven-right-leg kredsløb ifølge databladet for den benyttede instrumentationsforstærker AD620.

Som løsning på dette problem oprettes et selvstændigt buffertrin bestående af to voltagefollowers. Som vist på figur D.5 forbindes udgangene på de to op-amps med to lige store modstande, og  $V_{cm}$  kan nu udtages i midtpunktet og føres videre til feedback-kredsløbet, der fra dette punkt er konfigureret som foreslået på databladet for AD620.

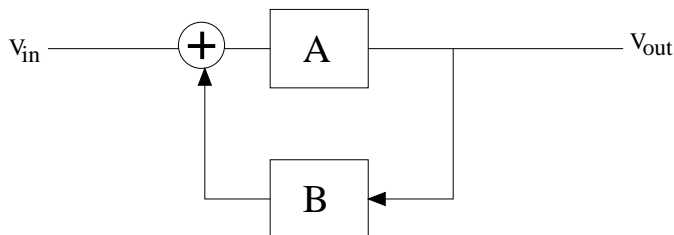


**Figur D.5:** Det designede driven-right-leg-kredsløb med buffertrin, og feedback-kredsløb.

For at sikre stabilitet i DRL-kredsløbet er der indbygget en kondensator C1. I det følgende udledes størrelsen af denne kondensator.

### Stabilitetssikring i driven-right-leg-kredsløbet

DRL-kredsløbet er et feedback-kredsløb med instrumentationstforstærkerens buffertrin som open-loop forstærker med gain A og DRL-op-ampen som feedback-forstærker med gain  $\beta$ . Se figur D.6.



**Figur D.6:** Generel struktur for et feedback-kredsløb. I vores sammenhæng udgøres A-gainet af buffer-trinet og  $\beta$ -gainet af DRL-op-ampen.

Det samlede gain  $A_f$  for et feedback-kredsløb kan ifølge [18] beregnes således:

$$A_f = \frac{A}{1 + A\beta} \tag{D.4}$$

Af dette kan det udledes, at systemet bliver ustabil, når loop-gainet  $A\beta$  antager reelle værdier  $< -1$ . Dette er tilfældet, når den samlede fase for  $A\beta$  er 180 grader, og amplitudeproduktet er  $> 1$ . For at sikre systemets stabilitet, er det nødvendigt at opretholde en vis gain- og fase-margin, der sikrer, at amplitudeproduktet ikke bliver  $> 1$ , mens fasen er 180 grader.

Da det er uden for denne rapports målsætning at redegøre mere udpenslende for tilbagekoblings-teori, forklares den følgende metode til sikring af stabilitet ikke nærmere - den anvendes blot på det nærværende stabilitetsproblem. Der henvises til [Sedra Smith].

Amplitudeplottet for  $1/A$  og  $\beta$  bodeplottes. Hvis de to amplitudeplots skærer hinanden med en hældning på  $-20$  dB/dec, opnås tilstrækkelig gain- og fase-margin, og systemet er stabilt. For at sikre dette indskydes en dominerende pol i feedback-kredsløbet vha. kondensatoren C1. Den dominerende pol sikrer, at knækfrekvensen for feedback-kredsløbet kan placeres uafhængigt af den anvendte op-amps interne kompenstation, og det er dermed muligt, at tvinge amplitudeplottet til en hældning på  $-20$  dB/dec over et større frekvensspektrum.

Feedback-kredsløbets overføringsfunktion kan bestemmes til:

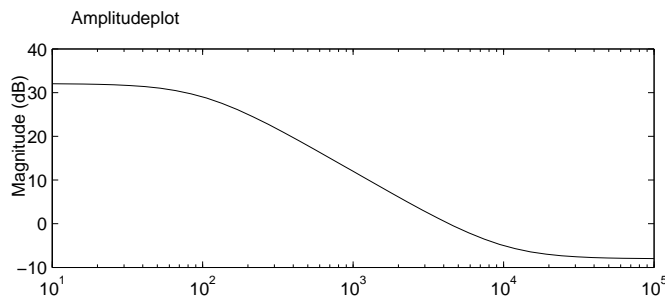
$$A_f = \frac{R_4(R_1sC_1 + 1)}{R_i((R_4 + R_1)sC_1 + 1)} \tag{D.5}$$

Amplitudeplottet er plottet for forskellige værdier af  $C_1$ , og værdien  $C_1 = 10$  nF er fundet passende. Med denne værdi ser amplitudeplottet ud som vist på figur D.7.

