

Resumé

Den følgende rapport beskriver arbejdet med gruppens semester-projekt for 2. semester på Aarhus School of Engineering. Formålet med projektet er at afprøve de metoder og emner som semesterets fag har introduceret. Rapporten beskriver arbejdsmetoderne der er brugt fra idé til produkt, det egentlige produkt og udviklingen af dette samt de overvejelser og værktøjer som er benyttet.

Projektet tager udgangspunkt i et system til at beskytte børn i hjemmet mod stød og forbrændinger fra elektriske apparater. Fra et computer program kan en bruger tænde og slukke enheder som er koblet til 230 Vac el-nettet uden behov for ekstra kabler. En kodelås forhindrer børnene i selv at tænde for strømmen igen og forældre kan dermed slukke for farlige elektriske apparater centralt, så børn ikke kommer til skade hvis de ikke er under opsyn.

Udviklingsforløbet er styret efter ASE-modellen som er en halv-iterativt projektledelses-metode. Produktet er udviklet på de platforme der er gjort brug af på 1. og 2. semester. Som enheder til modtagelse og afsendelse af data over el-nettet bruges STK500 kittet fra Atmel samt egen udviklet hardware, til at interface til 230 Vac el-nettet og en almindelig PC bruges som bruger interface. Som kodelås anvendes DE2 boardet fra Altera.

Projektet er endt ud i et funktionelt bruger interface, en X10 afsender enhed og en næsten funktionel X10 modtager.

Abstract

The following document describes the work and process of the groups 2nd term project at Aarhus School of Engineering. The purpose of the project is to use and evaluate the methods and subjects taught at this terms courses. The report describes how the product came from an idea to a physical product as well as the details of the product and the methods used.

The product developed is a child security system protecting unattended children from electrical shock and heat burns as a course of dangerous home appliances. From a computer program the user can turn on and off the power of mains outlets without the other cables than that of the mains. A code lock secures that only authorised users can access to program and that way parents can disable dangerous home appliances when they leave their children unattended.

Development is managed with the ASE-model, which is a semi-iterative project management process. Further more is it done on different known platforms introduces first and second term. As transmitting and receiving units Atmel's STK500 development board is used along with specialised hardware. A PC is used as the user interface and the DE2 board from Altera is used as the code lock.

The results include a functional user interface, an X10 transmitter and an almost functional X10 receiver.

Indholdsfortegnelse

Kapitel 1	Indledning	7
Kapitel 2	Projektafgrænsning	9
Kapitel 3	Systembeskrivelse	11
Kapitel 4	Krav	13
Kapitel 5	Kravspecifikation	15
5.1	Usecases	15
5.1.1	Bruger	16
5.1.2	Eksterne enheder	16
5.1.3	Barn	16
5.1.4	SMS modtager	16
5.1.5	DE2 Board	16
Kapitel 6	Arbejdsproces	19
6.1	Udviklingsmodeller	19
6.2	Møder, tidsplan, logbog og referater	20
6.3	HW/SW teams	21
6.3.1	Hardware teamet	21
6.3.2	Software teamet	21
Kapitel 7	Udviklingsværktøjer	23
7.1	LaTeX	23
7.2	Visual Studio	23
7.3	Atmel Studio	23
7.4	National Instruments Multisim	23
7.5	Microsoft Visio	23
7.6	Altera Quartus II	24
7.7	Filhåndtering	24
7.7.1	GitHub	24
7.7.2	Dropbox	24
7.7.3	Google Drev	24
Kapitel 8	Systemarkitektur	25
8.1	Hardware arkitektur	25
8.2	Domænemodel	25
8.2.1	BDD Hardware	26
8.2.2	Plantegning over HW	26
Kapitel 9	Design	27

9.1	Hardware design	27
9.2	Encoder	27
9.2.1	Højpasfilter	28
9.2.2	Zero Crossing Detector	29
9.3	Decoder	29
9.3.1	Båndpasfilter	30
9.3.2	Envelope Detector	31
9.3.3	Dipswitch	31
9.4	Software design	32
9.4.1	Applikationsmodel for X10 udtag	32
Kapitel 10 Implementering		35
10.1	Hardware implementering	35
10.2	Software implementering	35
Kapitel 11 Resultater		37
Kapitel 12 Konklusion		39
Kapitel 13 Individuel konklusion		41
13.1	Bjørn Sørensen	41
13.2	Jakob Schmidt	41
13.3	Jeppe Stærk	42
13.4	Jesper Christensen	42
13.5	Mick Kirkegaard	42
13.6	Poul Overgaard	42
13.7	Simon Kirchheiner	42
Kapitel 14 Litteraturliste		45
14.1	Bøger	45
14.2	Hjemmesider	45
14.2.1	Opslagsværker	45

Indledning

1

Med udgangspunkt i børnesikkerhed i hjemmet er der blevet udviklet et produkt, som kan hjælpe familier med børn, til at få et mere sikkert hjem.

Af problemstillinger som kan opstå i en almindelig husholdning kan nævnes:

- Fare for at et barn tænder for en kogeplade, eller andre elektriske varme aggregater, og efterfølgende kan brænde sig
- Fare for at et barn kan skære sig på køkkenknive som ligger i en skuffe

Den anden del af systemet er en babyalarm. Næsten alle mennesker i Danmark har deres mobiltelefon i nærheden hele tiden, så i stedet for at skulle have en babyalarm med rundt også, så kan man koble sin mobil til systemet og få besked når barnet giver lyd fra sig.

Dette ender ud i tre produkter:

- Afbryder til valgt 230 Vac stikkontakt
Beskyttelse mod kogeplader og lignende
- Låsemekanisme til at låse skabe og skuffer
Aflåsning af skuffe med køkkenknive
- Babyalarm til lyddetektering
SMS-beskeder i stedet for en ekstra "boks" i lommen

Systemet skal være nemt at sætte op og skal kommunikere over det eksisterende 230 V vekselspændings netværk i hus installationen.

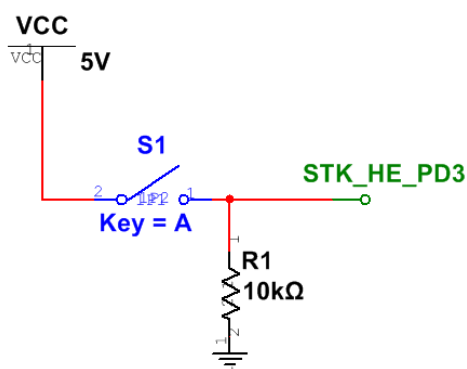
En central computer håndterer styringen i mellem enhederne og systemet kan aktiveres med et kodetryk.

Projektafgrænsning 2

Grundet begrænset tid og ressourcer er det nødvendig fra start at sætte nogle begrænsninger til hvilke dele af systemet der ønskes realiseres, som det ligeledes har været nødvendigt under forløbet at skære ned på hvad vi har ønsket realiseret.

X10 opererer normalt på 230 V nettet, men da vi ikke har autoritet til at arbejde med 230 V og af sikkerhedsmæssige årsager foregår realiseringen ved 18 V 50 Hz. Dette ændrer ikke på funktionaliteten eller virkemåden af systemet.

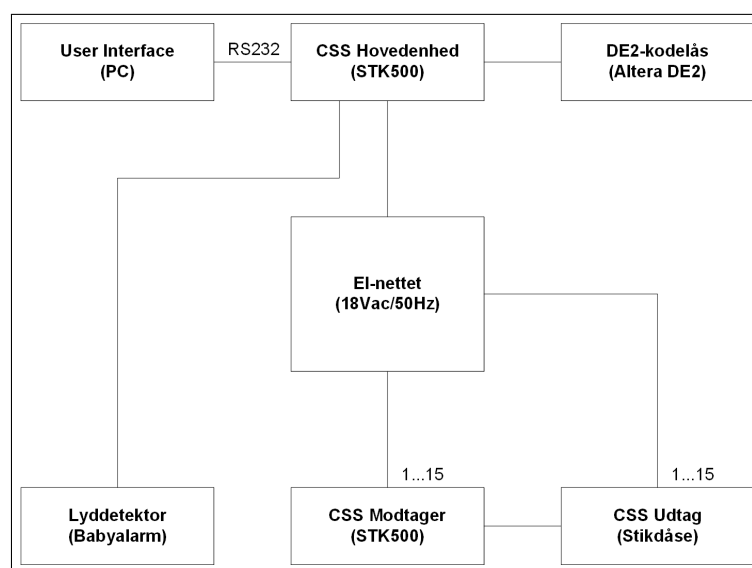
Lyddetektionen er desværre ikke nået realiseret som ønsket. Det er i stedet lavet med en knap der giver et højt signal som skal imitere at lyd er detekteret. Se figur 2.1.



