

Eksamensopgave - forside

Studerendes fulde navn: Daniel Mark Christensen

Studienr.: 20121049

Studieretning/plan: E-musik

Årgang (fx BA-2): BA-2

Dato: 01/12/2013

Opgavens titel: Individuel Praktikrapport

Fag: Pædagogik og formidling

Eventuel vejleder/underviser: David Mondrup

Antal anslag: 16011

Mailet kopi til biblioteket: ja nej sæt ét kryds

Mailet kopi til arkivet (krav): ja nej sæt ét kryds

Alt skriftligt materiale afleveres som udgangspunkt i studieadministrationen i to eksemplarer (ved intern censur)/tre eksemplarer (ved ekstern censur). Kontrollér dette i din studieplan.

Der ud over mailes ét eksemplar i pdf-format til studieadministrationens arkiv (opgaver-studieadm@musikkons.dk – dato, fag og navn skal fremgå af titlen) samt ét eksemplar i pdf-format til biblioteket (projekter-bib@musikkons.dk – dato, fag og navn skal fremgå af titlen).

Opgaver mailet til biblioteket vil være offentlig tilgængelige for studerende og undervisere på konservatoriet. Det er derfor vigtigt, at disse er anonymiseret, så der ikke fremgår personnavne i opgaven. Lydfiler skal fortsat afleveres i studieadministrationen i henholdsvis tre (intern censur)/fire (ekstern censur) eksemplarer.

Individuel Praktik-rapport

af Daniel Mark Christensen, DIEM BA2 2013

Målsætning:

Inden jeg fandt en elev havde jeg gjort mig nogle tanker om hvordan forløbet som udgangspunkt skulle foregå. Jeg ønskede at undervise ud fra mine spidskompetencer jeg har tilegnet mig indenfor visse områder. Disse er primært programmering (Max/MSP) og analoge kredsløb – og interaktionen mellem disse. Idéelt set skulle min ”kanin” være drevet af et ønske om at modtage indgående undervisning indenfor et felt jeg har erfaring indenfor.

Først havde jeg en aftale med en ung mand på 19 år fra kunstakademiet som ønskede at lære grundlæggende mestring af en DAW til musikproduktion, men efter længere overvejelser besluttede jeg mig for at holde fast i udelukkende at give undervisning indenfor programmering i Max og jeg smed et opslag op på diverse sociale medier for at finde en ny elev som ønskede at modtage undervisning i dette.

I min optik ville en realistisk målsætning først kunne udarbejdes efter min elev var fundet. Elevens forforståelse for området ville selvfølgelig have indflydelse på hvor omfangsrig en undervisning man ville kunne opnå på de begrænsede antal timer vi havde til rådighed. Dog havde jeg nogle grundlæggende mål jeg som minimum mente skulle være opnået ved udgangen af undervisningsforløbet:

1. Eleven skulle have en grundlæggende forståelse for programmets ”sprog”, dets syntax og brugerflade og kunne skelne mellem de forskellige grupper af kommando-eksekveringer.
2. Eleven skulle have kendskab til en bred vifte af objekter og forstå håndteringen af disse. Mestring af Max afhænger i høj grad af kendskab til programmets objekter for at kunne løse opgaven på den mest effektive måde.
3. Eleven skulle kunne benytte objekterne i en sammenhæng for at opnå de ønskede midler til funktionelle 'patches'.
4. Eleven skulle kunne bruge programmet til kreativ og musikalsk praksis og opnå mindst én fungerende patch til et ønsket projekt.

Beskrivelse af eleven:

Egentlig havde jeg forestillet mig at have en elev som var mere eller mindre uerfaren indenfor programmering men det viste sig at en af mine egne ”undervisere” som hjælper mig med programmering af mikroprocessorer altid har ønsket at blive sat ind i Max-miljøet.

Hans navn er Carl Frischknecht Jensen, han er lige godt 30 år gammel og er tidligere datalogi-studerende. Han er selvstændig erhvervsdrivende hvor han primært beskæftiger sig med web-design og IT-support – han har også en mindre webshop hvor han tilbyder elektroniske komponenter, microprocessorboards og andre materialer til elektronik.



