



## AARHUS SCHOOL OF ENGINEERING

### SUNDHEDSTEKNOLOGI 3. SEMESTERPROJEKT

---

# Rapport

---

#### *Gruppe 2*

Anne Bundgaard Hoelgaard	(201404492)
Mette Hammer Nielsen-Kudsk	(201408391)
Ditte Heebøll Callesen	(201408392)
Martin Banasik	(201408398)
Albert Jakob Fredshavn	(201408425)
Johan Mathias Munk	(201408450)

#### *Vejleder:*

Studertervejleder  
Peter Johansen  
Aarhus Universitet

16. december 2015

# Resumé

---

Dette projekt beskæftiger sig med blodtryksmåling, hvor der ud fra et blodtrykssignal kan bestemmes en systolisk- og diastolisk værdi, samt en pulsværdi. Formålet er at bygge en hardwaredel, der kan trykændre et analogt signal. Via et filter skal signalet forstærkes og sendes videre igennem en Data Acquisition (DAQ), der kan lave det analoge signal om til et digitalt signal. Herefter skal signalet videre ind i softwaren. Her er formålet at få programmeret en algoritme, der kan detektere en puls samt en systolisk- og diastolisk værdi. Disse systole- og diastoleværdier kan findes ud fra det digitale signal. Signalet bliver observeret fra starttidspunktet, hvor værdien for trykket begynder at vokse, indtil grafen når maksimum. Når værdien igen begynder, kan programmet detektere den højeste værdi, som er den systoliske værdi. Efter maksimum begynder grafen for signalet at falde igen, da trykket bliver mindre, indtil det når det laveste punkt, hvor værdien igen begynder at stige en smule. Dette lavpunkt er værdien for diastolen. Grafen er periodisk og ud fra den systoliske- og diastoliske værdi kan pulsen findes.

Herudover skal programmet kunne kalibreres og nulpunktsjusteres når forskeren synes nødvendigt. Projektet er lavet til forskningsbrug og derfor blev det vurderet, at det er nødvendigt for en forsker, at kunne gemme sine målinger. Programmeringen skal derfor også gøre det muligt for forskeren, at kunne gemme sine målinger i en database.

Resultatet for dette projekt viser at det er muligt vha. hardware- og softwaredelen, at måle et blodtryk. Gennem videreudvikling kan det blive muligt at detektere hypertension (Forhøjet blodtryk) og hypotension (Lavt blodtryk) i samme program. Dette ville være nyttigt, hvis projektet i fremtiden skulle bruges til patienter. I projektet er der dog udelukkende blevet arbejdet med måling af blodtryk.

# Abstract

---

This project has primarily been focused on blood pressure measurement, where systolic- and diastolic values, as well as the pulse, could be determined. The purpose of this project is to construct hardware and software which can manipulate the pressure of an analog signal as well as depict this signal using developed software. By using an analog filter, the analog signal is amplified and sent to a Data Acquisition (DAQ), where the signal will be transformed from an analog signal to a digital signal. This digital signal will afterwards enter a computer, running the software program. Here the purpose of the software is to program an algorithm, which can detect a pulse as well as the systolic- and diastolic values. These two values are found by looking at the visualized signal. From the beginning of the graph, and until the signal ascends to maximum and falls again. This maximum value is the systolic value. Afterwards the signal will begin to descend until a minimum is reached, where this minimum is the diastolic value. The graph will then slightly climb again and begin a new period. From these systolic- and the diastolic values, the pulse can be found. In addition to the above requirements, the program is able to be calibrated and zero adjusted, whenever the researcher finds it necessary.

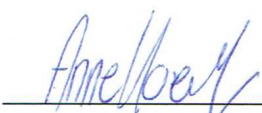
This project is made for research use and it was therefore decided that a necessity for a researcher, would be to save his measurements in a database.


The result of this project shows that it is possible to measure blood pressure through a combination of hardware and software. Through future development it could be possible for the program to detect hypertension (high blood pressure) and hypotension (low blood pressure). This would be useful if the project, in the future, would be used on patients. In this project however, the main goal has exclusively been measurements of blood pressure.

# Underskrifter

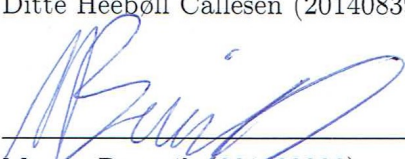
---

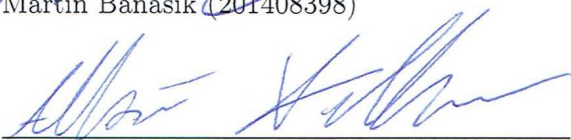
## Gruppemedlemmer

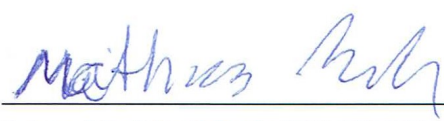
  
Anne Bundgaard Hoelgaard (201404492) 16/12-2015  
Dato

  
Mette Hammer Nielsen-Kudsk (201408391) 16/12-2015  
Dato

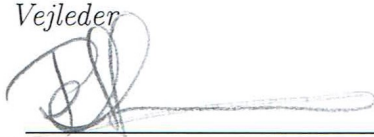
  
Ditte Heebøll Callesen (201408392) 16/12-2015  
Dato

  
Martin Banasik (201408398) 16/12-15  
Dato

  
Albert Jakob Fredshavn (201408425) 16/12-2015  
Dato

  
Johan Mathias Munk (201408450) 16/12-15  
Dato

## Vejleder

  
Peter Johansen 16/12-15  
Dato

# Godkendelsesformular

---

## Godkendelsesformular

Antal sider: 40

Forfattere:



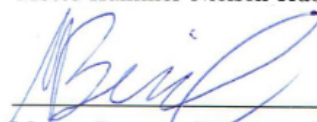
Anne Bundgaard Hoelgaard (201404492)



Mette Hammer Nielsen-Kudsk (201408391)



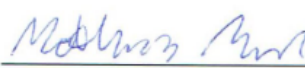
Ditte Heebøll Callesen (201408392)



Martin Banasik (201408398)



Albert Jakob Fredshavn (201408425)



Johan Mathias Munk (201408450)

Godkendes af Peter Johansen

Kunde Aarhus Universitet

Ved underskrivelse af dette dokument accepteres det af begge parter som værende kravene til udviklingen af det ønskede system.

Dato: 16/12-2015



Kundens underskrift



Leverandørens underskrift

# Ordliste

---

Ord	Forklaring
(F)URPS+	Et akronym, der repræsenterer en model til klassificering af softwarens kvalitet
GUI	Graphical User Interface (Grafisk brugergrænseflade)
VPN	Virtual Private Network
DAQ	Data acquisition
SysML	Systems Modeling Language – sprog til visuel fremstilling af systemer
UML	Unified modelling language – sprog til oversigtsfremstilling af klasser i programmering
KSS	Kommunikation og Samarbejde i Sundhedsvæsenet
BDD	Block Definition Diagram
IBD	Intern Block Diagram
SD	Sekvensdiagram
UML	Unified Modeling Language
ISE	Indledende System Engineering
KVI	Kardiovaskulær instrumentering
Hjerteinsufficiens	Hjertesvigt
DSB	Digital Signalbehandling
Hypertension	Forhøjet blodtryk
Hypotension	Lavt blodtryk
MTTR	Mean Time To Restore
MTBF	Mean Time Between Failure
Anatomi	En organismes indre og ydre opbygning
Veroboard	Printplade med isolerede kobberører
CMRR	Common Mode Rejection Ratio

# Indholdsfortegnelse

---

Resumé	i
Abstract	ii
Underskrifter	iii
Godkendelsesformular	iv
Ordliste	v
Kapitel 1 Indledning	1
Kapitel 2 Projektformulering og afgrænsning	3
Kapitel 3 Blodtryk	6
3.1 Hvad betyder blodtryk? . . . . .	6
3.2 Hypertension . . . . .	7
3.3 Hypotension . . . . .	8
Kapitel 4 Systembeskrivelse	9
Kapitel 5 Krav	11
Kapitel 6 Projektbeskrivelse	14
6.1 Projektgennemførelse . . . . .	14
6.2 Projektstyring . . . . .	16
6.3 Metoder . . . . .	16
6.4 Systemarkitektur . . . . .	17
6.5 Hardware . . . . .	18
6.5.1 Design) . . . . .	18
6.5.2 Implementering . . . . .	18
6.5.3 Test . . . . .	21
6.6 Software . . . . .	25
6.6.1 Design . . . . .	25
6.6.2 Implementering . . . . .	27
6.6.3 Test . . . . .	33
6.6.4 Integrationstest . . . . .	33
6.7 Resultater og diskussion . . . . .	33
6.8 Udviklingsværktøjer . . . . .	35
6.9 Opnåede erfaringer . . . . .	35
6.10 Perspektivering - Fremtidigt arbejde . . . . .	36
Kapitel 7 Konklusion	38

Indholdsfortegnelse	ASE
<b>Kapitel 8 Bilag</b>	<b>39</b>
<b>Figurer</b>	<b>40</b>



# Indledning 1

---

Rundt om i hele verden, måles der i dag blodtryk. Hvorfor? Fordi blodtrykket kan fortælle utrolig meget omkring modstanden i en persons blodårer, hvilket kan være virkelig vigtigt, da forsnævret, sammentrukne eller overforkalket årer kan føre til hypertension, der kan føre til f.eks. akut myokardieinfarkt. Der forskes nu til dages meget inden for blodtryk og hvad blodtrykket egentlig kan fortælle noget om. I forbindelse med en forskers forskning, udarbejdes der i dette semesterprojekt en blodtryksmåler til forskningsbrug. Ved et blodtrykssignal kan der detekteres systoliske- og diastoliske værdier, der som baggrund kan benyttes af forskeren til at analysere på blodtrykket. Formålet med dette projekt er at hjælpe forskningen inden for blodtryk.

Det bliver valgt at denne blodtryksmåler skal kunne modtage en spænding fra en transducer, nulpunktsjustere og kalibere efter forskerens ønske. Signalet skal vises i en graf, på et display, hvor værdier for puls, systoliske- og diastoliske tryk vises. Her starter og gemmer forskeren sine målinger.

I kravspecifikationen findes de krav, der er blevet stillet for projektet. Herunder er også de krav, som er blevet stillet mellem os og vejleder. Under systemarkitekturen findes informationer om, hvordan software- og hardwaredelen er opbygget. I afsnittet integrationstest kan der læses om, hvordan projektet er blevet testet.

Projektet har overvejende anvendt viden fra kurserne Sundhedsvidenskab, Kardiovaskulær instrumentering, Programmering, Analog signalbehandling, Digital signalbehandling og Indledende System Engineering fra Ingeniørhøjskolen på Århus Universitet. Sidstnævnte kursus har især været baggrund for arbejdsprocesserne og arbejdsmetoderne, som også kommer til udtryk i rapporten i form af diverse diagrammer og procesbeskrivelser.

Dette projekt består af to dele - en projektrapport og en projektdokumentation. Projektrapporten (dette dokument) giver indblik i udarbejdningsprocessen af projektet og består af problemformulering, systembeskrivelse, anatomi, konklusion m.m. Her bliver projektvalg og -erfaringer omkring styrker og svagheder i arbejdsprocessen beskrevet. Projektdokumentationen giver indblik i baggrunden for og tilblivelsen af projektet og slutresultatet, samt dokumentationen for arbejdsprocessen.

**Versionshistorik**

Version	Dato	Ansvarlig	Beskrivelse
0.1	04-11-2015	MH NK	Oprettelse af L <sup>A</sup> T <sub>E</sub> Xdokumenter
0.2	11-11-2015	ABH	Udviklingsværktøjer og krav
0.3	11-11-2015	DHC	Metoder
0.4	18-11-2015	ABH	Systembeskrivelse
0.5	19-11-2015	DHC	Perspektivering - Fremtidigt arbejde
0.6	24-11-2015	MH NK	Blodtryk
0.7	24-11-2015	AJF	Projektgennemførelse og -styring
0.8	24-11-2015	JMM	Projektformulering og afgrænsning
1.0	01-12-2015	MBA, MH NK	Referencelister i L <sup>A</sup> T <sub>E</sub> X
1.1	02-12-2015	MBA, MH NK	Figurlister i L <sup>A</sup> T <sub>E</sub> X
1.2	02-12-2015	MH NK	Resume
1.3	07-12-2015	MH NK	Abstract
1.4	09-12-2015	MH NK	Konklusion
1.5	11-12-2015	ABH	Tilrette krav og systembeskrivelse ift. endelig løsning
1.6	13-12-2015	MH NK	Indledning
1.7	13-12-2015	MBA	Opnåede resultater
1.8	13-12-2015	MH NK	Resultater og diskussion

# Projektformulering og afgrænsning 2

---

I daglig klinisk praksis er der ofte behov for kontinuert at monitorere patienters blodtryk, i særdeleshed på intensive afdelinger samt operationsstuer, hvor blodtrykket er en vigtig parameter i monitorering af patienters kardiovaskulære status.

Denne kontinuerte monitorering er også nødvendig i forskningsverdenen. Det er i forskerens interesse, at kunne måle blodtrykket når der laves hæmodynamiske undersøgelser. Her skal det være muligt for forskeren at kunne aflæse det diastoliske- og systoliske tryk, pulsen samt få vist en pæn kurve over blodtrykket. Det er målet, at opbygge en prototype, der kan registrere de spændinger, i milivolt (mV), der kommer fra tryktransducere og analogt forstærke samt filtrere signalet. Dette signal skal derefter konverteres til det digitale domæne.