Figur 2.1. Schematic over knap for lyddetektion

Systembeskrivelse 3

Fra kundens synspunkt består systemet af en computer og nogle kontrollerbare stikdåser rundt i huset samt en babyalarm. Her beskrives hele systemet mere detaljeret.



Figur 3.1. Systemoversigt

I figur 3.1 ser man, at systemet består af en computer, der har forbindelse til CSS-hovedenheden; denne er kundens kontrolenhed (User Interface). Her har kunden adgang til et simpelt menusystem, hvori kunden kan kontrollere stikdåserne (CSS-udtagene). Der er mulighed for at aktivere, deaktivere eller udlæse status på systemet. Der er endvidere også mulighed for at ændre mobilnummeret til den person, som skal modtage babyalarm-SMS'en. Hele systemet kræver en 3 cifret adgangskode, der indtastes ved hjælp af et eksternt hardwaremodul (DE2-kodelås).

Når en kommando bliver eksekveret i User Interfacet, sendes der data serielt ud via RS232 til CSS-hovedenheden. Denne data bliver encoderet til en X10-bitstrøm og sendt ud på el-nettet (18Vac). Denne bitstrøm bliver så aflæst af CSS-modtagerne, og hvis en CSS-modtager har den korrekte adresse, vil kommandoen blive udført på dens udtag. Et CSS-udtag kræver sin egen CSS-modtager.

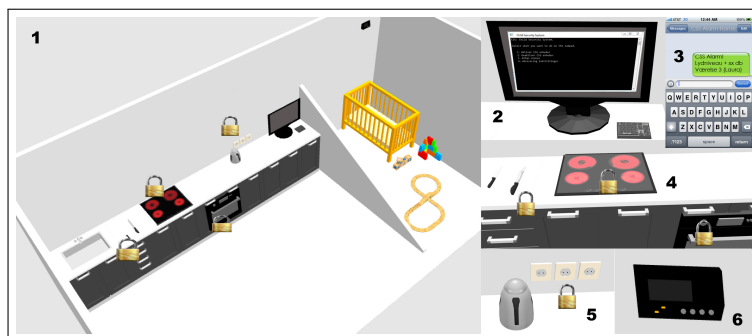
Babyalarmens funktion er at give CSS-hovedenheden besked om støj i det værelse, hvor lyddetektoren sidder placeret. Lyddetektoren er kablet direkte til CSS-hovedenheden og virker ved, at den ved et givent lydniveau¹ vil sende et signal til CSS-hovedenheden, som

¹se ikke-funktionelle krav i produktdokumentationen

via dennes serielle port fortæller PC'en, at der skal sendes en SMS til det forudbestemte mobilnummer med en advarsel.

Installation i hjemmet

Her ses, hvordan installationen kunne laves i en kundes hjem. Hængelåsene angiver hvilke enheder i hjemmet, der kan interageres med.



Figur 3.2. Installationsoversigt

1. Samlet oversigtstegning af CSS.
2. CSS-programmet med tilhørende DE2-kodelås.
3. SMS-besked udsendt af systemet, idet lydniveauet i værelse 3 (Laura) har været over det tilladte.
4. Overblik over, hvad systemet er tiltænkt at børnesikre. Køkken skuffe med skarpe genstande, kogeplader, ovn.
5. 230V udtag. X10 styret, således at det bestemmes, om udtaget skal være aktivt.
6. Babyalarm. Illustrationen vil variere i forhold til virkeligheden.

Krav 4

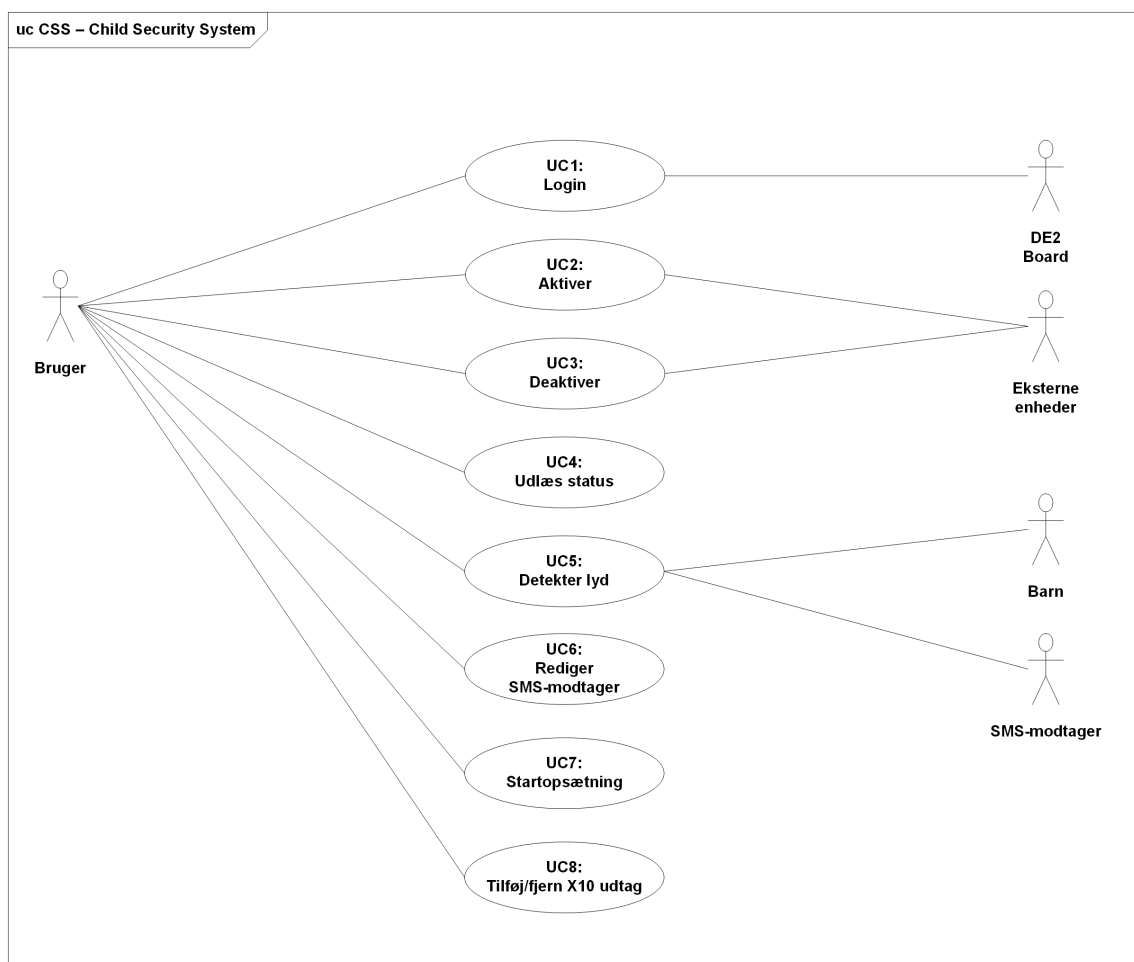
Child Security System er et system vi har udviklet med udgangspunkt i at gøre den almindelige husstand mere sikker for børn. Samtidig skal systemet have en integreret baby overvågning som er tiltænkt som erstatning for den almindelige baby alarm. Via en hovedenhed styres udvalgte enheder, hvorfra disse kan slukkes og tændes. Disse enheder er almindelige 230 V udtag som ønskes at kunne afbrydes eller syre en magnetlås til aflåsning af døre eller skuffer.

Baby overvågningen består af en sensor der måle lydniveauet i et rum som sender et signal til hovedenheden hvis grænseværdien overskrides. Dette signal fortæller hovedenheden at den skal sende en SMS til et telefonnummer efter kundes ønske.