Carl Jensen

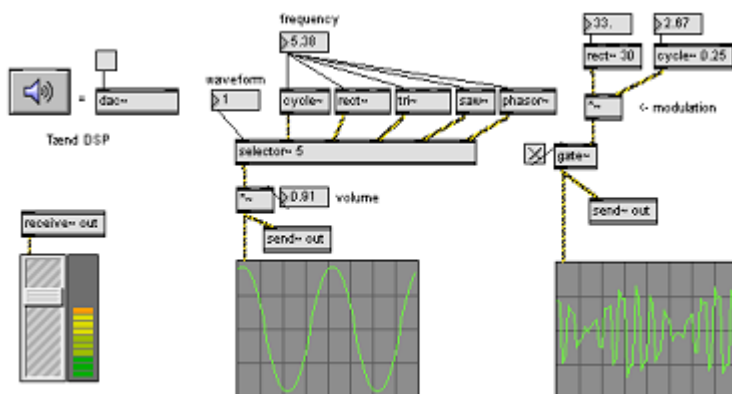
Selv mødte jeg ham under en workshop han afholdt under sit projekt 'DimsOs' hvor han og en anden (uddannet elektriker) underviser i at bygge simple analoge synthesizere. Siden da har jeg været eleven og han har været underviseren men nu var der pludselig noget jeg kunne lære ham, hvilket var en stor tilfredsstillelse for mig – at kunne give lidt tilbage. I starten havde jeg egentlig nogle betænkeligheder ved at have ham som elev, da jeg var i tvivl om han overhovedet ville kunne lære noget nyt, da han mestrer programmering i C og Java på hvad jeg vil kalde ekspert-niveau. Det skulle vise sig at det var min egen uvidenhed om de forskellige miljøers ligheder der lod tvivlen råde. For ham var det en meget ny verden med en mere eller mindre ny syntax.

Derudover er han trækbasunist og har spillet i big band. Dette er et plus da han derved har en musikalsk interesse men alligevel ikke har erfaring med digital signalbehandling eller produktion; så der var nogle principper og teknikker, som næsten er blevet til aksiomer for en erfaren elektronisk musiker, som det var nødvendigt at forklare – hvilket jeg gav mig selv en større forståelse eller fornyet syn på teoretiske principper jeg efterhånden ikke stiller spørgsmål ved.

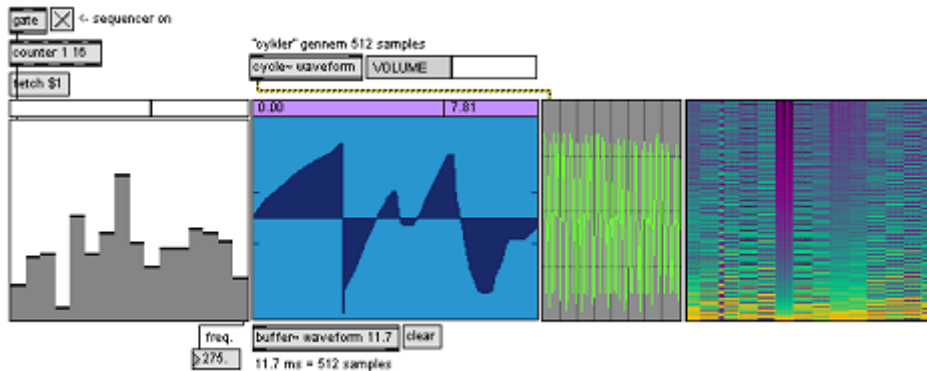
Dagbog:

1. **21/10** - Introduktion og opsætning af programmet. Brugerfladens opbygning.
Carl medbringer sin computer, Max installerer med studie-licens for 1 år. Jeg fortæller lidt om historien bag programmet og udviklingen til i dag. Vi gennemgår programmets opsætning. Carl kigger lidt på det derhjemme til næste gang.
2. **28/10** - De forskellige elementer - bang, toggle, nummerbox, float/int, messages
Der introduceres til programmets struktur, de forskellige elementer til beskedstyring; nummerbokse, 'bang'-eksekvering, forbindelser via visuelle "kabler": programmets "sprog". Carl drager paralleller til traditionel programmering, hvilket starter en interessant samtale op. Carl får lektier for til næste gang: at bygge en patch selv ved hjælp af de hidtil gennemgåede komponenter.
3. **31/10** - Max-objekter. Forbindelser, inlets, outlets.
Vi dykker ind i objekterne og dertilhørende dokumentation i form af eksempler på brug af hvert enkelt objekt. Vi ser på hvordan man kan "tale" til objekterne og hvide data ud af dem ved hjælp af ind-/udgange. Jeg er opmærksom på at gå pædagogisk og nænsomt frem da dette er en stor og ny verden for Carl.
4. **5/11** - MSP-delen, oscillatorer, oscilloskop, frekvensstyring.