Med denne komponentværdi opnås den ønskede effekt af den dominerende pol. Som vist på figur skærer de to amplitudeplots for  $1/A$  og  $\beta$  hinanden med en skæring på  $-20$  dB, og stabiliteten er dermed sikret.

### Sikring af patienten

For at sikre at patienten ikke udsættes for strømme, der er større end 0,5 mA indskydes beskyttende dioder og modstande ved indgangene til forforstærkeren. Dioderne sikrer, at spændingsforskellen mellem 2 elektroder maksimalt bliver  $\pm 0,7$  V. Ifølge [? , kim]r elektrode-/hud-impedansen min. 3 k $\Omega$  ved almindelig brug. Der placeres yderligere 4 k $\Omega$  i serie på indgangene



Figur D.7: Amplitudeplot for feedback-kredsløbet med  $C_1 = 10 \text{ nF}$ .

for at opnå maksimalt 0,1 mA strøm gennem patienten - standarden er dermed overholdt til fulde.

Det samlede design af forforstærkeren fremgår af oversigtsdiagrammet i appendix G.

### D.1.5 Implementation

Det samlede design af forforstærkeren er implementeret på et wrapboard. IC-komponenter er placeret i 16-bens wrapsokler, beskyttelsesdioder er loddet på wrapspyd, modstande og kondensatorer er loddet på komponentplatforme og placeret i tilsvarende wrapsokler. De beskrevne benforbindelser er oprettet ved manuel wrapping. Samtlige IC-komponenter forsynes med  $\pm 9 \text{ V}$  via 2 stk. 9 V-batterier. Der er lavet et stik på boardet til let tilslutning af kabel fra patientens elektroder.

### D.1.6 Test af forforstærker

Der er foretaget tests af forforstærkerdesignet for at undersøge om de krav, der er stillet til designet er overholdt. Ikke alle krav er imidlertid testbare, hvorfor der kun er foretaget test af forstærkningsgraden (gain 20) og en brugstest, hvor det er undersøgt, om driven-right-leg-kredsløbet er stabilt samt om patienten kan udsættes for strømme over 0,5 mA.

### D.1.7 Resultat af test

Som det fremgår af figur ?? er forforstærkerens gain verificeret til at være 20 for alle tre kanaler. Driven-right-leg-kredsløbet virker som ønsket, og det er stabilt. Beskyttelseskomponenterne sikrer som ønsket patienten med strømme, der er større end 0,5 mA. Forforstærkeren lever dermed op til de opstillede krav.





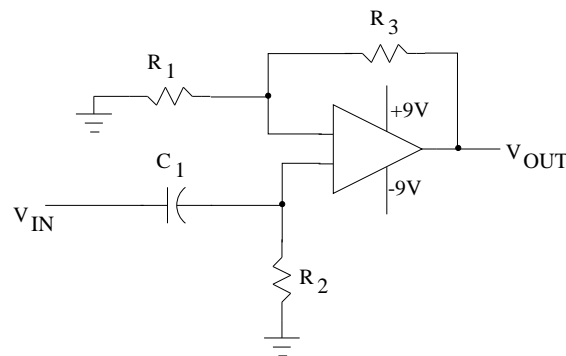
## D.2 Design og implementation af filter

### D.2.1 Krav til filter

- Der skal laves et filter for hver kanal - 3 i alt.
- Filtrering af EKG-signalet skal bestå af en højpasfiltrering og en lavpasfiltrering.
- Frekvenser under 0,5 Hz skal dæmpes med 20 dB/decade for at undgå baseline drift.
- Frekvenser over 150 Hz skal dæmpes med minimum 59 dB. Lavpasfilteret skal designes således at transitionsbåndforholdet minimerer den nødvendige samplingsfrekvens til 500 Hz.
- Forstærkertrinet skal forstærke EKG-signalet 50 gange.
- Der skal sikres imod negative spændinger til ADC'en.
- Der skal sikres imod spændinger over 3,3 V til ADC'en.