Kravspecifikation 5

Der er blevet udarbejdet en kravspecifikation ud fra følgende use cases se figur 5.1, her er der beskrevet hvilke use cases brugeren, DE2 boardet, eksterneenheder, barn og sms-modtager har kontakt med.

5.1 Usecases



Figur 5.1. Usecase diagram

For yderligere beskrivelse af hver enkelt use case og større forståelse henvises til projektdokumentation¹

¹Projektdokumentation s 9-14

Herunder ses en beskrivelse af hver enkelt aktør i systemet.

5.1.1 Bruger

Type Beskrivelse	Bruger aktøren er ejeren af systemet eller den voksne med adgang til Computeren. Vil typisk være forældre, barnepige osv. (Primær)
-------------------------	--

5.1.2 Eksterne enheder

Type Beskrivelse	Eksterne enheder, omfatter hvad man ønsker at aflåse eller slukke for. Vil typisk være skabe, komfur, el-kedel osv. (Sekundær)
-------------------------	--

5.1.3 Barn

Type Beskrivelse	Barnet eller børnene i huset, som systemet skal beskytte. (Sekundær)
-------------------------	--

5.1.4 SMS modtager

Type Beskrivelse	Typisk forældrene eller barnepigen. Den person der skal have besked om gråd eller anden støj fra børneværelset. (Sekundær)
-------------------------	--

5.1.5 DE2 Board

Type Beskrivelse	DE2 Board programmeret som kodelås i DSD øvelse 7 (Sekundær)
-------------------------	--

Ikke-funktionelle krav

Herunder ses en beskrivelse af ikke-funktionelle krav.

Brugbarhed (Usability)

1. UI skal kunne bruges efter gennemlæst manual.

Pålidelighed (Reliability)

2. Levetid: 5 år uden hardware nedbrud
3. Software opetid: Minimum 1 måned før genstart

Ydeevne (Performance)

4. System respons må maksimalt være 2,5 sekunder
5. Startuptid fra power-off til funktionel tilstand maksimalt 2 minutter
6. Systemkapaciteten er på maksimalt 15 CSS udtag
7. Ved lydtektion må der maksimalt gå 1 minut før SMS-besked er afsendt

Vedligeholdelse (Supportability)

8. X10 udtag kan udskiftes separat ved simpel omkodning ved hjælp af adresseswitchen
9. Systemet er plug'n'play i en almindelig husholdning
10. X10 udtag kan tilføjes og installeres løbende

Generelle krav

11. Systemet skal virke på det eksisterende 230 Vac netværk i almindelige husstande
12. Kommunikationen mellem X10 udtag og hovedenheden skal ske på X10 protokollen
13. Systemet skal kunne afsende SMS-beskeder
14. Systemet skal automatisk logge ud efter 1min uden aktivitet

CSS enheder

15. Udtag skal kunne være i en 1,5 moduls Fuga stikdåse
16. Udtag skal have en LED indikator som viser at den er aktiv
17. Hovedenheden skal kunne virke på 230 Vac/13 A tilslutning

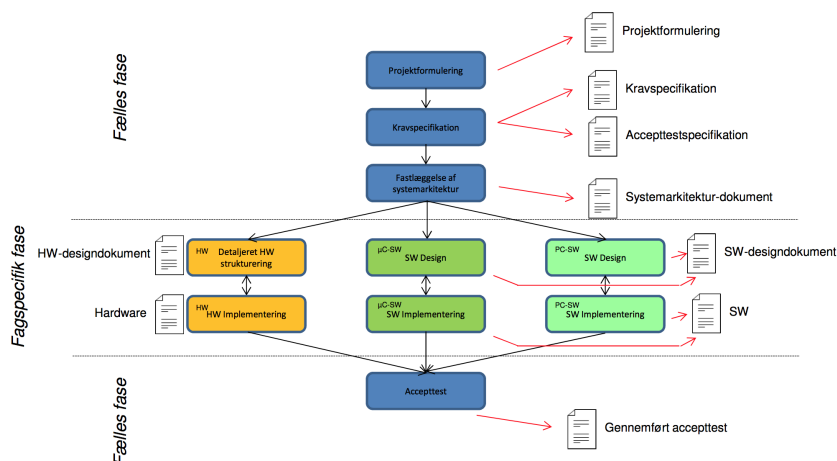
Eksterne enheder

18. Lyddetektoren skal registrere lyde på over 68 dB
19. Der må maksimalt afsendes 1 SMS-besked pr. minut ved gentagende reaktion fra lyddetektoren
20. Låse enheder må maksimalt være 8x5x3 cm
21. Låse enhederne skal kunne holde 5 kilogram

Arbejdsproces 6

6.1 Udviklingsmodeller

Med udgangspunkt i ISE-undervisningen¹ er projektarbejde opbygget omkring ASE-modellen som ses på figur 6.1. ASE-modellen er opbygget i 2 faser. En fællesfase og en fag specifik fase.



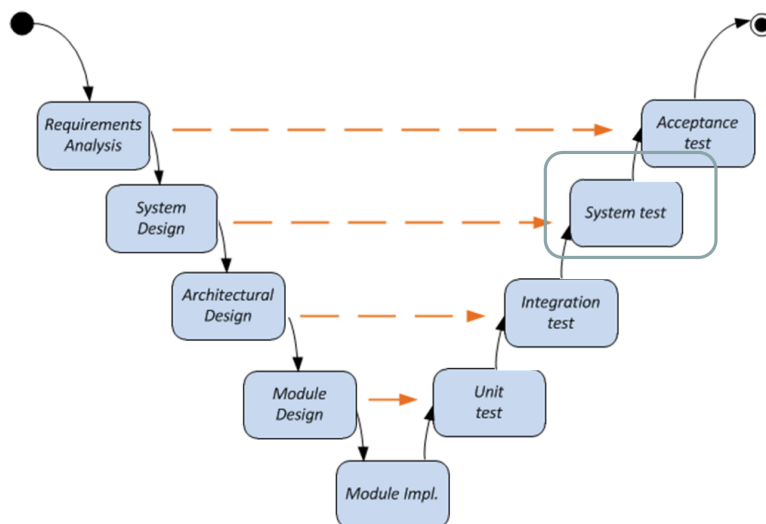
Figur 6.1. ASE-modellen

I fællesfasen arbejde hele gruppen sammen omkring udarbejdelse af de forskellige deldokumenter. I den fag specifikke fase deles gruppen op i mindre teams for at udvikle de fag specifikke deldokumenter.

ASE-modellen tager udgangspunkt i V-modellen som ses på figur 6.2.

V-modellen ses på figur 6.2. Ved at benytte V-modellens opbygning færdiggøres en fase inden en ny påbegyndes. Og ydermere planlægges testen af alle faserne parallelt med at fasen udarbejdes. F.eks udarbejdes accepttesten samtidigt med at kravespecifikationen udarbejdes.

¹Indledende System Engineering



Figur 6.2. V-modellen

6.2 Møder, tidsplan, logbog og referater

I forbindelse med projektforsløbet er der afholdt en række møder. Vejledermøder, gruppemøder samt reviewmøder.

Vejledermøder er forbindelsen mellem gruppen og gruppens vejleder. Her har det været muligt at få løbende feedback samt et indblik i om det der forventes også er det gruppen forventer. Vejledermøder har været fastlagt til én i ugen. Det er næsten opretholdt, dog med enkelte aflysninger.

Gruppen har hver uge holdt mindst et, nogle gange flere møder. Disse møder er brugt til at afklare uoverensstemmelser og planlægning af den kommende uge. Under gruppemøderne er der 2 gange brugt tid på en trivselsrunde. Her har det været muligt at give ris/ros til gruppen og eller enkelte. Gruppemøderne startede lidt løst, men dette blev hurtigt ændret til at have en fast mødeholder, som styrede mødets gang. Tidsplanen er under gruppemøderne blevet revideret, således at den altid var opdateret til vejledermøderne.

Gruppekonskrakt - bilag - skal underskrives og scannes så?