Vi tager hul på signal-processeringsdelen; MSP. Vi sætter lydkortet op og får "hul igennem". Vi kigger på MSP-objekter: forskellige oscillatorer og omskiftere. Dette vækker Carls interesse, jeg kan mærke han er meget inspireret af at bruge Max til lyd. Carl får lektier for til næste gang ud fra de gennemgåede dele.



5. **8/11** - Musikalsk brug af MSP – AM, FM, envelope (line), pseudo-VCA.
Jeg inddrager en musikalsk vinkel til signal-processing. Dette kræver forklaring af visse grundprincipper indenfor elektronisk lyd-syntese. Vi gennemgår modulation ved hjælp af oscillatorer og gates, envelopes osv. Carl får lektier til næste gang, opgaven går på at prøve at genskabe de basale moduler i en analog synthesizer.
6. **13/11** - Filtre, spektrogram, sequencere, wavetable osc med buffer~



Vi starter med at gennemgå lektierne fra sidste gang og udrede problemer der opstod og kommer sammen frem til løsninger på de ønskede mål han har haft. Vi bygger i dag en step-sequencer og en 'wavetable' synth hvor jeg introducerer til buffer~ objektet, som er en allokering i hukommelsen til samples. Dette finder Carl interessant fra en datalogisk vinkel og vi drøfter mulighederne ved dette. Vi kigger også på filtre og sonogrammet hvilket begge er nye for Carl så dette er meget inspirerende for ham.

7. **16/11** - Indlæsning af samples i buffere, play-objekt, adc~ og record~
David kommer forbi og følger undervisningen og tager noter. Det er oplagt at uddybe netop denne time da vi tog hul på en noget mere kompleks del af Max – nemlig sampling som kræver mere omfattende patches.

- 15.49: Carl ankommer, får max patch på USB stick
- 15.52: Vi går over til studiecomputeren, går i gang med at gennemgå dagens patch. Dagens overskrift: Sampling. Gennemgår (repeterer) buffer~.
- 15.56: Længdeattribut i buffer-objekt: Carl sammenligner med in/out i anden programmering, hvor man kan kalde dele af en datamængde.
- 16.00: Subpatcheren forklares, hvordan man kan lave under-patches i hovedvinduet.
- 16.03: Prøver at få lyd ud. Det lykkes ikke – troubleshooting.
- 16.06: Problemet løst – vi læste fra en forkert buffer ... bruger tid på at få forklaret hvad problemet var. Vi vælger at gribe chancen til at få større forståelse for programmets virkemåde og digital audio i det hele taget.
- 16.09: Jeg forsøger at forklare sammenhængen mellem pitch og tempo i digital lydafvikling. Her kommer jeg lidt ud i nogle problemer da jeg begynder at tvivle på hvordan det nu lige er det hænger sammen.
- 16.12: Brug af line-objekt til at undgå klik ved retrigging af samplet.
- 16.16: Play-objekt gennemgås til at afspille samples fra vores buffer.

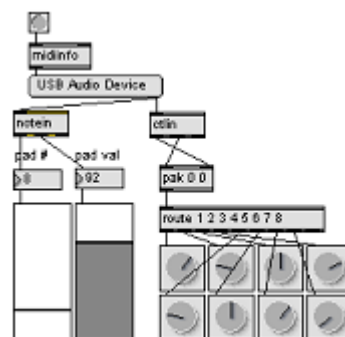
- 16.18: adc/dac objekt gennemgås.
- 16.20: Der rykkes over på Carls computer og Carl overtager styringen. Der prøves forskellige måder at få buffer objektet til at læse, markere og afspille audiofiler.
- 16.26: sfrecord~ objekt gennemgås – hvordan vi optager til en lyd-fil på disken.
- 16.33: Vi optager ind i forskellige positioner i bufferen.
- 16.36: Carl ytrer ønske om at arbejde med at bruge Max til at emulere Monome via en anden type controller. Snak om mulighederne ift. MIDI.
- 16.40: Næste gang: mere sampling, variable afspilningshastigheder m.m. Invitation til at komme med idéer.