### D.2.2 Højpasfilter

For at fjerne DC offset og forstærke EKG signalet, anvendes et simpelt 1. ordens aktivt højpasfilter. Filteret konfigureres som det ses på fig. D.8



Figur D.8: Højpasfilteret

Overføringsfunktion for filteret er:

$$H(j\omega) = \frac{V_o}{V_i} = \frac{(1 + \frac{R_3}{R_2})}{1 + \frac{1}{j\omega R_1 C_1}} \quad (D.6)$$

Den ønskede knækfrekvens er 0,5 Hz og den ønskede forstærkning er 50. Komponent værdier vælges til:  $R_1 = 1 \text{ K}\Omega$ ,  $C_1 = 1 \mu\text{F}$ ,  $R_2 = 330\text{K}\Omega$ ,  $R_3 = 49\text{K}\Omega$ .

Knækfrekvens og gain for højpasfilteret bliver således:

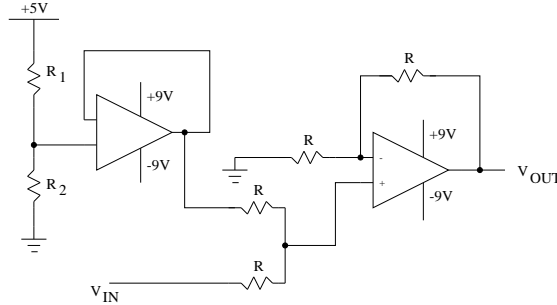
$$f_c = \frac{1}{2\pi R_1 C_1} = \frac{1}{2\pi 330\text{K}\Omega 1\mu\text{F}} = 0.48\text{Hz} \quad (D.7)$$

$$\text{Gain} = (1 + \frac{R_3}{R_2}) = 1 + \frac{49\text{K}\Omega}{1\text{K}\Omega} = 50 \quad (D.8)$$

$$\text{Gain(dB)} = 20\log 50 \approx 34\text{dB} \quad (D.9)$$

### D.2.3 Summer

EKG signalet skal hæves med 0,66V og til dette formål benyttes en ikke inverterende summer konfigureret som det ses af figur D.9



Figur D.9: Summer

Offsetjustering skal være konstant, så der vælges at benytte en 5V spændingsregulator som kilde til spændingsdeler sektionen. Dette sikrer mod baseline drift, når batterierne aflades. Spændingen på 0.66V påtrykkes gennem en buffer til summeren og signalet hæves ifølge:

$$V_o = \left(1 + \frac{R}{R}\right) \left(\frac{R \parallel R}{R} V_{EKG} + \frac{R \parallel R}{R} V_{0,66}\right) \quad (\text{D.10})$$

Komponentværdier er valgt til:  $R = 100K\Omega$ ,  $R_1 = 300K\Omega$ ,  $R_2 = 45,6K\Omega$ . Således fås output fra summeren til:

$$V_o = (1 + 1) \left(\frac{1}{2}\right) (V_{EKG} + V_{0,66}) = V_{EKG} + V_{0,66} \quad (\text{D.11})$$