Reviewmøder har fungere således at gruppen enten udførte review på en anden gruppen og herefter fremlagde dette. Omvendt modtog gruppen lignende review fra andre grupper. Disse review førte ofte til uklarheder, som gruppen herefter måtte tage stilling til i gruppemødet.

Alle møder blev ajour ført med logbog og mødereferat. Her har gruppen haft en fast sekretær.

6.3 HW/SW teams

6.3.1 Hardware teamet

Hardware teamet bestående af: Jakob, Mick, Poul og Simon har arbejdet meget sammen om opgaven. Samarbejde er nøgleordet for dette team. Alle fire har hovedsagligt deltaget i alle opgaver herunder.

6.3.2 Software teamet

I software gruppen bestående af Bjørn, Jeppe og Jesper har vi arbejdet sammen under de indledende faser og først i den detaljerede designfase har vi delt opgaverne op. Derfra har vi arbejdet individuelt, men dog med regelmæssige møder og afklaring for at sikre at interface aftaler og lignende stadig blev overholdt.

Udviklingsværktøjer 7

Gennem hele projektforsløbet er der anvendt forskellige programmer og værktøjer til de respektive opgaver. Nogle programmer havde vi kendskab til på forhånd hvor andre var helt ny for enkelte eller alle gruppe medlemmer.

7.1 LaTeX

Hele rapporten er skrevet i LaTeX. Dette valg kom i starten af projektet da IDA havde et tilbud om et gratis endags kursus, hvor hele gruppen blev enige om at deltage.

LaTeX er et kodebaseret tekstredigerings program som er designet netop til større rapporter. Formålet er at gøre forfatteren fri for at skulle bekymre sig om formateringer således at han/hun kan rette al fokus på indholdet i rapporten.

Texmaker er benyttet som teksteditor.

Det krævede dog lidt tid i starten at komme i gang med LaTeX, men da det var på plads fungerede det rigtig godt.

7.2 Visual Studio

7.3 Atmel Studio

Atmel Studio 6.1 er det brugte værktøj til programmering af software til CSS hovedenheden og X10 udtaget.

7.4 National Instruments Multisim

National Instruments Multisim er benyttet i forbindelse med design af kredsløbsdiagrammer.

7.5 Microsoft Visio

Som del af ISE-undervisning er der blevet undervist og anvendt Microsoft Visio til udarbejdelse af diverse diagrammer. Herunder UML og SysML. SysML er anvendt til at designe blok diagrammer og internal blok diagrammer for hardwaren. UML er anvendt til software relaterede diagrammer.

7.6 Altera Quartus II

Altera Quartus II er anvendt til VHDL programmeringen af DE2 kodelåsen.

7.7 Filhåndtering

Til håndtering af filer er nedenstående 3 løsninger brugt.

7.7.1 GitHub

GitHub er et sky-baseret versionsstyringsprogram. Det er brugt til de produktmæssige dokumentationer, dvs. software kode, hardware diagrammer og projektdokumentation samt projektrapporten.

7.7.2 Dropbox

Dropbox benyttes som cloud løsning. Dropbox har fungeret som fælles harddisk. Primært benyttet i forbindelse med de 2 afholde reviews. Ydermere er dropbox benyttet til deling af litteratur.

7.7.3 Google Drev

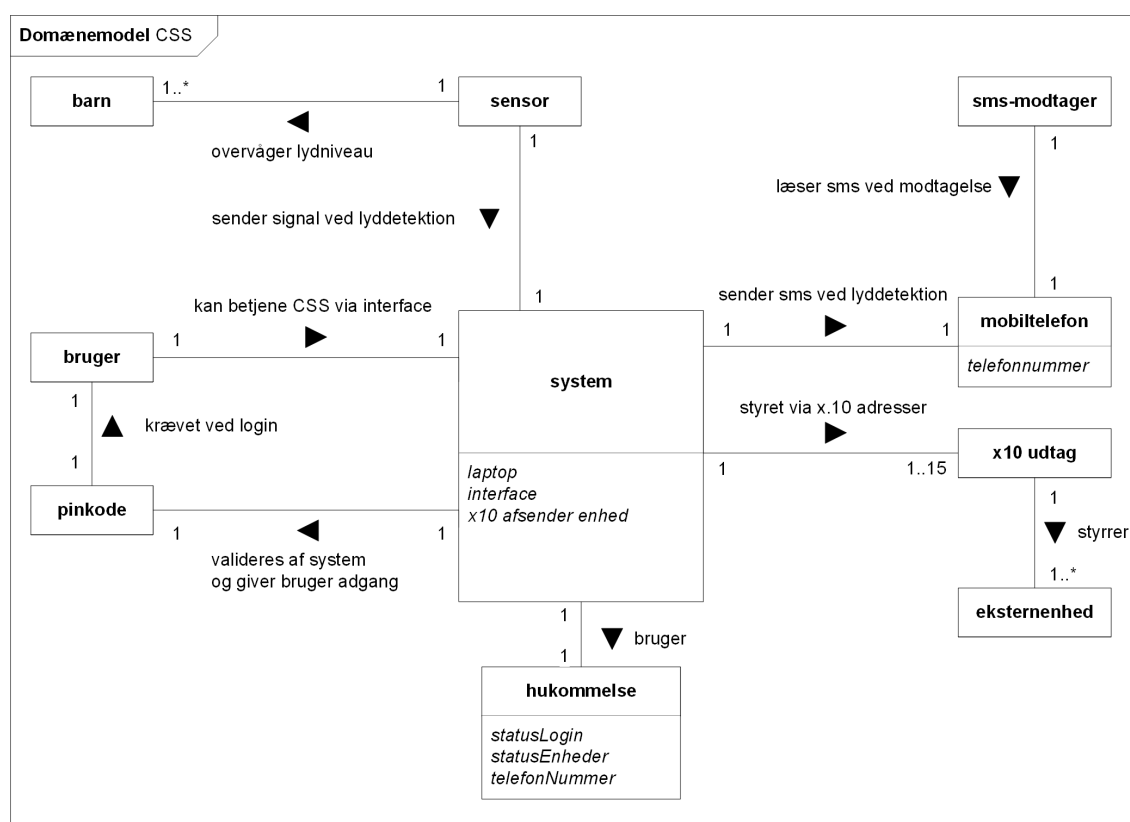
Logbogen, mødereferater er udarbejdet i Google Drevs dokument funktion. Og tidsplanen er udarbejdet i regneark funktionen. På den måde kan alle se og rette i det samme dokument samtidigt.

Systemarkitektur 8

8.1 Hardware arkitektur

Efterfølgende diagrammer viser hvordan hardware arkitekturen er opbygget.¹

8.2 Domænemodel

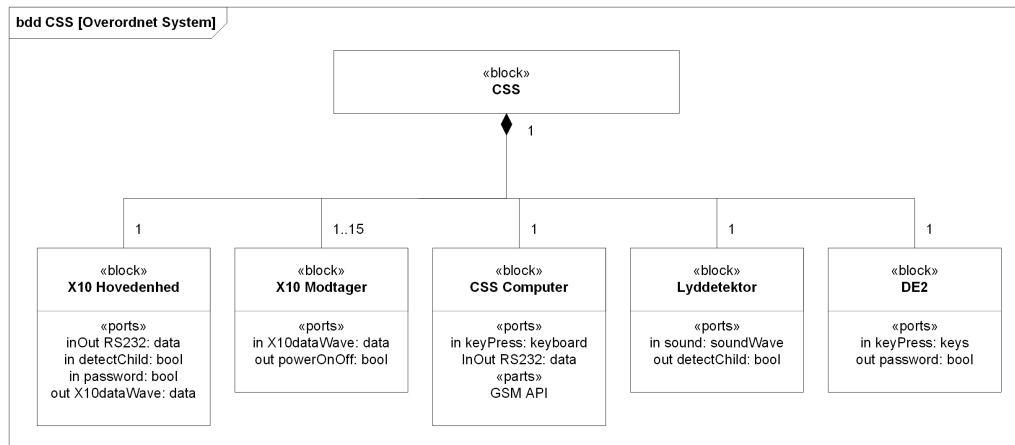


Figur 8.1. Domænemodel

Domænemodel er udarbejdet i samarbejde med kunden. Denne har til opgave at give et struktureret billede af systemets funktionalitet og sammenhæng. Domænenmodellen gør ikke brug af fagudtryk, men pile og kortfattede samt præcise sætninger anvendes for at beskrive sammenhængen mellem blokkene. Dette er med til at opnå en højere forståelse, af systemet som helhed, for kunden.