Herfra skal der programmeres en brugergrænseflade, der fremfører disse målinger, samt gøre det muligt, for forskeren, at gemme målingen i en database til senere brug. Resultatet bliver derfor et elektronisk kredsløb med forbindelse til et software program. For at de gemte data kan sammenlignes, kræver det at de alle er blevet gemt med samme forudsætning, dvs. at målingerne er blevet kalibreret og nulpunktsjusteret. Dette bliver håndteret i softwaredelen, hvor beregninger implementeres. Når forskeren kigger på blodtryksgrafen, vil han normalt se på et filtreret signal. I tilfælde af at det er i hans interesse at se på et ufiltreret signal, vil dette være muligt, ved et tryk på en knap.

## MoSCoW <sup>1</sup>

### *Must*

- Et elektronisk kredsløb, som forstærker signalet fra tryktransducere og filtrerer det med ét indbygget analogt filter
- Et program til at vise blodtrykket som funktion af tiden. Programmet skal opfylde en række obligatoriske krav. Det skal kunne:
  - Programmeres i C#
  - Kunne kalibrere blodtrykssignalet og foretage en nulpunktsjustering
  - Vise blodtrykssignalet kontinuert
  - Kunne gemme de målte data i en database

---

<sup>1</sup>Metode, der benyttes til at beskrive hvad programmet skal kunne og det der ønskes, men ikke vil ske  
- Must have, Should have, Could have, og Would like but won't get

- Kunne filtrere blodtrykket i selve programmet via et digitalt filter, som skal kunne slås til og fra (monitor mode = filtreret og afrundet; signaldiagnose mode = rå signal med alle udsving)
- Afbildning af systolisk/diastolisk blodtryk med tal

*Should*

*Could*

- Hardwaredelen skal bestå af ét Veroboard med indbyggede komponenter

*Would*

- Alarmering, hvis blodtrykket afviger indbyggede grænseværdier
- Forskeren skal kunne hente de gemte data ned igen

### Ansvarsområder

Idet gruppens størrelse ikke lægger op til samlet, at arbejde på alle dele samtidig, er projektets ansvarsområder blevet fordelt som følgende:

Navn	Ansvarsområder
Ditte Heebøll Callesen	Hardwaredesign, dokumentation
Albert Jakob Fredshavn	Hardwaredesign, dokumentation
Martin Banasik	Hardwaredesign, dokumentation
Johan Mathias Munk	Softwaredesign, algoritmeopbygning, dokumentation
Mette Hammer Nielsen-Kudsk	Softwaredesign, algoritmeopbygning, dokumentation
Anne Hoelgaard	Softwaredesign, algoritmeopbygning, dokumentation
Navn	Skrevet afsnit
Ditte Heebøll Callesen	Metode, systemarkitektur (hardware) og perspektivering
Albert Jakob Fredshavn	Projektgennemførelse og projektstyring
Martin Banasik	Systemarkitektur (hardware) og opnåede resultater
Johan Mathias Munk	Projektformulering, afgrænsning og systemarkitektur (software)
Mette Hammer Nielsen-Kudsk	Resumé, abstract, indledning, blodtryk, resultater, diskussion

Anne Hoelgaard

og konklusion

Systembeskrivelse, Krav, Systemarkitektur (software)

og udviklingsværktøjer

# Blodtryk 3

---

## 3.1 Hvad betyder blodtryk?

Alle har på et tidspunkt i deres liv fået målt blodtryk, men hvad betyder det egentlig? Blodtryksmåling er en meget enkelt undersøgelse, der giver vigtige informationer om blodkarrenes og hjertets tilstand. Blodtryk kan måles både invasivt og ikke-invasivt. Dette projekt omhandler invasivt blodtryksmåling, som er en måling af trykket direkte i en blodåre. Blodtrykket måles i enheden millimeter kviksølv (mmHg). Et normalt blodtryk ligger på omkring 120/80 mmHg [? ].

Hjertet pumper iltet blod ud i hele kroppen via arteriesystemet og sørger dermed for tilførslen af ilt og næringssubstanser til alle muskler og organer. De røde blodlegemer (erythrocytterne) er en vigtig bestanddel af blodet. Det er hæmoglobinet i de røde blodlegemer, som binder ilten og sørger for at ilten transporteres frem til vævene i kroppen. Når ilten er afgivet fra blodet til musklerne og andre væv, transporteres blodet tilbage til højre side af hjertet via venesystemet. Det af-iltede blod pumpes af højre hjertepumpekammer ud i lungerne, hvor blodet iltes på ny og derfra strømmer til venstre hjertehalvdel for igen at blive pumpet ud i kroppen af venstre hjertepumpekammer.

Det høje tryk er det systoliske tryk, som kan måles når venstre ventrikel trækker sig sammen. Det diastoliske blodtryk er blodtrykket i hjertets afslapningsfase (diastolen). Når hjertet trækker sig sammen (systolen) skaber det en trykbølge som forplanter sig ud igennem arteriesystemet. Trykbølgen kan erkendes som pulsen, der let kan mærkes f.eks. ved palpation af a. radialis ved håndleddet. Trykket i venesystemet er meget lavere end i arteriesystemet, da blodet passivt blot skal strømme tilbage til højre forkammer, hvor trykket er lavt.

Venstre ventrikel pumper iltet blod, under højt tryk, ud i aorta og arterierne. Disse blodårer er derfor tykvæggede og elastiske i modsætning til venerne, der er ganske tyndvæggede, fordi de kun udsættes for et lavt tryk.. Hjertet overfører, gennem systolen, energi til arterievæggen, som bruges i den resterende del af hjertets cyklus, til at presse blod gennem karsystemet.

For at forstå det følgende afsnit introduceres tre vigtige begreber nedenfor:

- Væskevolumen, der løber igennem et rør pr. tidsenhed, kaldes for væskestrømmen.
- Distancen, som en væske flytter sig pr. tidsenhed er strømningshastigheden.
- Blodvolumen, der løber gennem et væv pr. tids- og vægtenhed er gennemblødning.

Der er en trykforskel imellem begyndelsen og slutningen af et rør. Væskestrømmen i røret

afhænger af trykforskellen hen over røret og modstanden i røret. Dette kan udtrykkes i følgende ligning som minder om Ohms lov for elektriske kredsløb:

$$V\text{æskestrøm}(Q) = \frac{\text{Trykforskel}(\Delta P)}{\text{Modstand}(R)} \quad (3.1)$$

Drivkraften for væskestrømningen ( $Q$ ) er trykforskellen ( $\Delta P$ ) gennem røret. Hjertets kontraktioner gør, at strømmen i røret går fra et højere, til et lavere tryk. Modstanden ( $R$ ) i en arterie er bestemt af bl.a. gnidningsmodstanden mellem arterievæggen og blodet, blodets viskositet og diameteren af arterien. Når blodet løber igennem arterierne, falder trykket efterhånden i blodet. Ved stigende modstand mod væskestrømmen forøges trykfaldet. Når modstanden i rørvæggen stiger, formindskes væskestrømningen, hvis trykforskellen samtidig ikke er steget.

Et rørs modstand bestemmes ud fra tre parametre:

- Længden af røret.
- Den indre diameter på røret.
- Viskositeten af væsken.

Jo kraftigere hjertet pumper, desto større bliver trykforskellen og dermed blodstrømningen. Blodkarrets diameter er det, der har størst betydning for modstanden mod blodstrømmen. Hvis blodet presses igennem et snævert kar, er der en større del af blodet, der er tæt på karvæggen og bliver derved bremset af friktionskraft. Modsat, hvis diameteren på karret havde været større, ville en mindre del af blodet være i kontakt med væggen og derved ville det ikke blive bremset lige så meget. Modstanden er derfor mindre og blodstrømningen større, i et stort kar.

Blodets viskositet stiger, jo flere røde blodlegemer, der findes i blodet. Jo flere røde blodlegemer, desto højere viskositet. Hvis blodet har en høj viskositet og derved er tyktflydende, så skal der et større tryk til at holde en hvis væskestrøm [? ].

## 3.2 Hypertension

Hypertension er en meget almindelig lidelse, ca. 30% af den danske befolkning har forhøjet arterielt blodtryk [? ]. Derfor er det vigtigt ofte at få målt sit blodtryk, da forhøjet blodtryk ikke kan mærkes og er den vigtigste årsag til hjerte-kar-sygdomme. Der er tale om hypertension når blodtrykket er 140/90 mmHg eller højere. Ved hypertension bliver arbejdsbelastningen af hjertet forøget, da der skal pumpes blod ud af hjertet mod en større modstand i arteriesystemet.

Forhøjet blodtryk gør at arbejdsbelastningen bliver større. Derved sker der lige så stille en fortykkelse af muskulaturen i venstre ventrikel, da hjertet skal bruge flere kræfter på at pumpe blodet ud i aorta. Det øgede tryk påvirker blodkarrenes belastning, og kan medføre at mindre blodkar kan bryde under det store tryk. Hvis der er tale om blodkar i hjernen kan dette føre til en hjerneblødning. Hypertension er den hyppigste årsag til hjerneblødninger. Hypertension kan føre til en række andre komplikationer i form af

åreforkalkning, hjertheinsufficiens [? ], akut myokardieinfarkt, hjertekrampe, nyreskader og apopleksi. Forhøjet blodtryk behandles med lægemidler, i form af blodtryksnedsættende medicin (antihypertensiva). Samtidig er non-farmakologiske metoder, som rygestop, motion, reduktion af saltindtagelse, vægttab og reduktion af alkoholforbrug vigtige i behandlingen af hypertension.

### 3.3 Hypotension

Hypotension er modsat hypertension et lavt blodtryk, og ikke nær så almindeligt. Der er tale om hypotension når blodtrykket er 90/60 mmHg. Hypotension ses især ved en række alvorlige akutte tilstande som akut myokardieinfarkt, lungeemboli, sepsis eller alvorlige blødninger. Patienten kan i disse situationer være i en livstruende shock tilstand. Lavt blodtryk kan også i sjældne tilfælde forekomme kronisk, især ved sjældne stofskiftesygdomme med nedsat produktion af binyrebarkhormoner [? ].

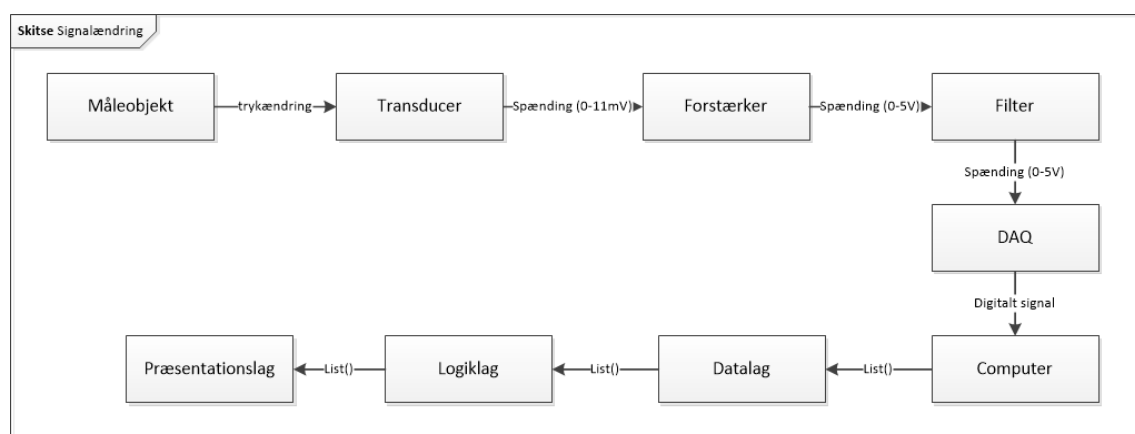


# Systembeskrivelse 4

Med udgangspunkt i projektformuleringen kommer dette projekts endelige system til at bestå af et software og hardware system, der kan tilsluttes et måleobjekt, hvorpå et blodtryk kan måles. Systemet skal kunne implementeres i forskningsmiljøer, hvor en eller flere forskere ønsker at analysere indhentede blodtrykssignaler. Visionen er, at systemet skal være let tilgængeligt og effektivt, hvilket vil komme til udtryk ved, at systemet fungerer stabilt.

I dette projekt realiseres en prototype af systemet. Det vil sige at flere dele af systemet udvikles ud fra forsimplede metoder ift. hvordan det vil være optimalt at implementere dem i virkeligheden. Her tænkes på hardware-, såvel som software-elementer. Hardwaren i prototypen realiseres på et Veroboard, så det er muligt at tage den med sig, samt er mere holdbar over tid. Softwaren i prototypen består af flere moduler. Disse er opbygget efter principperne i en tre-lagsmodel, hvilket vil sige, at koden indeholder et database-lag, logik-lag samt presentations-lag. Dette er valgt for, at skabe et overblik over hvilke dele af software-koden der har ansvaret for de enkelte funktionaliteter i systemet.

Hardwaren består af en forstærker og et filter. Forstærkerens opgave består i at forstærke det analoge signal fra max 11 millivolt til 5 volt. Filtret sørger for at filtrere unødigt støj fra det analoge signal. Signalændringen fra måleobjektet til visning af signal på en graf er skitseret herunder. Det skal pointeres at dette kun er en skitse for at skabe overblik, derfor er flere processer i softwaren udeladt af diagrammet. Database-laget består af en



Figur 4.1: Skitse af signalændring

lukket database samt indhentningen af blodtrykssignalet fra måleobjektet til transduceren, gennem hardware, inden det rammer software-delen. I den lukkede database gemmes det indhentede blodtrykssignal i en tabel. Signalet gemmes med et tidsstempel, samt

under et autogenerated Id, sammensat med det forsøgsnavn som forskeren indtaster på brugergrænsefladen ved begyndelsen af en måling.

Logik-laget er handlingslaget, og alt kommunikation til de resterende lag går gennem dette lag. Laget indeholder flere klasser, der indeholder metoder til indhentning af systoliske-, diastoliske og puls-værdier, ud fra det indhentede blodtrykssignal. Derudover indeholder laget også klasser, der har ansvaret for at foretage en filtrering af signalet når dette er valgt.

Præsentationslaget er forskerens vej ind i systemet, dette lag har til ansvar, at udskrive valgte data på brugergrænsefladen.