### D.2.4 Lavpasfilter

Lavpasfilterets knækfrekvens skal ligge omkring 150 Hz, og samtidig skal signaler mellem knækfrekvensen og Nyquist frekvensen dæmpes med 59 dB. For at minimere samplingsfrekvensen er det besluttet at anvende et let implementerbart 8. ordens switched capacitor filter med en dæmpning på 60 dB efter  $1,2 \cdot f_c$ . Filterets knækfrekvens kan bestemmes af en oscillator clock således:

$$f_c = \frac{f_{clk}}{100} \text{ Hz} \quad (\text{D.12})$$

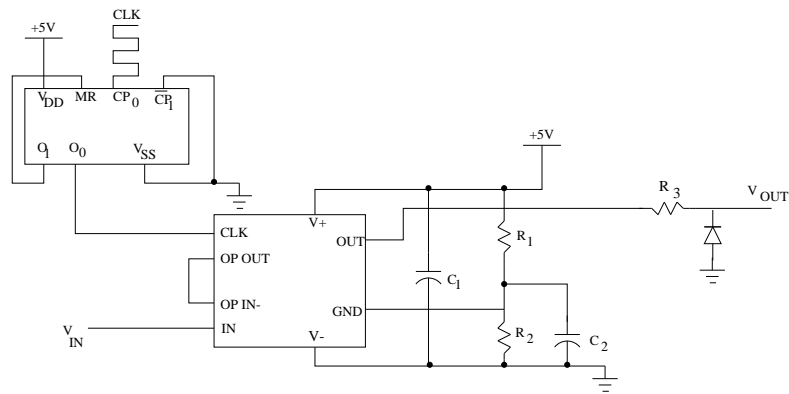
Den eksisterende oscillator fra mikrokontrolleren på 32.768Hz kan tilgås via et eksternt ben. Denne kan efterfølgende skaleres ned gennem en divide by n enhed og benyttes som clock til filteret. Til dette formål anvendes en decade counter som konfigureres til at dele med 2. Knækfrekvensen for filteret bliver da:

$$f_c = \frac{32.768 \text{ Hz}}{200} = 163,8 \text{ Hz} \quad (\text{D.13})$$

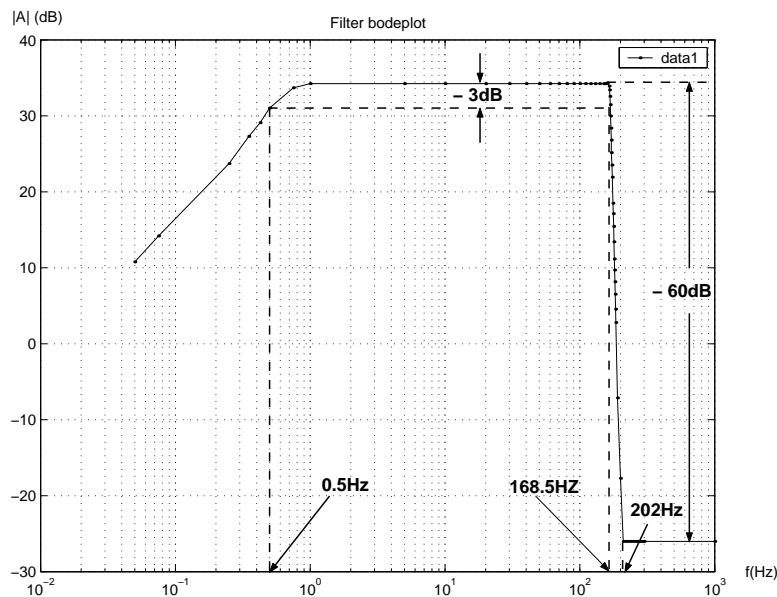
På grund af filterets transitionsbånds forhold  $R_t = \frac{f_s}{f_c} = 1.2$  opnås en dæmpning på 60 dB allerede ved 197 Hz.

Ved at benytte dette filter opnås en god udnyttelse af EKG'ets frekvensbånd samtidig med, at den nødvendige samplingsfrekvens ligger på et forholdsvis lavt niveau omkring 400 Hz, så den valgte samplingsfrekvens er fuldt ud tilstrækkelig.