¹For yderligere BBD/IBD se projektdokumentation afsnit System Arkitektur.

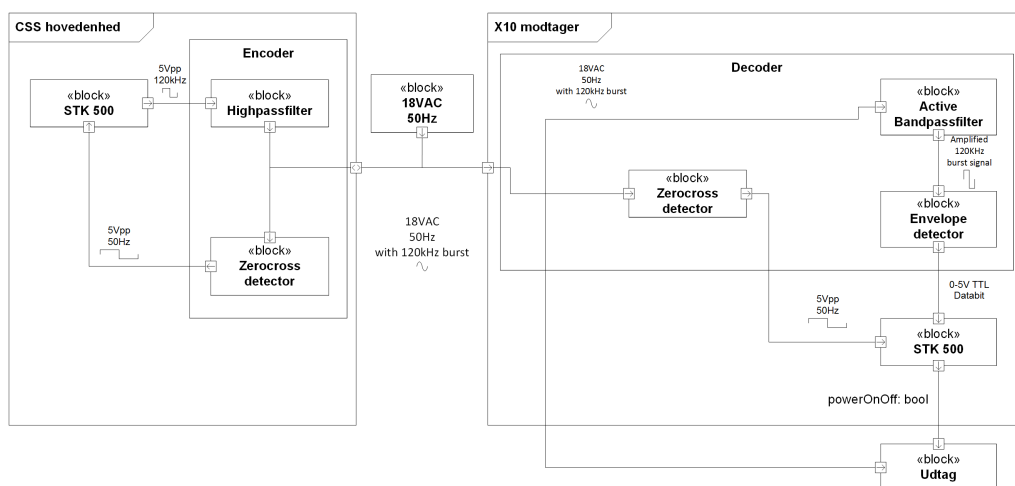
8.2.1 BDD Hardware



Figur 8.2. BDD Hardware

BDD diagrammet giver et overblik over hvad det samlede system består af. Vi ser en port beskrivelse som viser hvilke signaler hver blok består af.

8.2.2 Plantegning over HW



Figur 8.3. Plantegning over HW

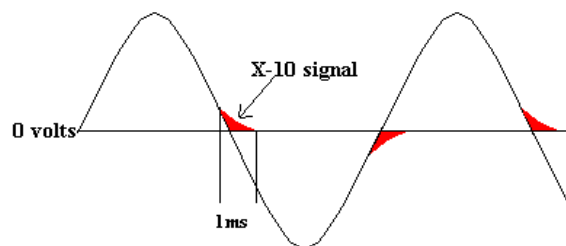
Plantegningen over HW giver et overblik over hvordan CSS hovedenheden og X10 modtager er forbundet, samt hvilken type signaler der bliver sendt imellem dem.

9.1 Hardware design

Generelt kan det overordnede hardware design beskrives som noget elektronik der sender information ud på nettet, som sendes og analyseres af noget elektronik i den anden ende. Disse vil blive beskrevet herunder som encoder og decoder¹.

9.2 Encoder

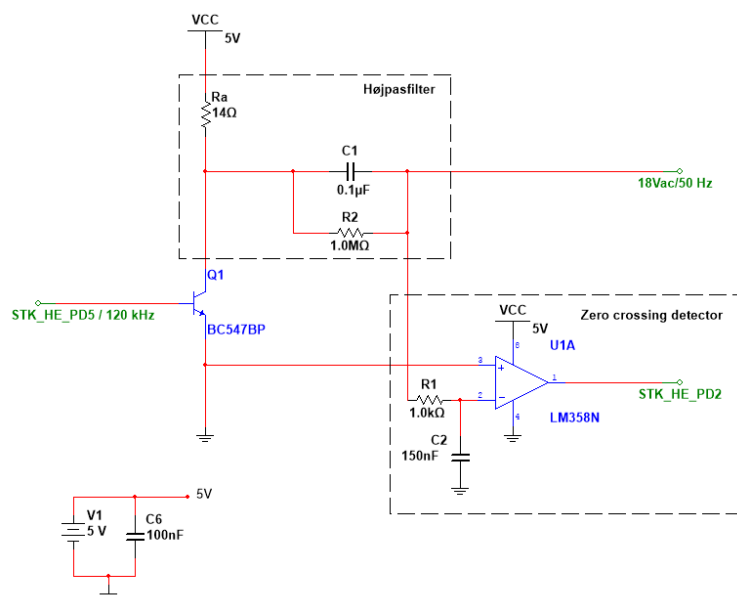
Encoderen, også omtalt som senderenhed, er den del af systemet der genererer X10 kommandoen og sender den ud på det eksisterende el-net. Et højpasfilter der lader 120 kHz burst passere mens det blokerer for nettets 50 Hz signal, udgør sammen med en zero crossing detector hele hardwaren for encoderen.



Figur 9.1. 120 kHz burst i zero crossing

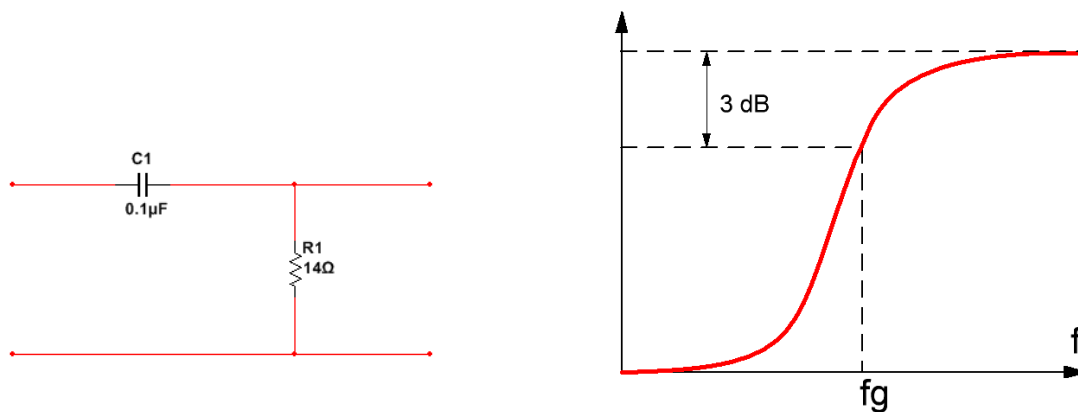
Ideen med at sende burst ud på nettet i zero crossing kan ses illustreret på figur 9.1

¹For yderlige beregninger se projektdokumentation afsnit HW-design



Figur 9.2. Samlet Encoder

9.2.1 Højpasfilter



Figur 9.3. Højpasfilter med værdier

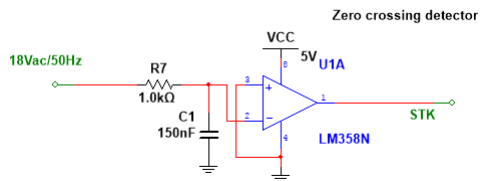
Figur 9.4. Kurvekarakteristik for højpasfilter

For at sende X10 kommandoer ud på det eksisterende 50 Hz el-net er det nødvendigt at koble elektronikken direkte herpå og eftersom det elektronik ikke tåler de høje spændinger fra nettet, er det nødvendigt at blokere det signal, men stadig at kunne sende de 120 kHz ud. Dette løses med et højpasfilter.

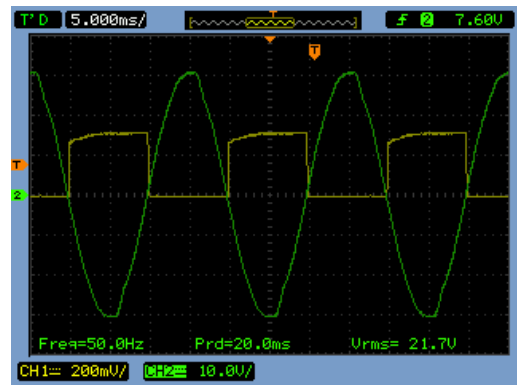
Den ønskede knæfrekvens skal ligge omkring de 120 kHz for at opnå mindst dæmpning herpå. Ved denne knæfrekvens skulle 50 Hz signalet være ubetydelig lille efter filteret.