Systemet skal udadtil have en brugergrænseflade i form af en touch skærm eller almindelig computerskærm med tilhørende tastatur. Det er denne skærm som den primære aktør til systemet, altså forskeren, interagerer med. Det tilstræbes, at opbygge brugergrænsefladen simpelt og efter forskerens logik, så opbygningen giver mening for systemets bruger. Efter indhentning af blodtrykssignal er systemet i stand til grafisk at vise signalet kontinuerligt, samt udskrive blodtrykssignalets systoliske-, diastoliske- og puls-værdier.

# Krav 5

---

Til projektet er der opstillet to krav, som er formuleret i projektoplægget. Det er et krav at disse funktioner implementeres i produktet. Kravene lyder på at der skal udvikles et system, som kan tilsluttes et væskefyldt kateter, samt vise en blodtryksskurve på en computerskærm. Mere detaljeret vil det sige, at systemet skal indeholde to elementer. Først et elektronisk kredsløb, som forstærker signalet fra tryktransducere og filtrerer det med et indbygget analogt filter. Derefter et program til at vise blodtrykket, som funktion af tiden.

Yderligere skal dette program opfylde kravene:

- Programmeres i C#
- Kunne kalibrere blodtrykssignalet og foretage en nulpunktsjustering
- Vise blodtrykket kontinuert
- Kunne lagre de målte data i en tekstfil eller en database
- Kunne filtrere blodtrykket via et digitalt filter, denne funktion skal kunne slås til og fra

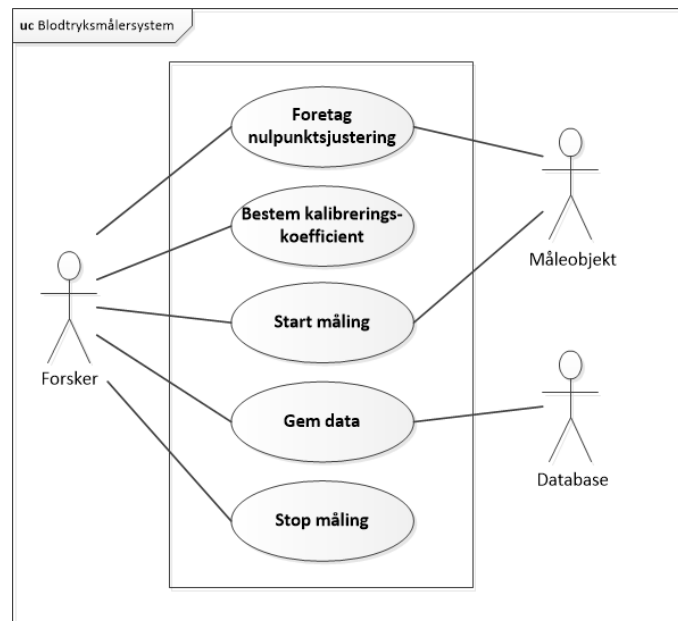
På baggrund af disse krav er der opstillet fem Use Cases, der tager højde for disse krav, samt beskriver aktørens interaktion med systemet. Disse Use Cases benyttes som kravspecifikation, der har til formål at specificere, hvilke krav der stilles til projektet. Udover ovennævnte krav vil der også blive arbejdet hen imod at systemet skal afbillede puls, det systoliske blodtryk og det diastoliske blodtryk med tal.

Kravene opstilles ud fra kundens ønsker samt leverandørens mulighed for realisering. Systemet består af en computer med programkode, en NI-DAQmx, en transducer og hardware Veroboard med et filter og en forstærker. Den fulde beskrivelse af hver enkelt Use Cases (fully dressed Use Cases) findes i dokumentationen.

## Aktørbeskrivelse

Use Case diagrammet viser de tre aktører: Forsker, Database og Transducer. Herunder er der en detaljeret beskrivelse af hver aktør.

**Forsker** er en primær aktør. Det er denne aktør, som foretager blodtryksmålingen, kalibrer og fortager nulpunktsjustering. Målingerne for blodtrykssignalet vises på displayet, som forskeren har tilgang til.



Figur 5.1: Use Case diagram

**Transducer** er en sekundær aktør. Transduceren har til formål at modtage blodtrykssignalet fra måleobjektet, som kan bestå af In Vitro, patient eller anden, som kan skabe et blodtrykssignal.

**Database** er en sekundær aktør. Denne aktør er en database, hvori blodtrykssignalets rådata gemmes. Ligeledes gemmes det indtastede forsøgsnavn og det autogenerated Id.

### Use Case beskrivelse

Use Case diagrammet viser ligeledes de fem Use Cases, der er for systemet: Foretag nulpunktsjustering, Bestem kalibreringskoefficient, Start Måling, Gem data, Stop måling. Disse Use Cases beskriver interaktionen mellem aktørerne og systemet. Herunder er der en kort beskrivelse formålet med hver Use Case.

**UC1: Foretag nulpunktsjustering** Når systemet startes op vil det første der møder forskeren være en GUI, hvorfra nulpunktsjustering kan foretages. Det forudsættes at forskeren har åbnet for transduceren, så den modtager atmosfærisk tryk inden at forskeren trykker på foretag. Systemet indhenter så en nulpunktsjusteringsværdi, denne værdi gemmes i softwaren og alt indhentet signal herefter vil dermed være indstillet til en offset på nul. Herefter går systemet videre til næste Use Case

**UC2: Bestem kalibreringskoefficient** Kalibreringen foregår uafhængig af om systemet kører. Forsker tilslutter hardware til væskesøjle ved 50 mmHg. Så aflæses output spænding fra hardwaren. Kalibreringskoefficienten kan så bestemmes ved en simpel beregning ud fra tryk og output spænding. Denne værdi indtastet så i fil, hvorfra softwaren kan tilgå koefficienten og derefter benytte den som omsætningskoefficient mellem volt og mmHg. Dermed er en kalibrering udført.

**UC3: Start måling** Det kræves at måleobjekt, hvorpå blodtrykssignal ønskes fra, er tilsluttet. Derfor tilslutter forskeren transduceren til måleobjektet. Derpå kan målingen

startes ved, at forskeren trykker på knappen START MÅLING. Herefter indhenter systemet blodtrykssignalet, som bliver udskrevet på display. Værdier for puls, systolisk- og diastolisk blodtryk udskrives på display.

**UC4: Gem data** Det er muligt for forskeren at gemme det indhentede blodtrykssignals rådata, indenfor en periode, valgt af forskeren. Dette gøres ved, at forskeren trykker på knappen START GEM. Systemet vil herefter begynde at sende rådata ind i databasen, hvor dataene gemmes. Dette vil systemet blive ved med at udføre indtil forskeren trykker på knappen STOP GEM. Ved tryk på STOP GEM vises filnavn på display.

**UC5: Stop måling** Det er muligt for forskeren at stoppe visning af blodtrykssignalet. Dette gøres ved, at forskeren trykker på knappen STOP MÅLING. Systemet vil herefter lukke forbindelsen til indhentning af data og grafen i chart vil fastholdes.

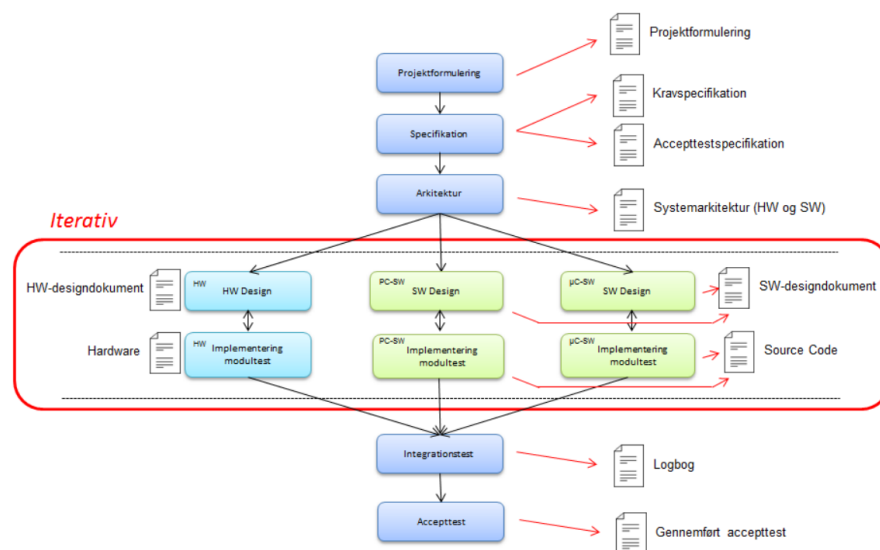
### **Ikke-funktionelle krav beskrivelse**

Ikke-funktionelle krav er struktureret efter (F)URPS+, hvor krav til systemets funktionalitet, brugervenlighed, pålidelighed, præsentation samt vedligehold er beskrevet. Disse krav er primært software-krav. Der opstilles bl.a. et krav om en maksimal tid, der må gå fra, at der er trykket på en knap, til at systemet reagerer. Når der er trykket på en knap, skal systemet foretage den ønskede proces, hvilket eksempelvis er, at ved tryk på GEM-knappen, at sende det ufiltrerede blodtrykssignal til databasen, hvor signaldata bliver gemt. Ligeledes er opbygning af display også en del af de ikke-funktionelle krav.

# Projektbeskrivelse 6

## 6.1 Projektgennemførelse

Dette projekt er gennemført vha. forskellige udviklingsprocessor, hvilket er med til at sikre kvalitet, og at deadlines overholdes. En af disse modeller er ASE-modellen [? ]. Denne model er en udviklingsmodel, der er udarbejdet af Aarhus Ingeniørhøjskole. Modellen er en gentagelig udviklingsproces drevet ud fra projektets Use Cases. Modellen er benyttet på den måde, at gruppemedlemmerne fastlægger en projektformulering, kravspecifikation og systemarkitektur, for derefter at designe og implementere de enkelte hardware- og softwaredele. Gennem en integrationstest ses det om hardware- og softwaredelene fungerer som de skal. Dette ender med en gennemført accepttest, således at det testes om systemet lever op til kravene.

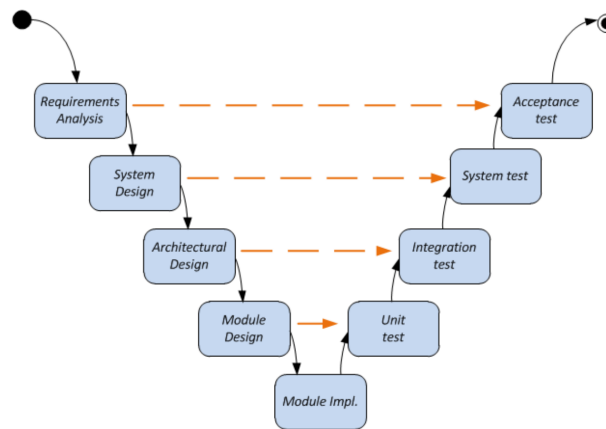


Figur 6.1: ASE-modellen

For at forstå ASE-modellen [? ] er det vigtigt at gennemgå Use Cases; et værktøj, som skal beskrive interaktioner mellem diverse aktører og selve systemet. Sammen med de ikke-funktionelle krav opnås et overblik over hvilke funktionalitetskrav, der stilles til systemet. På baggrund af kravspecifikationen kan accepttesten efterfølgende udarbejdes. I dette projekt er hardware- og software design implementering på lige fod, da projektet består af begge ting ligeligt.

V-modellen [? ] er en faseopdelt udviklingsmodel, der også er værd at nævne i dette pro-

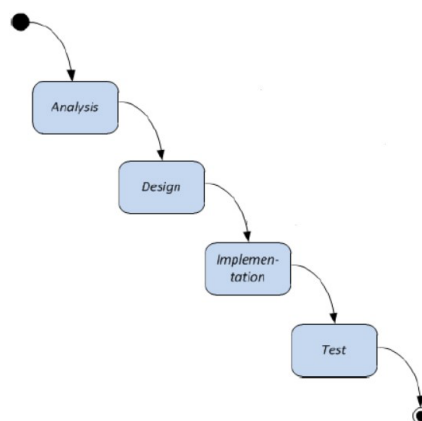
jekt. Den beskriver udviklingsfaserne og testfaserne sideløbende i forhold til projektet, og den er derfor benyttet til dette projekt sideløbende med ASE-modellen. Modellen fungerer således, at specifikationen af test foregår parallelt med udviklingen af selve systemet.



Figur 6.2: V-model

Den er blevet benyttet til hardware- og softwareudviklingen. Hardware og software skulle begge teste deres funktioner inden nye faser blev igangsat, for at verificere om disse funktioner virkede korrekt gennem forløbet. Fordelen ved at teste på forskellige niveauer er, at det skal sikre de udviklede delsystemer, således at de virker som planlagt. Det er vigtigt, at hver fase er udført, før den næste fase påbegyndes.

Vandfaldsmodellen [?] er også benyttet under dette projekt. Softwareudviklingen bærer præg af vandfaldsmodellen, da udviklingen er opdelt i faser, hvor hver fase er blevet gennemført, før den næste er påbegyndt. Dette er i relation til V-modellen, som blev beskrevet før og den er konstant strømmende nedad.



Figur 6.3: Vandfalds modellen

## 6.2 Projektstyring

Projektet er udarbejdet over et semester, hvor undervisningen og forelæsningserne delvist har udgjort grundlaget for teorien benyttet i projekt. Der blev i starten udarbejdet en tidsplan med enkelte faste deadlines og med mulighed for at sætte andre deadlines løbende i processen. Se Tidsplan under Bilag.

Projektgruppen har bestået af 6 gruppemedlemmer, som er blevet delt i to fokusområder, hardware og software. Fordelingen blev udarbejdet efter den enkeltes ønske. Projektet har været afhængig af, at der var god kommunikation mellem de to undergrupper. Da gruppen har været opdelt, har der været projektmøde hver uge, hvor gruppen har opdateret hinanden og vejleder. Samtidigt har gruppen forsøgt at sidde samlet, så på den måde at kunne opdatere hinanden løbende.