For at sikre AD konverteren imod negative spændinger forsynes det anvendte SC filter med 0-5V. For at sikre AD konverteren imod spændinger over den øvre forsyningspænding på 3,3V anvendes en 3,3 V-zenerdiode på udgangen. Lavpasfilteret konfigureres sammen med decade counteren og zenerdioden på udgangen som det ses af fig. D.10



**Figur D.10:** Lavpasfilteret konfigureres således at den oscillator som sidder på mikrokontrolleren kan anvendes til at bestemme knæfrekvensen. Oscillatoren nedskaleres gennem en divide-by-n enhed og anvendes som clock på lavpasfilteret. AD konverteren er sikret imod uønskede spændingsniveauer ved at forsyne lavpasfilteret med 0-5V og anvende en diode på udgangen



**Figur D.11:** Filterets frekvensrespons er illustreret på figuren. Transitionsbåndsforholdet sikrer at den nødvendige samplingsfrekvens er under 500 Hz

### D.3 Test af filter

Der er foretaget tests for at se om filteret opfylder de i kravspecifikationen opstillede krav. Forstærkningen er testet på alle tre kanaler med en 50 mV p-p sinusbølge fra en signalgenerator. Amplituden svarer til det forventede input fra forforstærkeren. Forstærkningen målt på udgangen fra filteret var 50 fra alle tre kanaler og lever dermed op til det opstillede krav. Det blev konstateret at ingen negative inputspændinger kunne måles på udgangen fra filteret. Ligeledes blev spændinger over 3,3 V klippet og ADC'en er således sikret mod negative spændinger og spændinger over 3,3 V. kravspecifikationen er opfyldt på dette punkt.

Frekvensresponsen for filteret er undersøgt og resultatet plottet på fig. D.11. Den nedre knæfrekvens for højpasfilteret blev målt til 0,5 Hz og den øvre knæfrekvens til 169 Hz. Frekvenser over 200 Hz dæmpes med 60 dB.

Aliasering kan undgås ved at sample med minimum 400 Hz og den nødvendige samplingsfrekvens er derved optimeret til under 500 Hz som angivet i kravspecifikationen. Filteret virker efter hensigten og implementeres med konfigurationen som vist på i appendiks G.



## Appendiks E

# Hardware specifikation - Seriel kommunikation

### E.1 Design og implementation for seriel kommunikation

#### E.1.1 Krav til seriel kommunikation

- Spændingsniveauer skal konverteres som i tabel E.1:

Signaloprindelse og størrelse	Konverteres til
mc, 0-0,6 V	pc, 5-15 V
mc, 2,7-3,3 V	pc, -15- -5 V
pc, 5-15 V	mc, 0-0,6 V
pc, -15- -5V	mc, 2,7-3,3 V

*Tabel E.1: Konvertering mellem spændingsniveauer i kommunikationsenhed*

- Forbindelsen skal indeholde 3 ledninger: transmit, receive og signal ground
- Transmitben på afsender forbindes til receiveben på modtager og omvendt
- Der skal forsynes med batterispændinger på 0-9 V
- Der skal strømisoleres, således at patientsiden er isoleret fra pc-siden
- Der skal anvendes DB-9-stik i begge ender

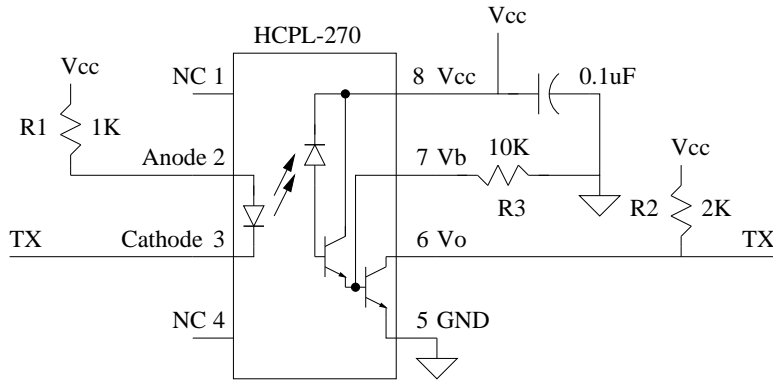
#### E.1.2 Design

Til den galvaniske adskillelse anvendes en optokobler. Denne anvendes, da den er ideel til isolering, ved et digitalt signal. Optokobleren placeres tættest på mc'en, da 0-3,3 V er lettere at anvende som forsyning, end  $\pm 6$  V. således behøves kun en positiv forsyningsspænding. Den anvendte optokobler er af typen HCPL-270L.