Kondensatoren forudbestemmes for beregningerne til en værdi på 0,1 nF.

9.2.2 Zero Crossing Detector



Figur 9.5. Zero crossing detector med værdier



Figur 9.6. Scope billede af Zero crossing detector, CH1(indgangssignal), CH2(Udgangssignal)

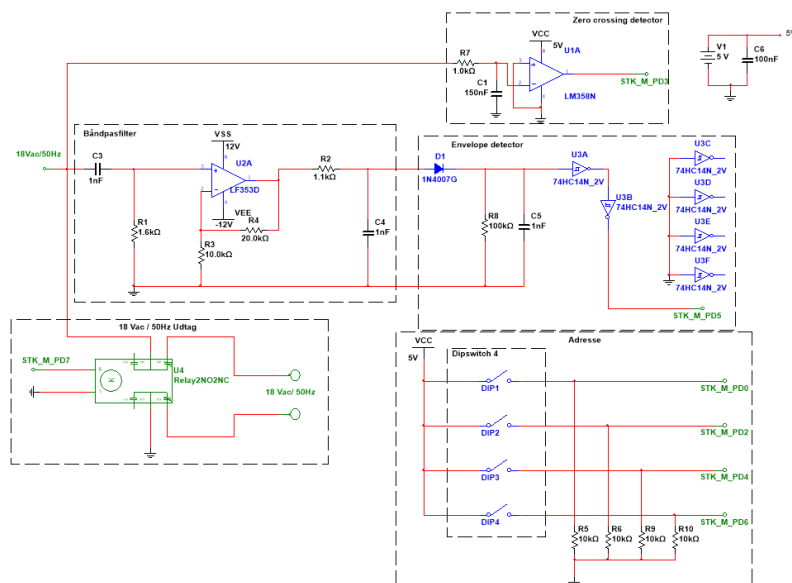
Zero crossing detectoren har til opgave at detektere nulgennemgang, det er et krav for at X10 protokollen kan virke. Der er placeret en Zero crossing detector både på Encoderen og Decoderen, da begge disse består af et STK-kit der kræver information om nulgennemgang. Opbygning kan ses på overstående figur ??, der er anvendt en operationsforstærker af typen LM358N, som toggler udgangssignalet ved hver nulgennemgang se figur 9.6. Operationsforstærkerens positive ben er koblet til stel for at lave et triggerniveau til 0 V.

Modstanden R_7 sidder der bl.a. for at beskytte Zero crossing detectoren mod 18 VAC nettet, men sammen med kondensatoren udgør den også et lavpasfilter. Under implementeringen kunne det konstateres at der kom støj ind på Zero crossing detectoren, og dette problem løste lavpasfilteret.

Der ønskes at dæmpe 120 kHz signalet, derfor designes lavpasfilteret ud fra en knækfrekvens på 1,0 kHz.

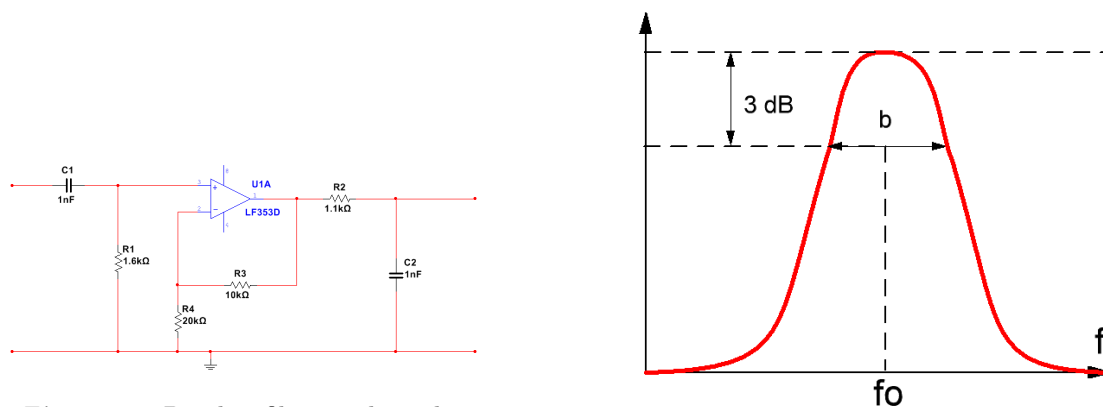
9.3 Decoder

Decoderen, som er den del i systemet der omdanner burst fra encoderen til X10 kommando, er opbygget af et båndpasfilter, zero crossing detector og en envelope detector. I starten af kredsløbet sidder båndpasfilteret, og dette blokerer for 50 Hz nettet og forstærker 120 kHz signalet der kommer fra encoderen. Herefter ledes signalet gennem envelope detector som omdanner burstet til et TTL signal.



Figur 9.7. Samlet Decoder

9.3.1 Båndpasfilter

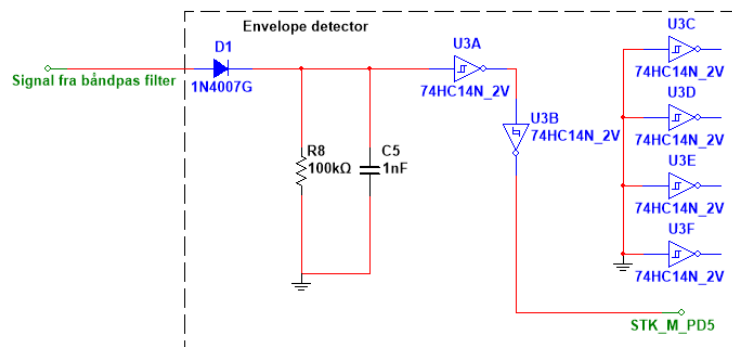


Figur 9.8. Båndpasfilter med værdier

Figur 9.9. Kurvekarakteristik for båndpasfilter

Båndpasfilteret har til opgave at filtrere alle signaler med frekvenser over og under 120 kHz fra samtidig med at det forstærker signalet i båndpasset. Forstærkningen opnås ved at koble en ikke inverterende OpAmp med modstande, hvis størrelse afhænger af den ønskede forstærkning, mellem et højpasfilter og lavpasfilter som illustreret på figur ???. Kurvekarakteristikken er illustreret på figur 9.9.

Da vi ønsker at forstærke 120 kHz signalet, designs højpasfilteret således at knækfrekvensen udregnes til 110 kHz, og for lavpasfilteret beregnes en knækfrekvens på 130 kHz.



Figur 9.10. Envelope detector

9.3.2 Envelope Detector

Envelope detectorens opgave er at udglatte burstsignalet fra båndpasfilteret og lave det om til et TTL (0-5V) signal som STK-kitten kan aflæse.

Den er opbygget af en diode, et RC-led og en schmitt trigger. Dioden har til opgave at sortere alle de negative halvperioder fra og kun sende de positive halvperioder til RC-leddet. Kondensatoren vil derfor kun opfange de positive halvperioder, og undgå at belaste det foranliggende kredsløb ved at aflade de negative halvperioder. Kondensatoren er med til at udglatte signalet da den ikke kan nå at aflades på en periode.