David kommenterer bagefter lektionen at jeg bør lade Carl sidde ved computeren, da jeg hidtil har haft hånden på musen og tastaturet og har vist de forberedte patches til ham. Efter det, kan jeg godt selv se at det er skørt ikke at lade ham sidde med programmet fra start til slut – på den måde tvinges han konstant til at opbygge komfortabilitet med programmet.

David nævner også at Carl er kompetent og hurtig bag computeren, men at en mere ”grøn” computerbruger hurtigt ville blive tabt. Jeg har dog dette i baghovedet, da jeg har valgt at tilpasse denne mere tjeppede undervisningsform til Carl så vi hurtigt kan nå langt med programmet.

8. **22/11** - MIDI beskeder med LPD8, henblik på monome klon.

Jeg har forberedt en rimelig kompleks patch til denne gang hvor jeg har sat en midi-controller med 8 tryk-sensitive pads og 8 drejknapper. (Akai LPD8) Vi ser på mulighederne med denne specifikke controller men drøfter også mulighederne generelt med midi-beskeder ind/ud af max. Vi bygger sammen en step-sequencer som styrer nogle oscillatorer og Carl sidder længe og leger og afprøver patchen.



9. **26/11** - Effekter: tape delay, convolution med pfft, fffb~

Jeg vælger til denne gang at introducere til fft og et par effekter (ekko og filterbank) David sidder med denne gang og tager noter. Carl har medbragt controlleren fra sidst men vi får ikke tid til at lege med den i dag.

- 15.03: Dagens patch: effekter.
- 15.05: Patch åbnet, lidt snak om hvad der kunne have været inddraget (polyfoni) som jeg havde problemer med at få til at virke så dette må forberedes til en anden gang.
- 15.08: Vi kigger på ”tapin/out” delay. Forklarer opbygningen ud fra analogt tape delay (som vi før har bygget sammen). Tonal effekt ved kort delay time. 30 ms grænse – analogi til billeder/film. Carl er meget betaget af mulighederne med dette objekt og vi leger med den indbyggede mic.

- 15.30: Jeg introducerer til bandpass-filterbank og forklarer lidt om fft-filtre. Vi leger og justerer på filtrene og drøfter mulighederne ved denne effekt. Carl er igen inspireret og man kan mærke at han ser muligheder i materialet.
- 15.57: Jeg viser ham pfft~ som er et objekt til fft-transformering af signaler. Vi snakker noget tid om fft og her er der matematiske oversættelser jeg ikke selv forstår så godt, så Carl kan kaste lidt lys på sagen og vi får begge noget ud af udforskningen af objektet.
- 16.14: Jeg viser et eksempel som gør brug af pfft~ objektet: convolution vha fft. Her "morpher" vi to signaler sammen via deres frekvens og fase. Vi leger lidt med at "åbne" for signalet på en lydfil med den interne mikrofon som gate. Carl drager parallel til vocoderen og vi får en snak om denne.
- 16.17: Vi snakker om inkorporeringen af midi-controlleren til at styre de effekter vi har set på. Snakker lidt om hvad der skal ske næste gang, vi aftaler at se på nogle "audio descriptors" som analyserer lydsignaler vha fft.

David kommenterer efter timen at det er godt at Carl nu har fået styringen, så han får 'reel erfaring'. Han bemærker også at der er god tid til at lege med programmet og at få legen og det kreative kædet ind.

Han udpeger dog at jeg måske skulle sætte ham på nogle flere opgaver, få ham til at bygge nogle patches selv så jeg ikke har gjort alt arbejdet for ham. Dette er en god pointe, han vil kunne lære meget af at blive sat på en opgave han skal løse ved hjælp af de objekter og værktøjer han nu kender.

10. 27/11 - bonk~, bark~, fiddle~, shifter~

Vi ser på disse 4 objekter som beskriver et givent lydsignal – amplitude detektion som kan bruges til at 'trigger' hændelser, beskrivelse af signalet i fft frekvensbånd og pitch-analyse + pitch-shifting. Vi leger lidt med bonk~ til amplitude detektion for at sætte optagelse af mikrofonen i gang og med at pitch-shifte real time signal fra mikrofonen.