For at optokobleren skal virke som ønsket, konfigureres den med to strømbegrænsende pull up-modstande R1, R2 se figur E.1. Modstanden R1 forbindes til anoden på lysdioden, der har signalet på katoden således, at spændinger i intervallet 0 V (logisk 0) giver lys i lysdioden i optokobleren. Spændinger i intervallet 3,3V ikke giver lys. R2 forbindes til en forsyningsspænding og til collectorbenet på én af transistorerne internt i optokobleren. Den anvendte transistor leder ikke strøm når der ikke er lys i lysdioden, dvs. når signalet er logisk 1.

Når transistoren ikke leder strøm, løber der heller ingen strøm gennem modstanden, og output-signalet fra isoleringsbarrieren er på niveau med forsyningsspændingen - logisk 1. Der benyttes en afladningskondensator mellem forsyningsbenet til optokoblerens output og jord, samt en strømregulerende modstand R3, på indgang Vb.

Der skal bruges 2 optokoblere. Én til transmit og én til receive, i hht. UTxD0, og URxD0. Disse overfører data til hver sin side af barrieren, og skal have 2 forskellige forsyninger, på hver side af barrieren.



Figur E.1: Opbygning og konfigurering af optokobler for TX.

### Serial transeiver

For at anvende RS-232-C er det nødvendigt med en form for konvertering mellem spændingsniveauer fordi RS-232-C foreskriver logisk 1 som en negativ spænding, og logisk 0 som en positiv spænding. Dette virker fint på pc'en, men mc'en anvender derimod som logisk 0, spændinger i den laveste ende af spændingsintervallet 0-3,3 V, og som logisk 1 spændinger i den højeste ende. Derfor anvendes en transeiver som varetager konverteringen mellem spændingsniveauerne. Der anvendes en transeiver af typen MAX3223, og denne skal i følge databladet vha. kondensatorer og forsyningsspænding konfigureres til de spændingsniveauer, der skal konverteres mellem. Forsyningsspændingen skal have den værdi, den ikke-negative spænding på mc-siden højst skal antage, dvs. 3,3V.

Af databladet for MAX3223 ses det, at der med en sådan forsyningsspænding skal anvendes 0,1µF som kondensatorværdier, og det er ligeledes opgivet, hvordan disse kondensatorer skal forbindes til MAX3223. Kondensatorerne fungerer som spændingskilde, så det er muligt at opnå højere outputværdier på pc-siden, end der forsynes med.

### E.1.3 Forsyning

Der anvendes 9V-batterier koblet sammen med spændingsomdanner som strømforsyning. Spændingsomdanner regulerer de 9 V til 3,3 V og holder forsyningsspændingen konstant. Spændingsomdanneren er af typen LE-33.

## E.2 Testspecifikationer for serial kommunikation

Før test af af den samlede serielle hardware skal hver del testes. Dvs. serielkabel, serielport på computer, MAX3223, optocoupler samt forsyning i form af batterier og spændingsomdanner.

## E.2.1 Individuelle tests

### Kabeltest

Kablet skal testes ved at forbinde det som et nul-modem-kabel til computerens serielport, sende en meddelelse gennem det og undersøge, om den modtagne meddelelse svarer til den afsendte. Ved et nul-modem-kabel kommer et signal afsendt på transmitter-benet i den serielle port tilbage på receiver-benet i samme serielle port.

### MAX3223-test

MAX3223 skal testes for konvertering, både fra mc til pc-niveauer og modsat. Førstnævnte konvertering skal testes ved på mc-siden, at anvende en funktionsgenerator og på pc-siden, at måle med et oscilloskop om der konverteres korrekt. På samme måde skal testes for pc-mc-konvertering. I testen skal anvendes den testede spændingsforsyning eller en kilde forbundet til lysnettet.