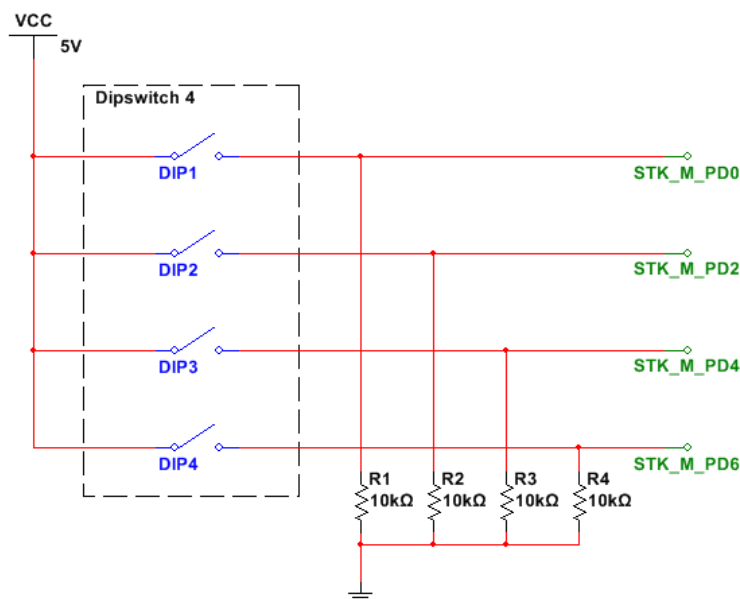
Modstanden R_8 er en afladningsmodstand, og den bestemmer hvor hurtigt kredsløbet skal aflades, jo højere modstand jo langsommere går det med at aflade.

Der er anvendt en schmitt trigger til at lave signalet om til et firkantet signal som STK-kitten kan aflæses. Grunden til at der er brugt 2 er fordi de er inverterende.

9.3.3 Dipswitch

For at en X10-kommando bliver udført af det korrekte udtag medsendes en adresse på fire bit. Til at simulere adresseringen for et udtag, er der lavet fire dipswitches der er forsynet med 5 V og forbundet til STK-500 modtageren, som det er illustreret på figur 9.11.

En åben kontakt vil give 0 og en lukket kontakt vil resultere i 1



Figur 9.11. Dipswitch for adressering af udtag

9.4 Software design

Med udgangspunkt i domænemodellen udviklet i arkitekturfasen er der udviklet applikationsmodeller for hver computer i systemet. Dette giver overblik over de funktionaliteter som skal implementeres på de forskellige platforme.

Applikationsmodellen består af at beskrive hvordan information fordeles i hvert UC. Dette opnåes med tre diagram typer. Sekvensdiagrammer som viser hvordan information bevæger sig sekventielt igennem systemets klasser, et klassediagram som sammenfatter de metoder og relationer som er fundet i sekvensdiagrammet og et tilstandsmaskinediagram som viser et systems forskellige tilstande. Det sidste er udeladt da det ikke er aktuelt for det opbyggede system.

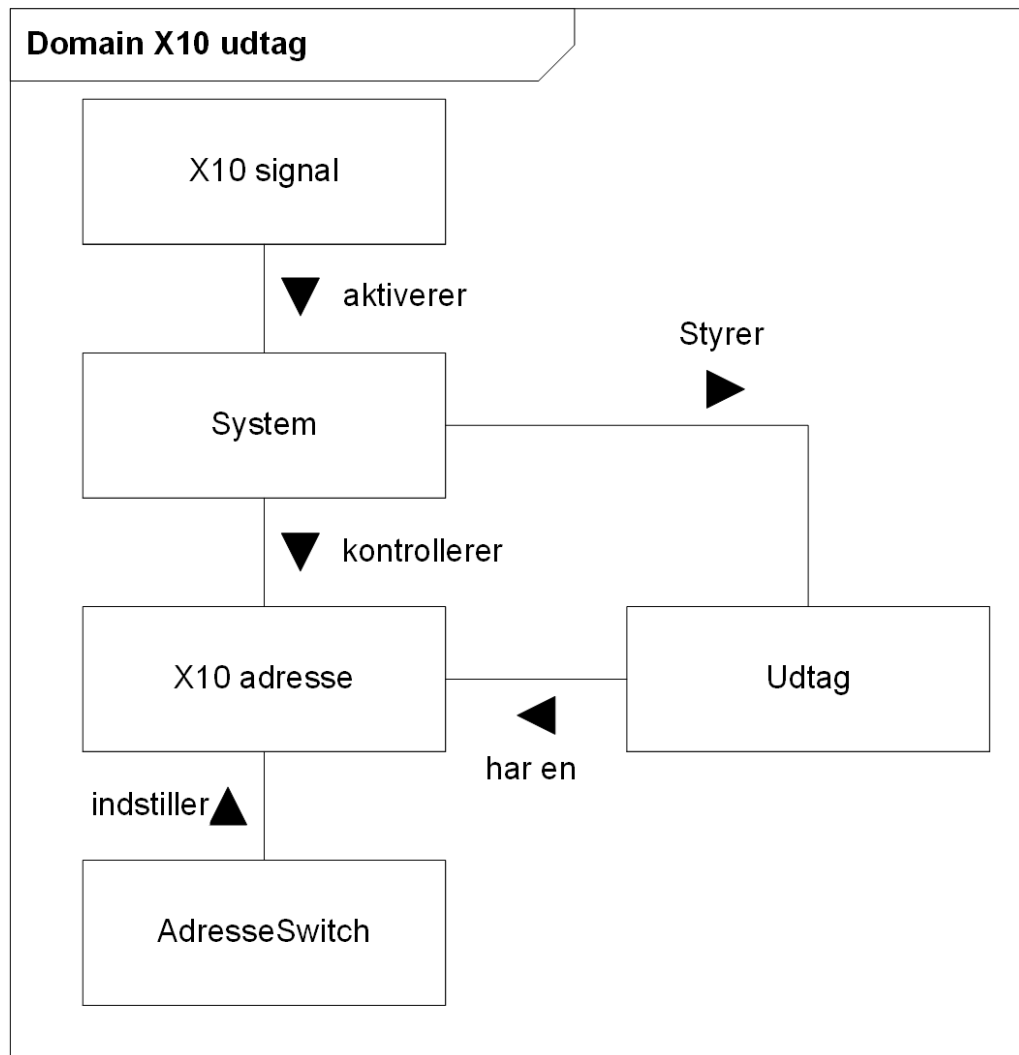
I det følgende vises applikationsmodellen for X10 udtaget. For modeller over CSS hovedenhed og PC henvises til projektdokumentationen.

9.4.1 Applikationsmodel for X10 udtag

Først er der lavet en detaljeret domænemodell for X10 udtaget. Denne er vist i figur 9.12. Denne laves ved at gennemgå UC beskrivelserne og finde de ting som har indflydelse på netop denne del af systemet.

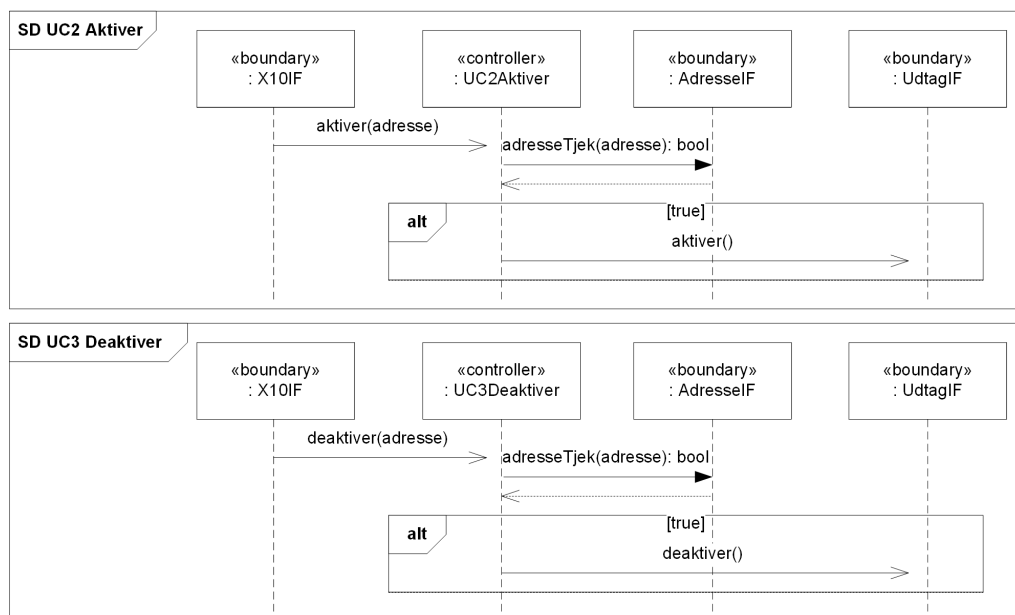
Med dette udgangspunkt laves der sekvensdiagrammer for hvert UC. Disse er vist i figur 9.13. De viser hvordan metodekald i mellem de konceptuelle klasser og giver et overblik over den basale funktionalitet.