Under projektet har alle medlemmer været med til, at sikre en administrativ kæde af deadlines til individuelle opgaver, samt dagsordener til hvert møde. Disse deadlines har sikret at opgaverne er blevet opfyldt op til møderne og det har derfor været nemt at følge op på. Da der har været overlap mellem forskellige opgaver er der løbende blevet kørt og lavet adskillige teste for at sikre at opgaverne har været vellykket.

## 6.3 Metoder

Til at kunne overskue arkitektur og designet af projektet, er flere forskellige arbejdsmetoder benyttet for at skabe det bedst mulige resultat. For at finde, hvad blodtryksmåleren skal gøre, er der blevet udarbejdet Use Cases. Disse beskriver systemet funktionalitet. Use Cases viser, hvad brugeren skal opleve fra systemet, men ikke, hvordan det sker. I Use Case diagrammet bliver det også vist, hvilke aktører der findes og hvordan de interagerer med systemet.

I projektet bruges accepttest til at teste blodtryksmåleren. Dette gøres ud fra kravspecifikationen, hvor det er angivet, hvilke krav der er stillet til systemet.

Accepttesten er en test, hvor der beskrives, hvad der skal ske og, hvad brugeren skal gøre. Testen er for at undersøge om produktet opfylder de krav, der er blevet sat for det. Accepttesten giver et godt overblik for udvikleren og for kunden, der nemt og hurtigt kan se om produktet virker som det skal.

Til beskrivelse af design af software og hardware er diagrammer og skemaer blevet udarbejdet i SysML og UML. SysML er et grafisk modelleringssprog, som kan bruges til at overskueliggøre systemer.

Til software er der blandt andet lavet en applikationsmodel i SysML, som består af et domæne-, klasse- og sekvensdiagram.

Domænemodellen viser sammenhængen mellem blokkene i systemet. Blokkene findes i Use Casene og derved bliver disse to ting koblet sammen.

Klassediagrammet viser, hvilke metoder blokkene har og hvordan de kommunikerer med hinanden. Her findes domæne-, kontrol- og grænsefladeklasser. Kontrolklasserne beskriver, hvordan data behandles mellem domæne- og grænsefladeklasser. Domæneklasser indeholder funktionalitet fra den pågældende softwareblok. Grænsefladeklasserne viser, hvordan, systemet interagerer med omverdenen. Diagrammet gør det nemmere at fremme en lav kobling og høj samhørighed i softwaren.



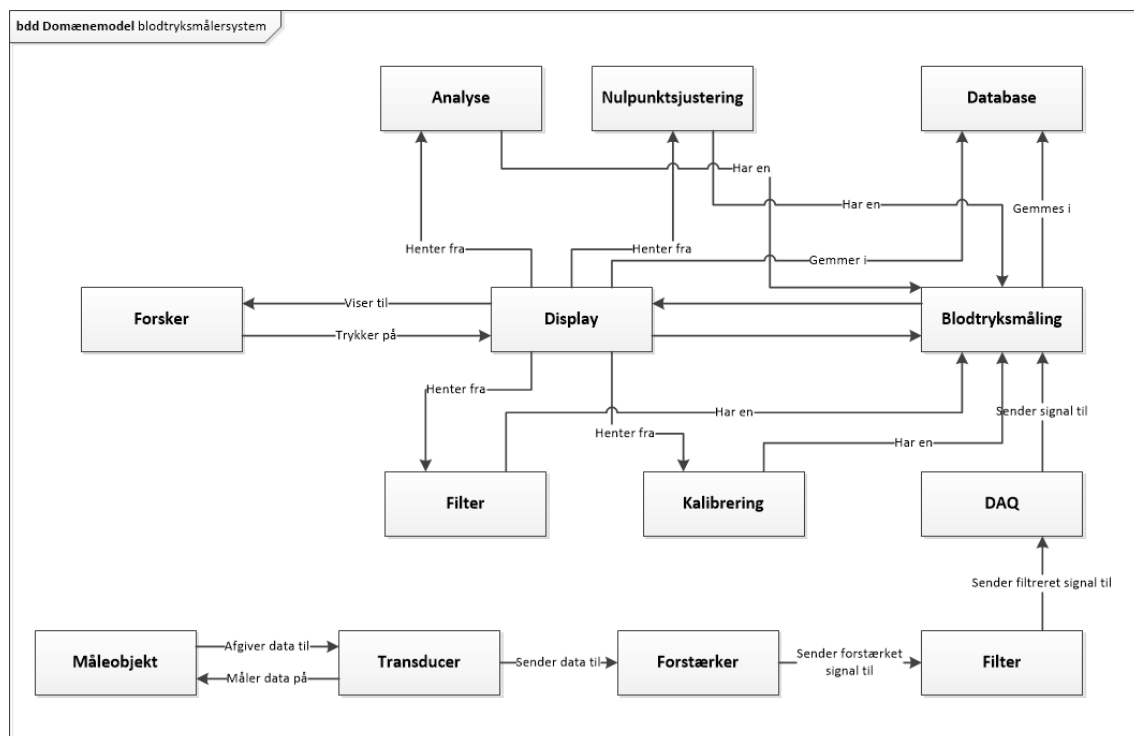
Sekvensdiagrammet fortæller, hvad der sker i selve koden. Igen går det ud fra Use Case, hvor vægten nu er på softwaredelen. Derved beskrives det, hvordan metoder bliver kaldt og hvordan de forskellige klasser interagerer. Hver Use Case skal her gennemgås i softwaren, så der skabes et overblik over vejen gennem koden.

For at skabe et overblik og indsigt i koden, er der i UML udarbejdet et aktivitetsdiagram og et klassediagram. Aktivitetsdiagrammet går i dybden med en specifik metode. Det er kun blevet gjort for relevante metoder. Her tydeliggøres det, hvordan hver metode fungerer og, hvad den indeholder. Klassediagrammet fortæller hvilke metoder, en klasse indeholder og hvordan klasserne hænger sammen.

Til hardwaren er der blevet brugt Block Definition Diagram(BDD), som viser hvilke blokke et system indeholder og hvilke porte de har. BDD er lavet til at give et overblik over systemet. Ud fra BDD'et er et Internal Block Diagram(IDB) blevet lavet. Her vises, hvilke signaler, som findes i systemet og hvordan de sendes rundt. Her vises portene igen og der skal være overensstemmelse mellem BDD og IDB.

Til udarbejdelsen af kredsløb blev Analog Discovery brugt til at simulere signalet, som i sidste ende skal komme fra transduceren. Først blev kredsløbet opbygget på et fumlebræt, hvor det blev testet for at afprøve om det lever op til kravene. Når det opfylder kravene, flyttes det over på et Veroboard. Veroboardet bliver igen testet før aflevering.

## 6.4 Systemarkitektur



Figur 6.4: Domæne Model

## 6.5 Hardware

I hardwaredelen skal der ligge en forstærkning og et lavpasfilter. Det differentieret signal fra transduceren skal forstærkes og filtreres før det kan sendes ind i DAQ'en.

### 6.5.1 Design)

#### Forstærkning

Transduceren måler en trykændring, som den omsætter til en spænding. Dette er udtrykt ved et differentieret signal, som sendes ind i forstærker-blokken.

Signalet fra transduceren er en lav spænding, som skal forstærkes op, for at passe med DAQ'ens input. Denne forstærkning udregnes ud fra det maksimale output fra transduceren og det maksimale input til DAQ'en. Se beregningerne under implementering.

Under simulering bruges Analog Discovery som en funktionsgenerator, der simulerer det differentieret signal.

#### Lavpas

I projektet skal der laves et 2. ordens lavpasfilter. Filteret skal laves for at sikre, at der ikke opstår aliasering.

Aliasering [?] er, hvor signalet bliver gentaget. Når man har signalet i det digitale domæne, bliver spektret for signalet en periodisk funktion.

Det skal sikres, at der ikke kommer overlap mellem signalet og et alias. Da det ellers kunne give anledning til misforståelser. Derfor laves et lavpasfilter, som sikre at der ikke ligger noget signal ved den halve samplingsfrekvens.

Lavpasfilteret skal være et Sallen-Key Butterworth-filter med en knæfrekvens på 50 Hz og en samplingsfrekvens på 1kHz. Ud fra oplysninger givet til projektet, vides det at filteret skal dæmpe signalet med 20 dB, under antagelse af at den forekommende støj er mindre end signalet, også når det forekommer over knæfrekvensen.

Ved en typisk blodtryksmåling forekommer der ikke signal over 50 Hz, samtidigt er signalet her aftaget med ca. 70 dB. For at få signalet, ved den halve samplingsfrekvens til at være  $1/2 \cdot LSB$  (Least Significant Bit), skal det ydeligere dæmpes 20 dB. Derfor oplyses filteret til at være 50 Hz, da dette giver en minimum dæmpning på 20 dB pr. dekade.

### 6.5.2 Implementering

#### Forstærkning

For at få den rette forstærkning er det blevet valgt, at benytte instrumentationsforstærkeren INA-114. Her kan transduceren sættes på med det differentierede signal. INA-114 er valgt da følgende gælder[?] for instrumentationsforstærkere:

- Differentielt input - single ended output
- Gain justering med ændring af kun én modstand
- Meget høj indgangsimpedans
- Stor Common Mode Rejection Ratio (CMRR)

For at udregne den korrekte forstærkning, bruges følsomheden fra transduceren og eksistationsspændingen. Først udregnes det maksimale output fra transduceren:

$$9V \cdot 250mmHg \cdot 5\mu \cdot 10^{-5}uV/V/mmHg = 11.25mV \quad (6.1)$$

Da det er besluttet at det maksimale input til DAQ'en [?] er 5V, kan forstærkningen (Gain) nu udregnes:

$$\begin{aligned} 5V &= 11.25mV \cdot G \\ G &= 444.44 \end{aligned} \quad (6.2)$$

For at få den rette forstærkning udregnes den eksterne modstand ( $R_g$ ) til INA-114 [?]. INA-114's forstærkning afhænger af størrelsen på  $R_g$ , hvis modstanden er stor, er forstærkningen lille og omvendt.  $R_g$  udregnes ved formlen:

$$\begin{aligned} G &= 1 + \frac{50k\Omega}{R_g} \\ 444.44 &= 1 + \frac{50k\Omega}{R_g} \Rightarrow R_g = 112.75\Omega \end{aligned} \quad (6.3)$$

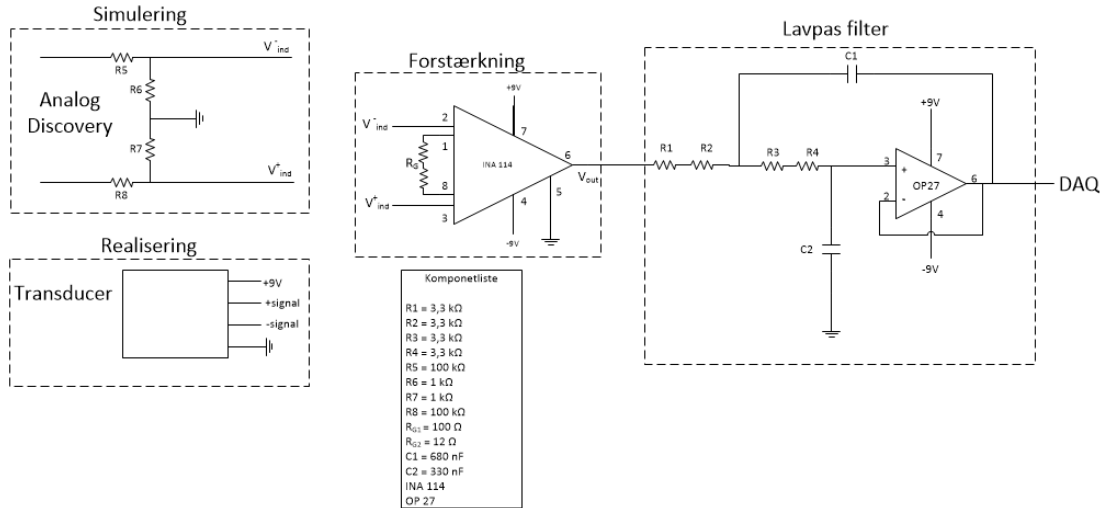
Derved fås en værdi for den eksterne modstand til INA-114, som skaber den ønskede forstærkning.

Den ønskede forstærkning kan bruges, da det passer over ens med båndbredden. Dette kan undersøges da produktet af forstærkning og båndbredde er konstant og båndbredden skal ligge over knæfrekvensen for filteret. Se beregning i Dokumentation ligning 3.4.

For at imødekomme usikkerheden ved Analog Discovery, der bruger lave spændinger, laves et kredsløb efter spændingsdelerprincippet. Signalerne fra Analog Discovery skal sendes igennem dette kredsløb, hvor de efter spændingsdelerprincippet gøres mindre.

Derved kan Analog Discovery sende signaler med en højere spænding ind i kredsløbet og usikkerheden mindskes. Hvis INA-114 skal have 11.25mV skal Analog Discovery sende 1.1352V ind.

På figur 6.5 ses et diagram af det endelige kredsløb med komponentværdier. Her ses, hvordan det ser ud ved realiseringen med transduceren og ved simuleringen ved Analog Discovery.



Figur 6.5: Diagram over HW

### Lavpas

For at opnå den ønskede effekt i lavpasfilteret, blev det oplyst at  $f_c = 50$  Hz,  $f_s = 1$  kHz,  $R_1 = R_2$  og  $C_2 = 680$  nF. Ud fra disse værdier, udregnes de resterende komponentværdier for filteret.