### Optocoupler-test

Det skal testes, at der ikke er fysisk forbindelse mellem de to sider af optokobleren. Dette skal gøres ved at sende et signal ind på den ene side af optokobleren og på den anden side at måle, at det ikke har indflydelse på outputtet, hvad spændingen på inputsiden er. I testen skal anvendes den testede spændingsforsyning eller en kilde forbundet til lysnettet.

### Spændingsforsynings-test

Det skal testes, at der til enhver tid (og spænding på batteriet) leveres præcist 3,3 V som output fra spændingsomdanneren. Dette måles med et multimeter.

## E.2.2 Test af samlet hardware til seriel kommunikation

Med den samlede hardware forbundet til serielporten på computeren, skal det testes at der kan sendes og modtages fx tekstmeddelelser over forbindelsen. Dette gøres ved at loop-transmitte og receive til mc'en, så det bliver sendt tilbage til pc'en.

## E.3 Test

Dette afsnit beskriver tests udført på den serielle hardware mellem microcontrolleren og serielporten på pc'en i henhold til de førnævnte testspecifikationer.

### E.3.1 Spændingsforsynings-test

Der er foretaget test for at sikre at både  $V_{cc}$  og den isolerede  $V_{cc}$  forsyning er korrekte. Testen resulterede i at den målte spænding ved anvendelse af et multimeter, viste at forsyningerne leverede +3.3 V.



### E.3.2 Individuel test: Tx/Rx-optokobler

Tests for galvanisk adskillelse mellem mc og MAXIM3223/RS-232. Ved tests kontrolleres der ved at måle potentialet mellem henholdsvis den isolerede ground og udgang for optokobleren, samt ground for  $V_{cc}$  og optokoblerens indgang. Dette gøres med et indgangssignal på 3,3 V. Først testes for Tx-optokobleren, ved at teste i forhold til den isolerede ground. Dernæst i forhold til  $V_{cc}$  ground. For Tx-optokobleren testes indgang som ben 3 på HCPL-270L, og den isolerede ground testes som ben 2 på LM33 for den isolerede forsyning. Det målte potentiale var mindre end +0,001 V. For Tx-optokobleren testes udgang som ben 6 på HCPL-270L og  $V_{cc}$  ground testes som ben 2 på LM33 for  $V_{cc}$  forsyningen. Det målte potentiale var -0,002 V.

Begge potentialer indikerer at mc'en er galvanisk afskilt fra MAXIM3223/RS-232, idet der ingen potentialer er.

Der testes for Rx-optokobleren ud fra samme test principper som for Tx-optokobleren. For Rx-optokobleren testes ben 6 på HCPL-270L og den isolerede ground testes som ben 2 på LM33 for den isolerede forsyning. Det målte potentiale var betydeligt mindre end +0,146 V. For Rx-optokobleren testes udgang som ben 3 på HCPL-270L og  $V_{cc}$  ground testes som ben 2 på LM33 for  $V_{cc}$  forsyningen. Det målte potentiale var +1,6 V.

Begge potentialer indikerer at mc'en er galvanisk afskilt fra MAXIM3223/RS-232.

### E.3.3 Individuelle tests: MAXIM3223

Tests er foretaget for henholdsvis måling af output på MAXIM3223's ben 17, med et simuleret Tx-input på +3.3 V på ben 13, samt måling af output på ben 15 med et simuleret input fra RS-232 på  $\pm 10$  V på ben 16.

Microcontroller/Optokobler  $\rightarrow$  MAXIM3223: +3,3 V  $\rightarrow$   $\pm 6,2$  V. MAXIM3223  $\rightarrow$  Microcontroller/Optokobler:  $\pm 10$  V  $\rightarrow$  +3,67 V.

Testen viser det ønskede resultat, hvor en  $\pm$  spænding omformes til en positiv spænding der kan godtages af microcontrolleren, og et positivt signal fra mc'en konverteres til en  $\pm$  spænding som kan anvendes af PC'en.

### E.3.4 Individuelle tests: Kabeltest

Et nul-modem's kabel til kommunikationen mellem den serielle hardware og RS-232 porten er testet ved at loope transmitter og receiver benene i hun-stikket til den serielle port. Testen viste, at et udsendt signal i kablet blev modtaget igen.