11. 30/11 - Arduino over serial med mux shield til fremtidigt projekt.

Denne gang har vi aftalt at se om vi kan smelte de to verdener sammen (analoge signaler og Max) ved hjælp af et arduino board med multiplexere som muliggør at kunne sende mange signaler ind på én gang. Vi har længe snakket om at bygge en stor controller med den høje opløsning (10 x midi-opløsning!) Dette lykkes rimelig godt, vi får hul igennem til et analogt signal gennem et potentiometer og ser på eksempel-patches vi henter fra nettet som viser hvordan man kan tale med serial-porten og modtage beskederne fra arduinoen.

12. 2/12 - Poly~ til at trigger bonk-optagede samples med LPD8 (ikke afholdt endnu)

Vi har aftalt at prøve at se på polyfonisk afspilning, som vi ikke fik til at virke tidligere. Slutresultatet skulle gerne være et færdigt selvstændigt "instrument" i form af Akai LPD8-controlleren og en sammenstykket patch ud fra de moduler vi før har gennemgået.

Retrospektivt set, har det været et givende og dynamisk forløb. Både min elev og jeg har fået stort udbytte af undervisningen. Det har været en fornøjelse at have en elev med så meget baggrundsviden og erfaring indenfor programmering, så man hurtigt kunne nå langt i processen. Det kunne dog også være spændende at have med en helt nybegynder at gøre, det ville sætte formidlingen og overskueligheden i centrum og man ville som formidler ikke få lov at slippe afsted med overfladiske forklaringer af et område eller skippe vigtige vadestener til målet.

Jeg er nu opmærksom på nogle mangler i forløbet som jeg vil forbedre til næste gang: Jeg fokuserede meget på at forberede materialet, fra bunden af, så der ikke var nær så megen tid til at reflektere videre over opbygningen af elevens praktiske erfaring. Det vigtigste var på givende tidspunkt at vise programmet og præsentere mulighederne – efter David Mondrups kritik kan jeg se at jeg skulle overlade mere af det i hænderne på min elev. Det har helt klart fremskyndet hans fortrolighed med programmeringsmiljøet efter jeg har ladet ham få styringen. Derudover kunne man næste gang lægge mere vægt på at lade eleven nå frem til løsninger ved hjælp af de objekter der har været fokus på. Man kunne opstille enkle opgaver eller ligefremt lade eleven selv komme med en idé til et ønsket projekt og lade vedkommende prøve at nå dertil selv, for derefter at tage de relevante problemstillinger og blokader der opstår op og løse dem sammen bagefter.

"Hvad jeg hører, glemmer jeg, hvad jeg ser, husker jeg, hvad jeg gør, lærer jeg." - Platon

Det har været overraskende berigende for min egen forståelse af programmet at skulle italesætte koncepterne og principperne – jeg har været nødsaget til at sætte mig dybere ind i stoffet for at kunne formidle det på en præcis og fyldestgørende måde. Som Joseph Joubert sagde, at undervise er at lære to gange.

Set ud fra målsætningen, har eleven nu godt styr på objekterne og elementerne der skal til for at sammensætte funktionelle patches – der har også været lidt tid til at udforske de kreative og ekspressive aspekter undervejs. Planen er at få en færdig patch op at stå i løbet af næste/de næste gange og dykke mere ned i interaktionen; der opstår nemlig ofte idéer til finjusteringer og småforbedringer undervejs når man for alvor får afprøvet det færdige "instrument". Man kan godt argumentere for at selve programmeringsdelen efterhånden er blevet så kreativ at det i sig selv er kan agere som selvstændig kunstform. Man tager nogle delvist ubevidste valg når man opbygger en patch, visuelle som funktionelle – og det at udforske miljøets potentiale åbner næsten uden undtagelse op for øjeblikke af kreativ udfoldelse og åbenbaringer ved mødet med en ny teknik.

Da jeg har et godt samarbejde med Carl fra tidligere, vil vores fælles forståelse af Max/MSP gavne vores arbejdsprocesser i forbindelse med projekter der involverer interkommunikation mellem den analoge og digitale verden. Dette forløb er blot et springbræt til en længerevarende kreativ proces.