Overføringsfunktionen for et 2. ordens filter er:

$$H(z) = \frac{\omega_n^2}{(s^2 + 2 \cdot \zeta \cdot \omega_n \cdot s + \omega_n^2)} \quad (6.4)$$

For at finde overføringsfunktionen for det gældende system, vides det at følgende ligninger gælder [? ]:

$$\begin{aligned} \omega_n &= 2 \cdot \pi \cdot 50 = \frac{1}{\sqrt{R_1 \cdot R_2 \cdot C_1 \cdot C_2}} \\ 2 \cdot \zeta \cdot \omega_n &= \frac{1}{C_2} \cdot \left( \frac{R_1 + R_2}{R_1 \cdot R_2} \right) \end{aligned} \quad (6.5)$$

Dette indsættes i den generelle overføringsfunktion og det simplificeres, blandt andet ved at det vides at  $R_1 = R_2$ . Se Beregning af overføringsfunktion i Bilag for nærmere udregninger:

$$H(z) = \frac{1}{\frac{C_1 \cdot C_2 \cdot R^2}{s^2 + s \cdot \frac{2}{R \cdot C_2} + \frac{1}{C_1 \cdot C_2 \cdot R^2}}} \quad (6.6)$$

Når der arbejdes med et 2. ordens Butterworth filter, vides det at udsvinget  $\zeta$  skal have værdien 0.7 [? ]. Under beregningerne, var der usikkerhed omkring, hvad værdien af  $\zeta$  skulle være. Da kredsløbet skulle realiseres og dokumenteres blev det derfor overvejet at ændre samtlige komponentværdier så de passede med en  $\zeta = 1$ . Inden dette blev gjort blev det dokumenteret at det gælder at  $\zeta$  skal have en værdi på 0.7.

Den sidste overføringsfunktion sammenlignes med den generelle for 2. ordens systemer. Det gælder at  $C_2 = 680 \cdot 10^{-9}$  nF. Det er så muligt at isolerer forskellige led. Først isoleres

der for modstanden (midterste led i nævneren) og den udregnes til  $R = 6687\Omega$ .

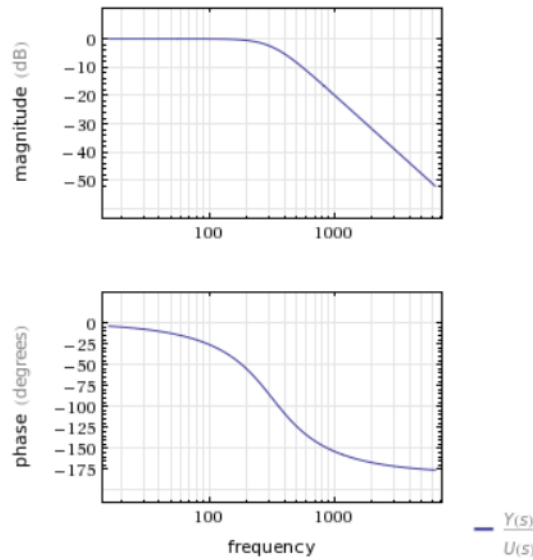
Nu kan det tredje led i nævner isoleres og kondensatoren  $C1$  udregnes til  $C1 = 333 \cdot 10^{-9} nF$ .

Nærmere beregninger kan ses under Systemarkitektur i Dokumentationen.

Alle komponentværdierne til lavpasfilteret er fundet og det kan nu realiseres.

Under udviklingen af lavpasfilteret er komponentstørrelserne, blevet ændret for at kunne realisere det. De brugte komponentværdier er:  $R = 6.6k\Omega$ ,  $C1 = 330 \cdot 10^{-9} nF$  og  $C2 = 680 \cdot 10^{-9} nF$ . For at være sikker på at filteret har de ønskede karakteristika, laves et bodeplot for den endelig overføringsfunktion:

$$H(z) = \frac{62500000000}{610929 \cdot \left( s^2 + \frac{250000}{561} \cdot s + \frac{62500000000}{610929} \right)} \quad (6.7)$$



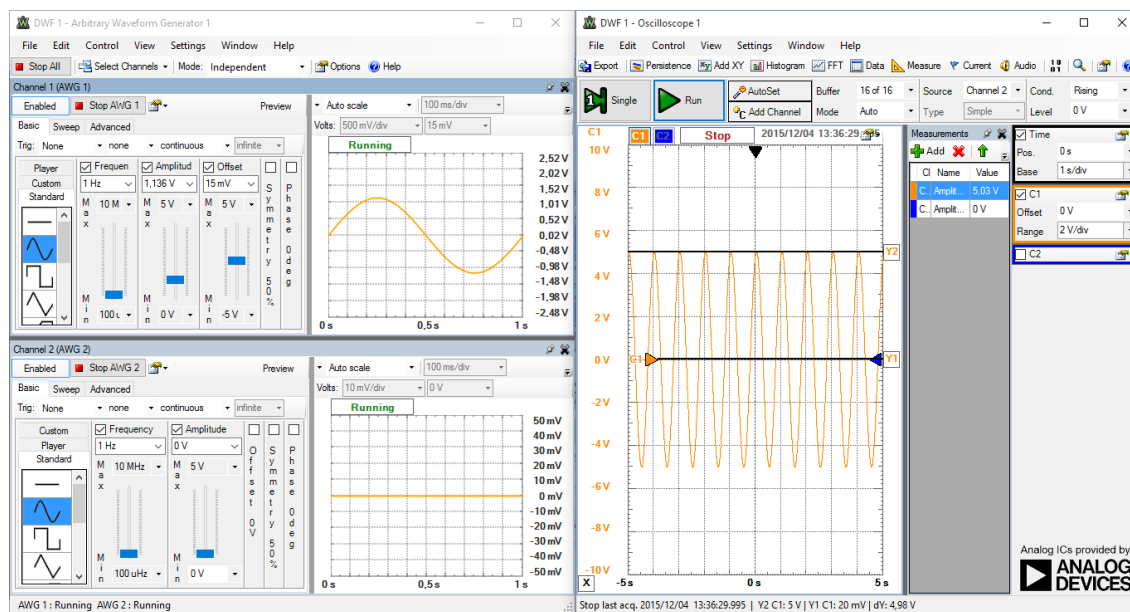
Figur 6.6: Bodeplot

Ud fra den nye overføringsfunktion udregnes en nu  $\zeta$  for at kontrollere at værdien ikke har ændret sig. Denne udregnes til  $\zeta = 0.709$ . Derfor kan det konkluderes at filteret stadig har den ønskede funktionalitet.

### 6.5.3 Test

#### Forstærkning

For at teste forstærkningen, sendes et differentieret signal ind vha. Analog Discovery. Her observeres der hvor meget signalet bliver forstærket. På figur 6.7 ses det signal, som sendes ind i forstærkningsblokken og det, der måles på udgangen af blokken.



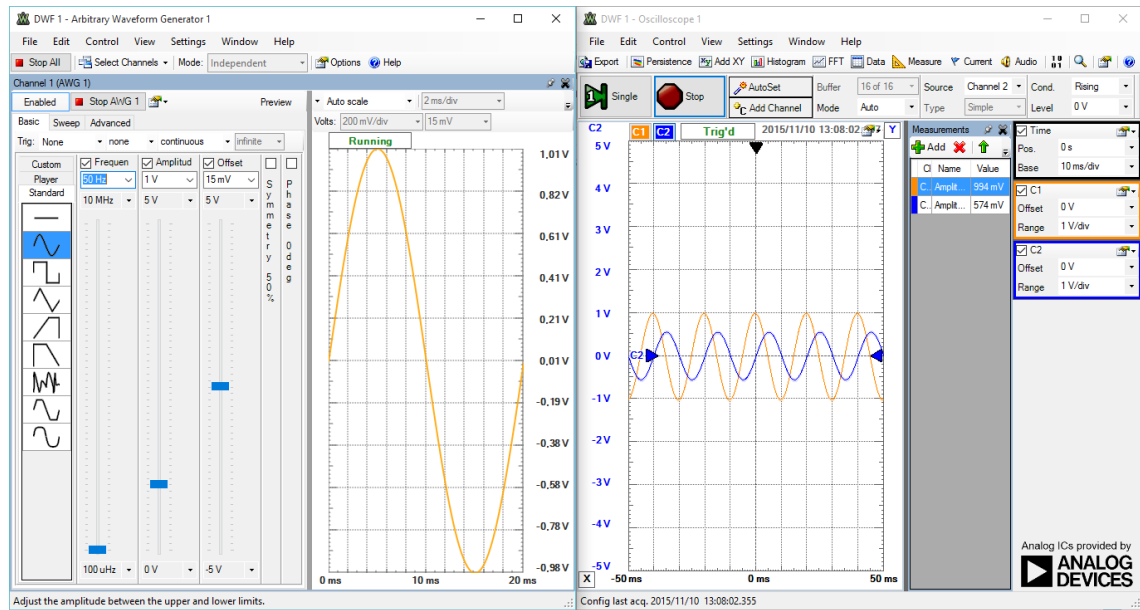
Figur 6.7: Forstærkningsblok

Der sendes et differentieret signal ind i INA-114. På udgangen ses det at signalet er blevet forstærket op til 5V DC. Herved er maksimum input til forstærkningsblokken blevet forstærket så det passer med maksimum input til DAQ'en. Signalet bliver ikke ændret på andre måde i denne blok.

## Lavpas

For at teste lavpasfilteret foretages målinger med en sinus, hvor frekvensen varierer for hver måling. Derved aflæses fasen mellem indgang- og udgangssignal, samt amplituden for hver måling. Der laves flere målinger, både før, ved og under knækfrekvensen. Ved knækfrekvensen skal fasedrejningen være  $90^\circ$ . Dette kan aflæses på figur 6.8.

Se Modultest i Dokumentationen for mere dokumentation.

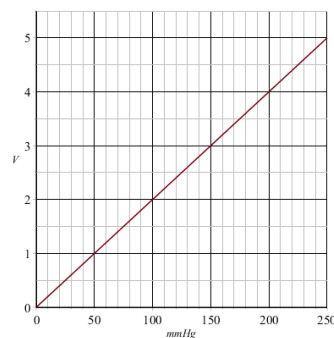


Figur 6.8: Måling for 50 Hz

### Kalibrering med vandsøjle

Efter forstærkning og lavpasfilteret er blevet testet hver for sig, udføres en kalibrering af systemet vha. en vandsøjle. Her bruges en udleveret vandsøjle med tre målepunkter, hvor det er angivet hvor højt trykket (målt i millimeter kviksølv, mmHg) er ved hvert af disse punkter. Derved kan det testes om hardwaredelen måler den rigtige spænding i forhold til mmHg.

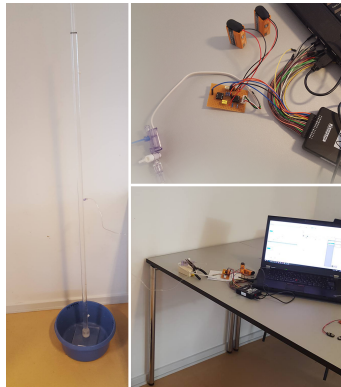
Ud fra den maksimale spænding (målt i Volt, V) og mmHg kan det udregnes, hvad hardware skal vise ved 100 mmHg, se figur 6.9.



Figur 6.9: Graf til kalibrering, fra udregninger

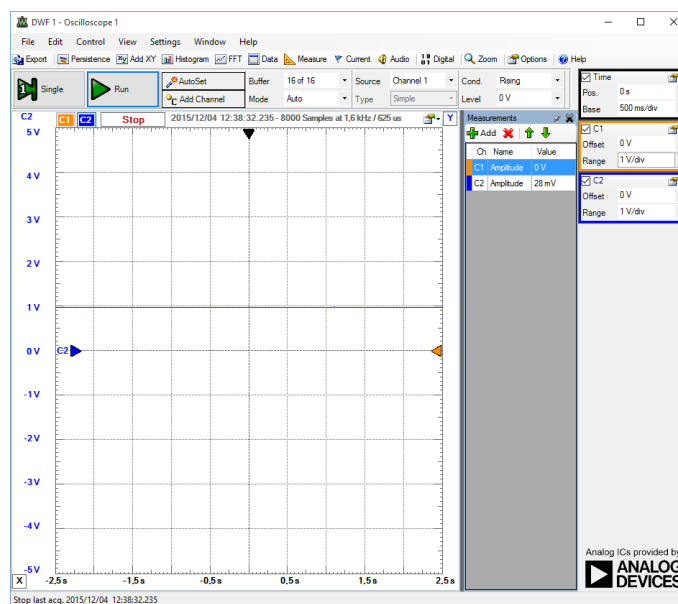
Testen udføres ved at fylde vand i søjlen til et bestemt punkt. Transduceren skal være tilkoblet netop dette målepunkt, mens de andre er lukket. Transduceren er sat til hardwaren, der hvor Analog Discovery tidligere har været sat til. Transduceren er tilkoblet 0-9V ved batterierne. På samme måde, som ved simuleringen, aflæses målingen på computeren ved hjælp af programmet WaveForms. Da det vides hvilken trykændring der måles på, ved vi fra grafen til kalibreringen hvilken spænding den skal vises. Dette fortages for de tre målepunkter på vandsøjlen, hvor hver måling sammenlignes med den udregnede

graf. For hver måling, skal transduceren flyttes til et af de andre målepunkter.



Figur 6.10: Opstilling

Ud fra grafen i figur 6.9 vides, hvad svaret til hver måling skal være. På figur 6.11 ses målingen, da transduceren var tilkoblet målepunktet for 50 mmHg. Ud fra figur 6.9 vides det at målingen skal vise 1V DC.