### E.3.5 Samlet test

Her udføres først test ved samlingen af den serielle hardware, hvorefter en test over seriel porten foretages.

#### Test ved samling

Ved test af samlingen testes der for, om et inputsignal på +3,3 V, der sendes ind, giver et tilfredsstillende udgangssignal ved receiver benet på kabel-hunstikket. Desuden testes det hvorvidt et  $\pm 10$  V signal fra transmitter benet på hunstikket, resulterer i et input til mc'en på +3,3 V.

Resultatet for testen viste at et inputsignal på +3,3 V gav et outputsignal på  $\pm 6,2$  V, hvilket også blev vist ved testen af MAXIM3223. Samtidig viste testen, at et inputsignal fra RS-232-porten resulterede i en spænding på +3,67 V, hvilket er acceptabelt for mc'en.

**Test ved seriel kommunikation**

Ved den endelige test, blev Tx og Rx for optokoblerne loop'et, hvorved test på occiloscop, samt kommunikation over terminalprogrammet COMMO, skulle vise om kommunikation over den serielle hardware var muligt. Alle taster blev afprøvet, hvor enkelte taster blev holdt nede i længere tid. Testen viste at kommunikation op til og med BAUD-Rate på 57K forløb uden problemer. Derimod var der fejl i tegnene, når BAUD-Rate kom op på 115K. Optokobleren kunne ikke følge med, hvilket resulterede i, at det der kom tilbage var ikke genkendeligt.



## Appendiks F

# Interfacing af LCD-display

### F.1 Displayets ben og funktioner

Displayet har 14 eksterne ben hvoraf 3 ben bruges til at forsyning og kontraststyring. De resterende 11 ben forbindes til mc'en og benyttes til kommunikation mellem de to enheder. Benfunktionerne for displayet er illustreret på figur F.1 og er forklaret nedenfor.

Ben	Navn	Funktion
1	Vss	Gnd
2	Vdd	5V forsyning
3	Vlc	Kontraststyring
4	RS	L: instruktion H: data
5	R/W	L: skriv til lcd H: læs fra lcd
6	E	Enable
7	DB0	Data
8	DB1	Data
9	DB2	Data
10	DB3	Data
11	DB4	Data
12	DB5	Data
13	DB6	Data
14	DB7	Data

*Figur F.1: Displayets benkonfiguration*

Displayet kontroller består af to 8-bit registre, et instruktionsregister og et dataregister. Instruktionsregisteret gemmer informationskoder som display clear, cursor skifte og adresse information. Kommunikation med dette register er nødvendigt under initialisering af displayet, hvor den grundlæggende opsætning af ønskede funktionaliteter foregår.

Dataregisteret gemmer midlertidigt overført data fra mikrokontrolleren og skriver det efterfølgende ind i display data RAM (DDRAM), så det kan vises på displayet. Der kan i modsætning til instruktionsregisteret, hvor der kun kan skrives til, både skrives til og læses fra dataregisteret. Adgang til displayets registre bestemmes af ben 4, register select (RS). Når RS er lav kan der skrives til instruktionsregisteret og når RS er høj er dataregisteret aktivt. Ben 5, read/write (R/W) holdes lav når der skrives til instruktionsregisteret eller dataregisteret. Desuden kan R/W benyttes til at læse et busyflag (BF) fra displayets ben 14 (DB7), som indikerer om displayet er optaget og ikke kan modtage en ny kommando.

Ben 6, enable (E) benyttes til at lache data til og fra displayet og ben 7-14 (DB0-DB7) benyttes under skrivning til at instruere displayet om hvilken karakter der ønskes skrevet og hvor den ønskes placeret.

Mc'ens ben P4.0-P4.7 benyttes til display ben DB0-DB7. P5.0-P5.2 benyttes til RS, R/W, E.



## **Appendiks G**

# **Konfiguration - Eksterne kredsløb**



**DEL**  
**VII**  
**BILAG A**