Dette resulterer i et klassediagram med grundfunktionaliteten beskrevet, se figure 9.14. Denne bruges under implementeringen og ender ud i et statisk klassediagram som beskriver det endelige program med alle hjælpemetoder.

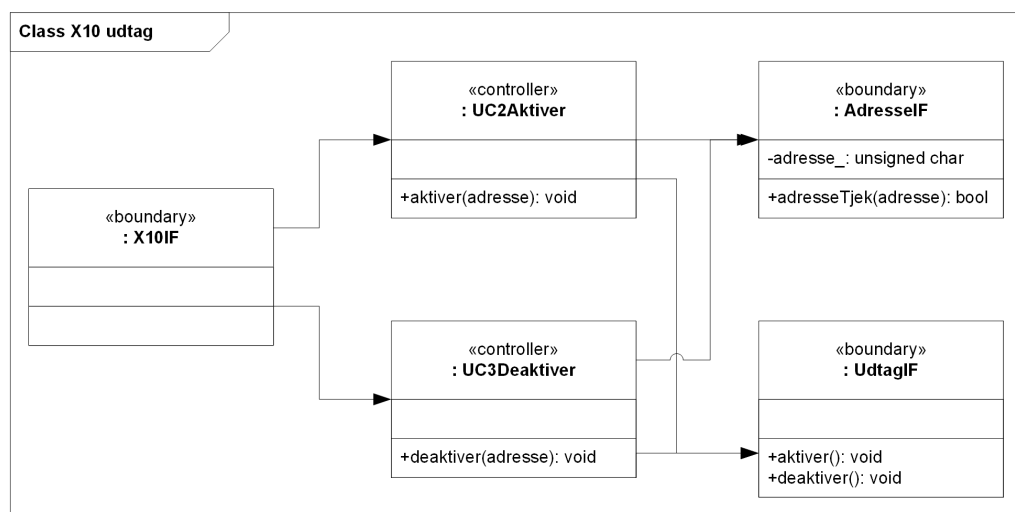


Figur 9.12. Domænemodel for X10 udtag

Denne analyse af funktionalitet giver et klart overblik til implementeringsfasen.



Figur 9.13. Sekvensdiagram for X10 udtæg



Figur 9.14. Klassediagram for X10 udtæg

Implementering 10

10.1 Hardware implementering

10.2 Software implementering

Resultater 11

Konklusion 12

13.1 Bjørn Sørensen

Min overordnede konklusion på dette projektforsøg er en rigtig lærerig og succesfuld implementering af alle de metoder og arbejdsredskaber som er blevet introduceret på dette semester. Det har været en rigtig spændende proces fra den indledende brainstorm til det endelige projekt og jeg syntes generelt gruppen har været rigtig god til at følge den samme struktur og arbejdsgang med udgangspunkt i ASE-modellen.

Med en baggrund som elektronikmand har jeg skulle koncentrere mig en del om ikke at blande mig i elektronikken da jeg har været i software gruppen og i øvrigt læser på IKT linjen. Dette har dog været en god lejlighed til at få afprøvet de grænseflade aftaler vi har indgået og stole på dem.

Vi havde i gruppen fra starten store ambitioner, men har i løbet af perioden fået skåret idéen helt ind til benet. Dette syntes jeg egentlig har været lidt ærgerligt da den oprindelige idé var rigtig god, men må også erkende at vi havde en fast deadline som ikke gjorde det muligt at nå helt i mål med idéen.

13.2 Jakob Schmidt

Sammenlignet med 1. semester projektet, så har processen i det her projekt været meget mere struktureret. Sammenholdet og overblikket mellem hvert projektmedlem har været rigtig godt. Det gode samarbejde og overblik skyldes at vi jævnligt har holdt møder, både som selvstændig gruppen, men også sammen med vejleder. Desuden er der afholdt enkelte trivsels runder, hvor vi hver i sær har skulle fortælle hvordan vi selv følte det gik i projektet.

Inden vi fik det endelige overblik over hvor omfattende vi kunne lave vores projekt, var det tydeligt at vi havde lidt for stor ambitioner til projektes omfang. Dette blev skåret ned efter første reviewmøde da vi fra anden vejleder blev anbefalet på det kraftigste at revurdere vores use cases.

Personligt har jeg beskæftiget mig med elektroniken i projektet, det kom som naturligt valg da jeg læser på elektro linjen. Fagligt har der været nogle komplikationer med at få de enkelte moduler til at virke som tiltænkt. Projektet har været spændende men samtidig udfordrende, specielt eftersom den nødvendige teori først var helt på plads i den sidste fase af forløbet.

Overordnet set er jeg rigtig tilfreds med resultatet af vores projekt, på trods af at vi langt fra fik realiseret alle de ting vi havde udtænkt fra første brainstormmøde. Efter revurdering

af projektets omfang er alle dele desværre stadig blevet færdige, men ideen er der og de vigtigste dele blev implementeret.

13.3 Jeppe Stærk

13.4 Jesper Christensen

13.5 Mick Kirkegaard

Vi besluttede i starten af projektfasen at bruge skriveredskabet Latex, og jeg blev lidt uventet primus-motor i dette værktøj. Dette har sammen med SVN-værktøjet Github givet en del hovedpine. Men jeg er rigtig glad for at være kommet igennem processen med disse værktøjer.

Selve projektet fik vi sparket igang med en fantastisk brainstorm med en masse gode ideér som vi hurtigt fandt ud af, var alt for optimistisk. Der måtte derfor hurtigt skrues ned for ideérne og fokuseres på hovedindholdet i projektet.

Jeg syntes processen har været rigtig sjov, det har været spændende at bruge de værktøjer man lærer på studiet til noget reelt elektronik. Bare det at forstå X10 kommunikationens virkemåde har været rigtig spændende for mig som Elektro studerende. Dette var en af nødderne der skulle knækkes før projektet virkelig blev spændende.

I hardwaregruppen har vi haft et tæt samarbejde og virkelig vendt og drejet alle aspekterne i projektet med hinanden. Og med vores ugentlige møder, har vi kunnet følge lidt med i software gruppens arbejde med projektet. Alt-i-alt et godt forløb. Savner dog et karaktergivende projekt næste gang, så man kan ligge endnu mere sjæl i det.

13.6 Poul Overgaard

13.7 Simon Kirchheiner

Vi har været syv mand i vores projektgruppe, og det har givet os mulighed for at øve os på at samarbejde med flere om et projekt. Vi har været en god projektgruppe med mange forskellige kompetencer, det har medført at man kan hjælpe hinanden godt. Det var en stor fordel at hele projektet var opdelt i faser og havde løbende møder og reviews, så folk ikke bare kørte på helt selv. Faserne har hjulpet med at få et overblik over projektet, og gjort at vi har tænkt over de ting vi skulle lave. I starten af projektet havde vi store ambitioner om at nå mange forskellige ting, men som tiden skred frem måtte vi se i øjnene at det ikke var realistisk.

I forhold til 1. semester har vi brugt mere tid på at dokumentere det vi skulle lave, og fået en forståelse for de forskellige arbejdsmetoder vi har lært i ISE. Jeg har været på HW delen og der har ASB/MSA fagene givet den nødvendige viden til at kunne løse de problemstillinger vi stod overfor. Jeg tog ansvaret for at skrive logbog/referater, det har givet et godt grundlag for at få struktureret arbejde, da alle kan gå ind i logbogen/referatet og se hvad vi snakkede om.

Vi fik et produkt næsten som planlagt, grundet tidsmangel fik vi ikke lavet vores lyddetektor, men vi fik kommunikation over AC nettet til at fungere korrekt. Jeg har fået et stort udbytte ud af dette projekt, og jeg synes vi har haft et godt samarbejde i projektgruppen. Generelt har gruppen fungeret godt og jeg mener vi har et rigtig godt produkt.

Litteraturliste 14

14.1 Bøger

14.2 Hjemmesider

14.2.1 Opslagsværker

Generelt C++ opslagsværk <http://www.cplusplus.com> [2014-05-24]