Figur 6.11: Måling ved 50 mmHg

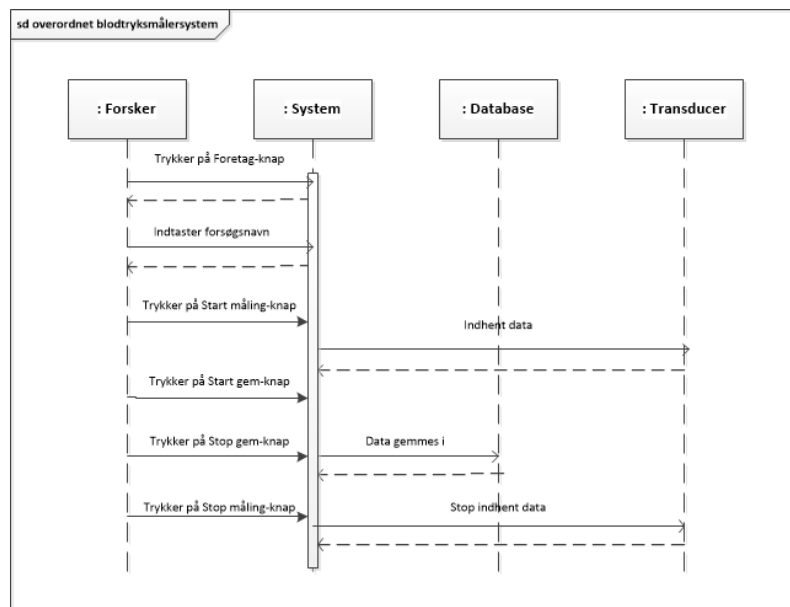
Ud fra figur 6.9 kendes udgangsspændingen også for de to andre målepunkter. Se under Modeltest i Dokumentationen for billeder af målingerne ved 10 mmHg og 100 mmHg. Målingen for 50 mmHg er her valgt ud, da denne måling ligger til baggrund for kalibreringen i softwaren.



## 6.6 Software

### 6.6.1 Design

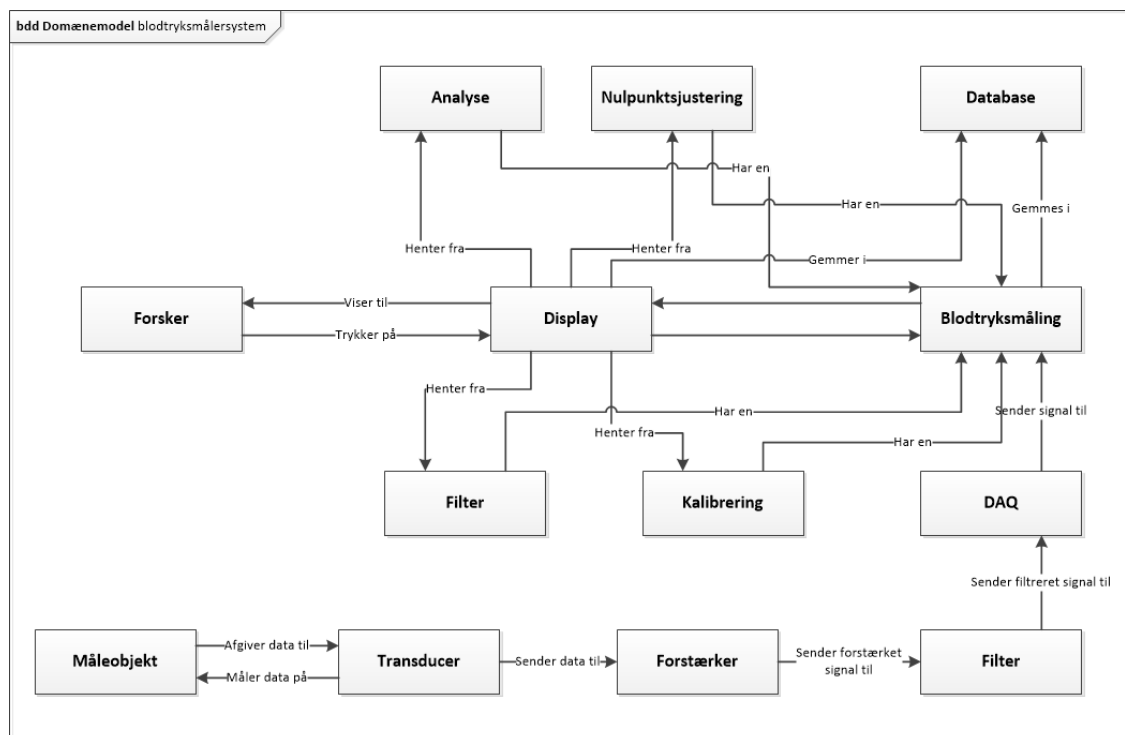
Overordnet set ønskes det at udvikle et system, der kan interagerer med en forsker. Diagrammet herunder viser at forskerens opgave består i at starte en måling, foretage en nulpunktsjustering og gemme de ønskede rådata, samt uafhængigt af systemet at bestemme en kalibreringskoefficient, hvis det ønskes. Diagrammet er en simpel illustration, som viser systemets adfærd gennem alle fem Use Cases. For yderligere præcisering af Use Cases se dokumentationen under Kravspecifikation. Formålet med dette diagram er at skabe et overblik over det samlede system.



Figur 6.12: Overordnet sekvensdiagram for systemet

### Problemidentifikation

Første step i softwaredesignet er at klarlægge, hvilke klasser systemet skal bestå af. Til dette er en domænemodel udarbejdet med udgangspunkt i de fem Use Cases. Modellen har til formål at vise, hvilke dele systemet skal holde styr på.

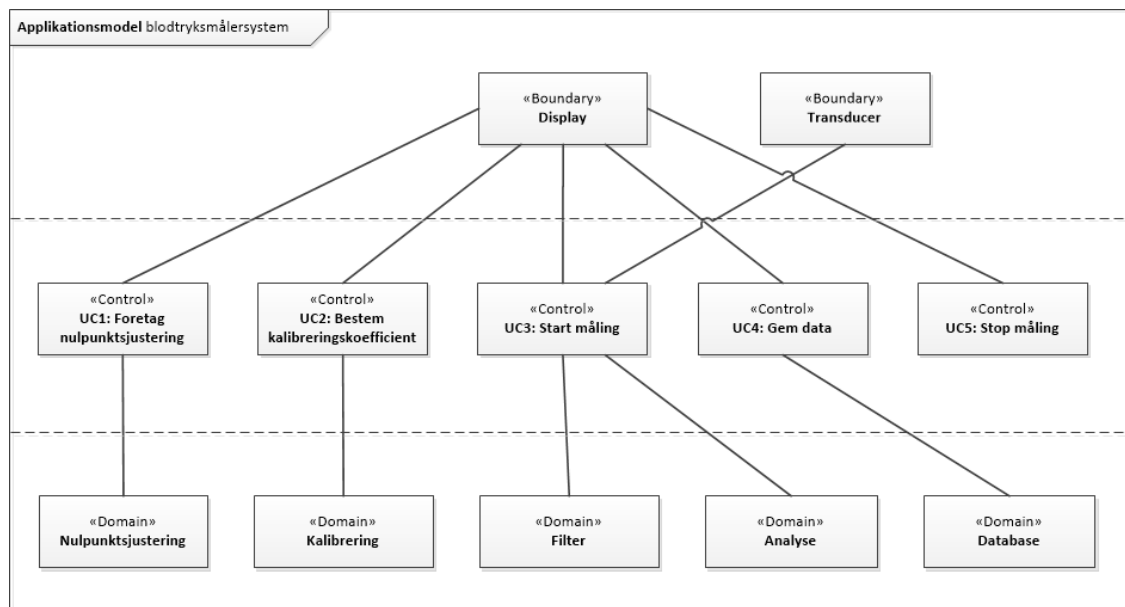


Figur 6.13: Domænemodel

Hardwarekomponenterne er medtaget for at vise signalets vej fra måleobjekt til systemet.

### Klasseidentifikation

Domænemodel leder hen til udarbejdelse af en applikationsmodel. Denne model har til formål at vise hver enkelt classes individuelle formål, samt hvilke domæne klasser der kommer i spil ved den pågældende Use Case.



Figur 6.14: Applikationsmodel for software

### Metodeidentifikation

Klasserne i applikationsmodellen er med til at definere, hvilke blokke sekvensdiagrammer må indeholde. Dermed også med til at definere hvilke metoder, der er nødvendige for at få udført de ønskede handlinger mellem blokkene. Sekvensdiagrammer for hver enkelt Use Case kan ses i dokumentationen under Softwaredesign.

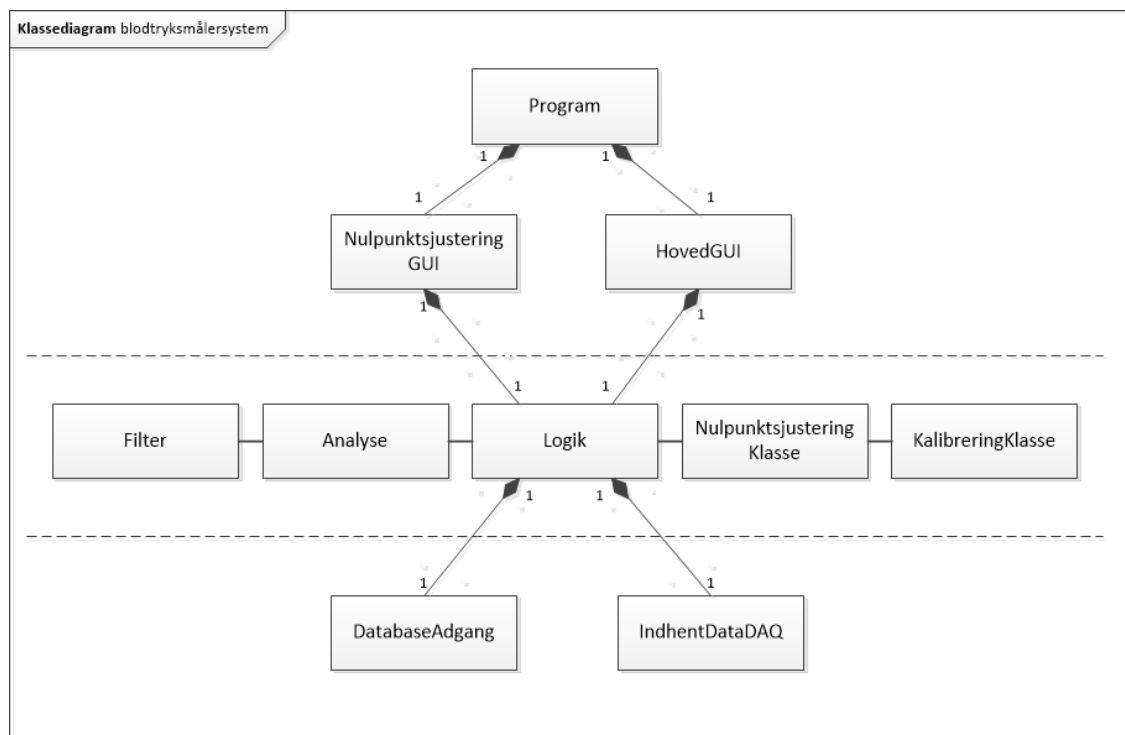
#### 6.6.2 Implementering

På baggrund af designfasen for softwaren kan implementeringen påbegyndes. Softwaredesignet viser at systemet skal implementeres med en GUI-applikation, hvor aktøren kan interagere med systemet. Derudover er det kendt at softwaren skal indeholde en række klasser, hvori funktionaliteter som kalibrering, nulpunktsjustering, digitalt filter og indhentning af systolisk, diastoliske og puls værdier skal placeres. I det følgende beskrives de overvejelser, der er gjort i forhold til implementering af disse funktionaliteter og software-systemet.

Implementeringen af softwaren sker i Visual Studio 2013 i sproget C#. Dette er valgt da programmet er et godt værktøj til at arbejde med GUI-applikationer, samt til håndtering af tråde og trådkommunikation. Tråde benyttes i softwaren da systemet, der skal implementeres, er et eventdrevet system. Det vil sige, at systemet skal kunne håndtere mange handlinger på en gang. Handlingerne igangsættes af events, der kommer af aktørens interaktion med systemet. Trådkommunikationen fungerer således, at en tråd kan sende et signal ud, som andre tråde kan reagere på.

#### Klasse implementering

På baggrund af designmodellerne er det besluttet at opbygge systemkoden efter principperne i en trelagsmodel[? ]. Trelagsmodellen indeholder et præsenterations-lag, et logik-lag og et data-lag. Alt kommunikation i trelagsmodellen skal gå igennem logik-laget. Fordelen ved trelagsmodellen er, at den skaber et godt overblik i koden, og det fremstår tydeligt hvad hver enkelt klasse specifikke ansvar er. Et overordnet klassediagram over systemet er udarbejdet, se figur 6.15. Hvilke metoder hver enkelt klasse indeholder kan ses i Bilag under "Klassediagram".

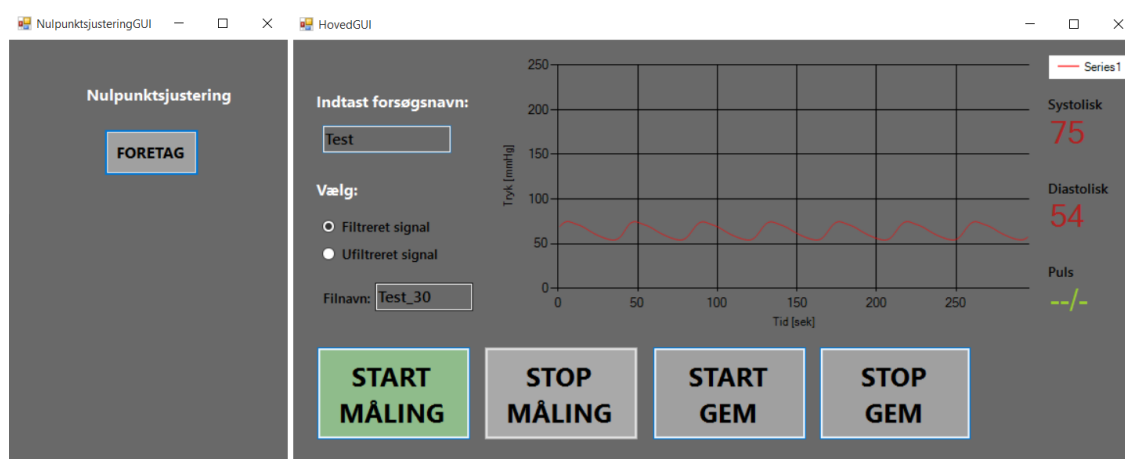


Figur 6.15: Klassediagram

## Brugergrænseflade

Displayet (GUI) er Forskerens indgang til systemet. Derfor er det vigtigt at denne er opbygget efter forskerens logik. Til at klarlægge dette er principperne om en god brugergrænseflade taget i mente.

Det er et krav at Forsker indtaster et Forsøgsnavn inden en måling kan startes, derfor er komponenterne implementeres således at knappen Start Måling først bliver aktiveret, når Forsøgsnavn er indtastet i den tilhørende tekstboks.



Figur 6.16: NulpunktsjusteringGUI og HovedGUI

Af figur 6.16 ses det, at grafen er en væsentlig del af displays HovedGUI. Det vælges at få vist signalet i grafen, som en kurve, og førsteaksen indstilles til samples fra 0-300, og

andenaksen til værdier fra 0 til 250 mmHg, hvilket er givet i kravspecifikationen.

### Observer - Strategy

Observer og Strategy er to programmeringsmønstre, det er valgt at opbygge softwarekoden efter. Grundet at de i samarbejde med hinanden er gode til, at håndtere en overførelse af data fra et lag til et andet. Observer definerer et "en til mange" relation mellem objekter således, at en ændring i et objekts tilstand medfører, at de mange objekter informeres om ændringer og dermed opdateres automatisk. Dette implementeret med to interfaces IObserver og ISubject. I disse interface er generelle metoder Notify(), Attach() og Gennemsnit() placeret, hvis ansvar er at flytte data fra en klasse til en anden, når metoden kaldes. Mønstret opbygges som en push.

Strategy mønstret indkapsler algoritmer og gør at metoder oprettet i interface kan overskrives i klasser der arver fra interfacet. Således at den nødvendige funktion kan tilføjes i den enkelt klasse.

Hvordan mønstrene er implementeret med relevante metoder kan skematisk ses i dokumentationen figur xx. I projektet blev mønstrene i første omgang benyttet fra logik-laget til præsentrations-laget i forbindelse med at overføre data til visning i graf. Men undervejs viste det sig nødvendigt også at implementere mønstrene fra data-laget til logik-laget, således at det kan kontrolleres, hvor stor en mængde data, der sendes op af gangen.

### Samplefrekvens

Samplefrekvensen er som krav givet til 1000 Hertz. Hvilket svarer til at systemet modtager 1000 samples i sekunder. Varigheden af en sample er givet ved:

$$\frac{1}{f_s} = \frac{1}{1000} = 0.001 \text{ sek} \quad (6.8)$$

Det har vist sig under arbejdet med softwaren, at grafen ikke kan følge med, når den modtager så mange målinger i sekundet. Derfor er det valgt at skære i antallet af målinger pr. sekund, der skal viderebearbejdes i logik-laget og udskrives i præsentrations-laget. Antallet skæres ned til 50 målinger pr. sekund. Dette gøres ved at gennemsnittet af 20 målinger efter hinanden bestemmes, hvorefter gennemsnitsværdien returneres og gemmes i en liste, der sendes videre i systemet. Herefter findes gennemsnittet af de næste 20 målinger og således fortsætter det.

### Nulpunktsjustering

Formålet med en nulpunktsjustering er at flytte signalets offset op eller ned, så det atmosfæriske tryk altid er placeret ved 0 V på outputsignalet. Justeringsfaktoren er givet ved, hvor  $x$  er det målte atmosfæriske tryk i Volt modtaget gennem DAQ'en:

$$f_{aktor_{jus}} = 0 - (x) \quad (6.9)$$

Systemet ønskes nulpunktsjusteret for at sikre, at alle de målte blodtrykssignaler har samme udgangspunkt. Hvilket gør at målingerne kan sammenlignes. Systemet foretager nulpunktsjusteringen ved at returnerer gennemsnittet af de første 20 målinger fra DAQ'en, såfremt den tilsluttede transducer er åbnet så atmosfærisk tryk måles.

## Kalibrering

I dette projekt betyder kalibreringen, at kalibreringsfaktoren fra volt til millimeter kviksølv bestemmes. Denne bestemmes ved at tilkoble en væskesøjle til systemet. Væskesøjlen fyldes med vand til markering, så den vil give et kendt tryk (mmHg). Herefter kan output i volt fra hardwaren måles. Kalibreringskoefficienten er givet ved:

$$koeff = \frac{x[mmHg]}{y[Volt]} \quad (6.10)$$

x angiver trykket fra væskesøjlen, denne hardcodes til 50 mmHg. Værdien for y angiver det målte spændingsoutput på hardwaren. Optimalt set er kalibreringskoefficienten 50.

Kalibreringen implementeres i softwaren ved brug af konfiguration. Forskeren beregner omsætningskoefficienten ud fra ligning 6.10. Resultatet af denne beregning indtaster Forsker i konfigurations XML-filen under App.settings. XML-filen kan tilgås uden opstart af systemet, derfor bliver kalibreringen uafhængig af, hvornår systemet kører og kalibreringen kan dermed foretages på et vilkårligt tidspunkt.

Det er vigtigt at pointere, at nulpunktsjusteringsfaktoren lægges til samtlige værdier i signalet før kalibreringskoefficienten ganges på. Dette udføres i kodens logik-lag.

## Digitalt Filter

[?] Formålet med implementering af et digitalt filter er at fjerne støj fra det indhentede signal. Dette gøres ved at udglatte signalet. Til dette kan en række forskellige filtre benyttes. I projektet er det valgt at implementere et glidende middelværdifilter (moving average filter).

Fordelen ved dette filter er, at det er simpelt at forstå og det er optimalt at bruge på signaler i tidsdomænet. Skulle signalet være vist i frekvensdomænet ville valget have faldet på et andet filter.

Det glidende middelværdifilter fungerer ved midling af en række punkter fra inputsignalet for, at frembringe hvert punkt i outputsignalet. Hvilke punkter, der tages fra inputsignalet, vil flytte sig én plads for hvert beregnet outputpunkt, heraf kommer den glidende effekt. Matematisk er filteret givet ved:

$$y[i] = \frac{1}{M} \cdot \sum_{j=0}^{M-1} x[i+j] \quad (6.11)$$

Hvor x[] er inputsignalet, y[] er outputsignalet og M er antallet af punkter, der benyttes i det glidende middelværdifilter. Denne beregning benytter sig udelukkende af punkter placeret på den samme side af output samplenummeret, hvilket vil føre til en relativ forskydning mellem input og output. M sættes til en længde på fem. Implementeringen af filteret er vist i et aktivitetsdiagram i dokumentationen under Software Implementering.

Filtret er implementeret således, at der minimum skal være 5 målinger i listen der skal filtreres førend, at filteret vil starte filtreringen. Dette er en begrænsning der ikke er optimal, men accepteres. Da den ikke vil være en begrænsning der vil påvirke Forskerens resultater i betydende grad.

Systemet gør det muligt for Forsker at vælge på GUI om det filtrerede eller ufiltrerede signal ønskes udskrevet.

## Analyse

Analyse dækker over bestemmelsen af de systoliske, diastoliske og puls værdier ud fra blodtrykssignalet. Dette er implementeret i en klasse kaldet Analyse. I en blodtrykskurve er den systoliske værdi givet ved maksimum på kurven og den diastoliske er givet ved minimumsværdien på kurven.

Metoder hvori den maksimale værdi og den mindste værdi i listen der udskrives i grafen er derfor implementeret. Værdierne udskrives i labels på GUI. I præsentations-laget er implementeret en timer, der håndterer at de systoliske og diastoliske værdier på GUI opdateres hvert 3. sekund. Intervallet på 3 sekunder er valgt, da det er passende tid til at aflæse den pågældende værdi.

I forhold til implementering af puls er der gjort en række overvejelser om mulige løsninger. Puls er defineret ved slag pr. minut og på en pulsperiode vil der være en systole og diastole. Pulsen må derfor kunne bestemmes ved at tælle antallet af systoliske værdier på 6 sekunder. Antallet ganges med 10 for at få den rette enhed. En anden mulighed er også at bestemme pulsen ved at finde antallet af samples mellem to systoliske værdier. Omregnes samples til sekunder og ganges op til et minut, må dette være lig med Måleobjektets øjeblikkelige puls. Det er ikke lykkedes at implementere puls ved projektets aflevering.

## Database

I systemet er der implementeret en lokal Database. Databasen er oprettet gennem hosten webhotel10.iha.dk. Formålet med Databasen er at lagre det målte blodtrykssignals rådata. Det er valgt at implementere Databasen af typen SQL, da denne database-type indeholder de funktioner, som er nødvendige for dette system. Data gemmes i tabeller i Databasen. Indledningsvis for at oprette den nødvendige tabel defineres en type til hver værdi. SQL-koden til oprettelse af tabel er vist på figur 6.18.

```
1 CREATE TABLE [db_owner].[SEMPRJ3] (  
2     [Forsøgsnavn] NVARCHAR (20) NOT NULL,  
3     [Id] BIGINT IDENTITY (1, 1) NOT NULL,  
4     [Datostempel] DATETIME NOT NULL,  
5     [Blodtryksmåling] VARBINARY (MAX) NOT NULL,  
6     PRIMARY KEY CLUSTERED ([Id] ASC)  
7 );
```

Figur 6.17: SQL-kode til oprettelse af tabeller i database

For yderligere forklaring til de valgte SQL-typer, se dokumentationen under Software Implementering.

### 6.6.3 Test

### 6.6.4 Integrationstest

Til sidst i projekt forløbet blev en integrationstest udført. En integrationstest laves primært for at teste om softwaren fungerer korrekt og om enhederne/modulerne der anvender hinanden. Testen retter sig mod afprøvning af det komplette program, med de eksterne systemer, i dette tilfælde sammen med hardwaren. Softwaren er langsomt blevet sammensat af de forskellige enheder som fremvisning af graf, indhentning af data,

kalibrering, nulpunktsjustering, digitalt filter og at gemme. Hver gang én enhed har været færdig er den blevet testen, hvorefter en ny færdig enhed er blevet sat på osv. Tilsidst er der blevet udført en test hvor alle enheder sættes sammen, software og hardware, og der testes derpå. Testen kan sammenlignes med en "Big Bang" test, da det var første kan vi satte den færdige software sammen med hardwaren. I dette tilfælde blev In Vitro maskinen i Cave Lab brugt til at skabe et blodtrykssignal. Dette blev sendt ind i vores system og sammenlignet med en anden "rigtig" blodtryksmåler, som var tilkomplet samtidigt. In vitro maskinen danner et tryk, som efterligner et hjerteslag. Trykket bliver skabt i vand, som presses igennem en falsk hjerteklap, derved er der opbygget en model af et hjerte som kan givet til blodtrykssignal. Billede af opstilling Der startes med at laves en nulpunktsjustering på vores system, som bagefter viste en flot signal, som lå så konstant at det var svært at se cursoren. Samtidig stod systolisk- og diastolisktryk i tal på GUI. Her viste det sig, at vores system konstant afveg fra den anden blodtryksmåler med en værdi på 2, både på systolisk og diastolisk tryk. Da den konstant afveg, vurderes det at afvigelsen skyldes at vores system ikke var kalibreret eller at den anden blodtryksmåler ikke var nulpunktjusteret. Der kunne ikke udføres en kalibrering da væskesøjlen var gået i stykker. Det blev også forsøgt at nulpunktjusterer den anden blodtryksmåler, men heller ikke dette kunne gøres. Derfor blev resultatet af testen af vores system virker som det skal efter kravene stillet til det. Billeder af vores måling kontra den anden blodtryksmåler : gerne sammensat i et billede.

## 6.7 Resultater og diskussion

Resultatet af dette projektforsøg er blevet en prototype af en hardwaredel og et softwaresystem, hvis egenskab er at detektere den systoliske- og diastoliske værdi ud fra et blodtrykssignal. Denne prototype er blevet udviklet til forskningsbrug. Det kunne være en mulighed at videreudvikle systemet således, at det kan detektere hypotension og hypertension og så alarmere. På denne måde kunne systemet altså bruges til patientbrug, på hospitaler, sygehuse, lægehuse osv.

I starten af projektet var der stillet krav til både hardware- og softwaredelen, som alle er blevet opfyldt. Derudover er der blevet stillet nogle krav fra vejleder, som er forsøgt løst. Nogle er lykket, andre er ikke (se problemrapport). Nogle af punkterne i accepttesten var ikke testbare, hvilket selvfølgelig bør opfyldes førend det ville være muligt at videreudvikle systemet.

I de ikke-funktionelle krav, var målsætningen en MTTR (Mean Time To Restore) på maksimalt 5 timer. Dette var ikke testbart, da systemet stadig var prototype og ikke et færdigt produkt. Det tænkes ydermere, at det endelige produkts hardwaredele forefindes som en sammenkobling af alle dele, så forskeren kun skal styre en enkelt sammenkoblet hardware del, i stedet for fire dele (DAQ, Veroboard og to batterier). Det ville gøre det lettere for forskeren at styre og transportere hardwaren og hvis nogle af delene i hardware skulle gå i stykker, ville det være nemt for forskeren blot at udskifte hele hardwaren, i stedet for at skulle fejlfinde og udskifte den bestemte del. Dette er selvfølgelig under forudsætningen af tilstedeværende reservedele - forsker vil altså kunne gendanne systemet indenfor 5 timer. Da systemet stadig er en prototype, bestående af flere løse komponenter, der let kan frakobles, er risikoen for nedbrud relativt høj, hvilket ikke er optimalt for et færdigt produkt. En sammenkobling af alle løse komponenter, som nævnt før, skal sikre en længere Mean Time



Between Failure.

Som projektformuleringen beskriver, er der udviklede en hardwaredel og et softwareprogram, der sammen kan detektere et blodtryk. Prototypen kan på under tre sekunder vise systolisk- og diastolisk blodtryk via en graf. Formålet og projektformuleringen er dermed blevet indfriet.

Der er blevet arbejdet og designet en GUI, der i sidste ende har fået et relativt professionelt design, som vil være let, for en forsker, at benytte. Til videreudvikling vil det være nemt at implementere systemet, på f.eks. et sygehus, da de, som skal benytte systemet uden problemer vil kunne sætte sig ind i hvordan systemet håndteres.

Skal der påpeges en negativ ting om projektet og den endelige rapport, så må det siges at der har været en del forvirring omkring seta. Seta ønskes, for det meste, at være 1, hvilket der er blevet arbejdet med i hele starten af projektet. Derfor var størrelserne på alle komponenterne bestemt ud fra en seta på 1. Fordi der arbejdede med et 2. ordens butterworth sællens-key lavpasfilter, så er seta ikke 1, men 0,7. Dette var forvirrende og tog en del tid og forstå, hvorefter alle komponentstørrelserne skulle omregnes, så de passede til en seta på 0,7. Ved hardware har der derudover været en manglende forståelse for de signaler som sendes ind og måles ved udgangen. Det har givet en stor usikkerhed for, hvornår de forskellige blokke har virket korrekt. Derfor er der blevet spildt tid på at tro at systemet ikke virkede, mens det i virkeligheden gjorde som det skulle.

Ved softwaren er det specielt teorien bag tråde og derfor forståelsen derom, som har manglet og derved skabt flere problemer. Teorien for tråde skulle have været i undervisningen, i kurset IT3, dette har dog ikke været fyldestgørende i forhold til brugen i projektet.

Kravene om kalibrering og nulpunktsjustering er også blevet opfyldt, selvom det volde mange problemer. Teoriene bag en kalibrering og en nulpunktsjustering var forståeligt, men hvordan det skulle fås til at fungere i en kode, var der ingen anelse om. Så det var problematisk at komme i gang med kodningen af disse. Derudover holde væskesøjle, som bruges til kalibrering, ikke særlig længe, så derefter blev det ikke nemmere. I løbet af projektet er tidsplanen skredet adskillige gange. Blandt andet pga. mange uforudsete problemstillinger og udefrakommende faktorer, som f.eks. DSB-miniprojekter og en KSS-eksamen, der lå midt i det hele.

## 6.8 Udviklingsværktøjer

Gennem projektarbejdet har vi anvendt en række forskellige værktøjer til udvikling af blodtryksmåler-systemet. Disse er yderligere uddybet herunder.

### Visual Studio 2013

Software delen af projektets programmering er skrevet i sproget C-sharp. Her er Visual Studio 2013 anvendt som kompiller, da programmet gør det nemt at omskrive tekst til kode. Visual Studio 2013 indeholder også funktionen Windows Form Application, der visuelt kan fremstille de ønskede resultater i form af knapper, grafer og labels mv. i en samlet brugergrænseflade, som aktøren interagerer med.

### Microsoft Visio 2016

Microsoft Visio er et tegne værktøj, der i dette projekt er anvendt til at designe både

SysML og UML diagrammer, som benyttes ved organisering af hardware og software design. Microsoft Visio er det oplagte valg, da diagrammer lavet i programmet får et enkelt og overskueligt udseende, og dermed fremstår det tydeligt for læseren hvad diagrammet vil vise.

### **Analog Discovery og Waveform fra Digilent**

Analog Discovery og Waveform er i projektet benyttet som omformer og signal generator under testfasen. Her fungerer Analog Discovery som en Waveform generator, så et analog signal kan sendes videre ind i lavpasfiltret, forstærkeren og derefter ind i DAQ'en. I den endelig implementering erstattes Analog Discovery og Waveform med transduceren.

### **NI-DAQmx**

NI-DAQmx er et værktøj udarbejdet af National Instruments, som anvendes til at omforme det indkomne analoge signal fra transduceren (Analog Discovery) til et digital signal. Værdier fra NI-DAQmx er af en type som kan anvendes i selve softwarekoden.

### **L<sup>A</sup>T<sub>E</sub>X**

L<sup>A</sup>T<sub>E</sub>X er anvendt i projektet til design og opsætning af projektrapport og projektdokumentation. L<sup>A</sup>T<sub>E</sub>X er god til tekstformatering, hvor opsætning og strukturer defineres samlet for hele rapport, samt god til versionsstyring. Til at skrive selve koden benyttes programmet TeX-maker som kompiler.

## **6.9 Opnåede erfaringer**

I dette projekt handlede det om, at integrere en hardware- og softwaredel. Derfor blev projektgruppen fra starten inddelt i en software og hardware gruppe, med indflydelse på hvilket område man ønskede at arbejde med. Ud over dette blev det aftalt hvem der stod for de praktiske ting, såsom mødeindkaldelser, mødereferater og tidsplan. Det gjorde at man i projektgruppen altid vidste, hvad der var blevet lavet og hvad de fremadrettede opgaver var. Det har gjort at tidsplanen er skrevet af en og at der derfor er en let forståeligt rød tråd igennem udviklingen af tidsplanen. At mødereferaterne kun er skrevet af en gør, at der har været en præcis skabelon, formatet og skriftsprog er det samme. Det samme kan siges om mødereferaterne. Det giver en bedre sammenhæng og en rød tråd igennem afsnittene. Selvom projektgruppen var delt i to, har grupperne altid siddet og arbejdet sammen. På denne måde fik hele projektet en sammenhæng og en rød tråd igennem rapporten. ISE blev brugt til udarbejdelse af dokumentation ved hjælp af diagrammer. Igennem projektet blev der i begge grupper, løbende lavet enhedstests og integrationstests i forbindelse med prototypen.

I hardwaredelen kunne der bruges erfaringerne fra fagene ASB og KVI, om hvordan forstærkning og filtre fungerer og bygges til det ønskede projekt. I faget KVI blev instrumentationsforstærkeren og strain gauges introduceret, hvorpå der kunne arbejdes videre med metoderne fra faget, hvilket har givet erfaringer til at bruge teorien i praksis, i form af projektet.

I softwaredelen blev der brugt erfaringerne fra 2. semesterprojekt ved bl.a. brug af tre-lagsmodellen og oprettelse af en Database. Disse erfaringer er nu blevet udvidet til at bruge

tråde ved indsamling af data og en kontinuerlig visning i en graf. Ligeledes er Observer-Strategy princippet og nye algoritmer til udskrivelse af systolisk- og diastolisk værdi, nu en del af koden. Der er et digitalt filter i softwaren, hvorfra viden og erfaringer fra DSB er blevet brugt til at opsætte sampling, aliasering og filter.

Underviserne har været enormt behjælpelige, hvilket har aflastet projektvejlederen, således at møderne kunne blive afholdt væsentlig mere effektivt, og det har givet et detaljeret og præcist svar, at snakke med underviseren der har undervist i det fag, der har volt problemer, da det ikke kan forventes af vejlederen, at have styr på det hele.

I dette semester blev det valgt at skrive rapporten og dokumentationen i  $\text{\LaTeX}$ , som var nyt for de fleste. Dette har voldt utrolig mange problemer og frustrationer. Der er blevet arbejdet særdeles meget med dette skriveprogram og der har flere gang været gruppemedlemmer, der har haft lyst til at slå en knytnæve igennem deres computerskærme, men til sidst har alle kunne se fordelene ved at benytte  $\text{\LaTeX}$  som skriveprogram. Det ser virkelig professionelt ud og det er lækkert at billeder og figurer ikke hopper rundt, som de f.eks. gør i skriveprogrammet Word. Derfor har alle i semestergruppen været glade for valget af  $\text{\LaTeX}$ .

## 6.10 Perspektivering - Fremtidigt arbejde

I fremtiden vil blodtryksmåleren kunne udvides gennem flere muligheder. Da blodtryksmåleren er lavet til forskningsbrug, har der i projektforsløbet ikke været nogle idéer til at udvikle den til patientbrug. Dette er dog ikke umuligt, da alle grundprincipperne omkring en god blodtryksmåler er opfyldt, og der ville kunne arbejdes på usability til sygeplejeskerne. Softwaren skal også opdateres til patientjournaler og lign. Men igen, er det overhovedet ikke umuligt at optimere blodtryksmåleren til patientbrug. Det vil her være en god idé også at lave en alarm til blodtryksmåleren, som viser når blodtrykket er for lavt/højt. En anden forlængelse af systemet kunne være en metode, som skal kunne vise gemte målinger. Dette vil være nyttigt hos både forskere og til en udvidet patient-blodtryksmåler, da det skaber overblik og perspektiv for forskeren/lægen.

Et log-in vindue er en anden ting, som kunne forbedre systemet, for på den måde at skabe større sikkerhed for forskeren og dataene. Et log-in vindue vil gøre at en forsker kan være sikker på, at hans målinger og forskning ikke kan tilgås af andre. Dette kræver en større udvidelse, hvor der skal laves et log-in vindue og en database, hvor password og brugernavn gemmes. Der skal også laves en metode, som kan tjekke om det indtastede password og brugernavn passer overens med det i databasen.

Generelt skal de standarder, som findes for blodtryksmålere undersøges grundigere. Specielt brugergrænsefladen, men også resten af systemet som enheder og visning af graf, skal rettes til efter de passende standarder.

Hvis systemet ydeligere skulle tilpasses forskning, kunne det gøres gennem en bedre navngivning af data i tabellen eller et bedre overblik over, hvordan data bliver gemt f.eks. gennem en liste for de gemte målinger. På den måde vil det blive nemmere for forskeren at finde frem til gamle målinger. I fremtidigt arbejde skal systemet også være i stand til at vægte de enkelte værdier som fås i blodtryksmåleren. Dette betyder at systemet skal kunne vurdere integriteten af tallene, og se hvilket tal der skal lægges mest vægt på, ud af de fem, som derefter summeres, for at give en.

I forhold til hardware er målet, at det hele skal samles i en kasse. Så det på den måde

ikke er muligt at ændre eller stille ved det. Derved skal filteret og forstærkningen laves på en printplade. Samtidigt skal det ved kassen være en plads til batterierne, hvor det er muligt at kunne skifte dem, når nødvendigt. Derved fås et samlet system, som nemt kan flyttes rundt på og som ikke er i fare for at gå i stykker. Dette system kan også, i stedet for udskiftelige batterier, have et genopladeligt lithiumbatteri, som vil kunne lække besværet ved ofte at udskifte batterier.

# Konklusion 7

---

Formålet med projektet var, at udvikle en blodtryksmåler, som består af en hardware- og en softwaredel. Hardwaredelen består af en forstærker blok og et lavpasfilter. Hardwaren modtager et differentieret signal fra transduceren som forstærkes og filtreres inden det sendes gennem en DAQ og videre ind i softwaren. Softwaren vil gennem algoritmer, kunne detektere systoliske- og diastoliske værdier. Ud fra de to værdier ville programmet kunne udregne pulsen.

Da projektet blev lavet til forskningsbrug skulle forskeren kunne gemme sine målinger i en database. Herudover var formålet med projektet, at blodtryksmåleren skulle kunne blive kalibreret og nulpunktsjusteret, når det var ønsket.

I dette projekt er det lykkedes, at udvikle en repræsentativ prototype, der opfylder de overordnede formål. Systemet viser, på under tre sekunder, den systoliske- og diastoliske værdi. Det er derfor hurtigt og nemt, for en forsker, at få målbare data.

Selvom dette kun er en prototype, så er hardwaredelen blevet udviklet til at være brugervenlig. Kredsløbet er blevet loddet på et Veroboard og lagt i en kasse. Batterierne holdes udenfor kassen, da det på denne måde vil være ubesværet for forskeren, at skifte batterierne. På kassen vil en lille diode lyse, når batterierne er sat til.

På trods af et funktionsdygtigt slutprodukt, må det konkluderes, at arbejdsprocessen blev stressede. Både hardware- og softwaredelen tog længere tid end forventet. Hensigten med den første tidsplan var god, men pga. DSB miniprojekter og en KSS eksamen, blev tidsplanen skubbet og især softwaredelen blev presset hen mod slutningen. Derfor er det ikke lykkedes at opfylde alle punkter i kravspecifikationen (disse punkter er beskrevet i problemrapporten).

Alt i alt er der blevet udviklet en fornuftig prototype, der med en smule videreudvikling, uden besvær, ville kunne bruges af en forsker.

# Bilag 8

---

Bilagene kan findes på den tilførende CD. Herunder findes en liste over bilagene.

1. Samarbejdsaftale
2. Beregninger af overføringsfunktion
3. OP27 Datasheet
4. INA114 Datasheet
5. Transducer Datasheet
6. NI-6009 DAQ Datasheet

# Figurer

---

4.1	Skitse af signalændring . . . . .	9
5.1	Use Case diagram . . . . .	12
6.1	ASE-modellen . . . . .	14
6.2	V-model . . . . .	15
6.3	Vandfalds modellen . . . . .	15
6.4	Domæne Model . . . . .	17
6.5	Diagram over HW . . . . .	20
6.6	Bodeplot . . . . .	21
6.7	Forstærkningsblok . . . . .	22
6.8	Måling for 50 Hz . . . . .	23
6.9	Graf til kalibrering, fra udregninger . . . . .	23
6.10	Opstilling . . . . .	24
6.11	Måling ved 50 mmHg . . . . .	24
6.12	Overordnet sekvensdiagram for systemet . . . . .	25
6.13	Domænemodel . . . . .	26
6.14	Applikationsmodel for software . . . . .	26
6.15	Klassediagram . . . . .	28
6.16	NulpunktsjusteringGUI og HovedGUI . . . . .	28
6.17	Aktivitetsdiagram for metoden updateListe() . . . . .	32
6.18	SQL-kode til oprettelse af tabeller i database . . . . .	33