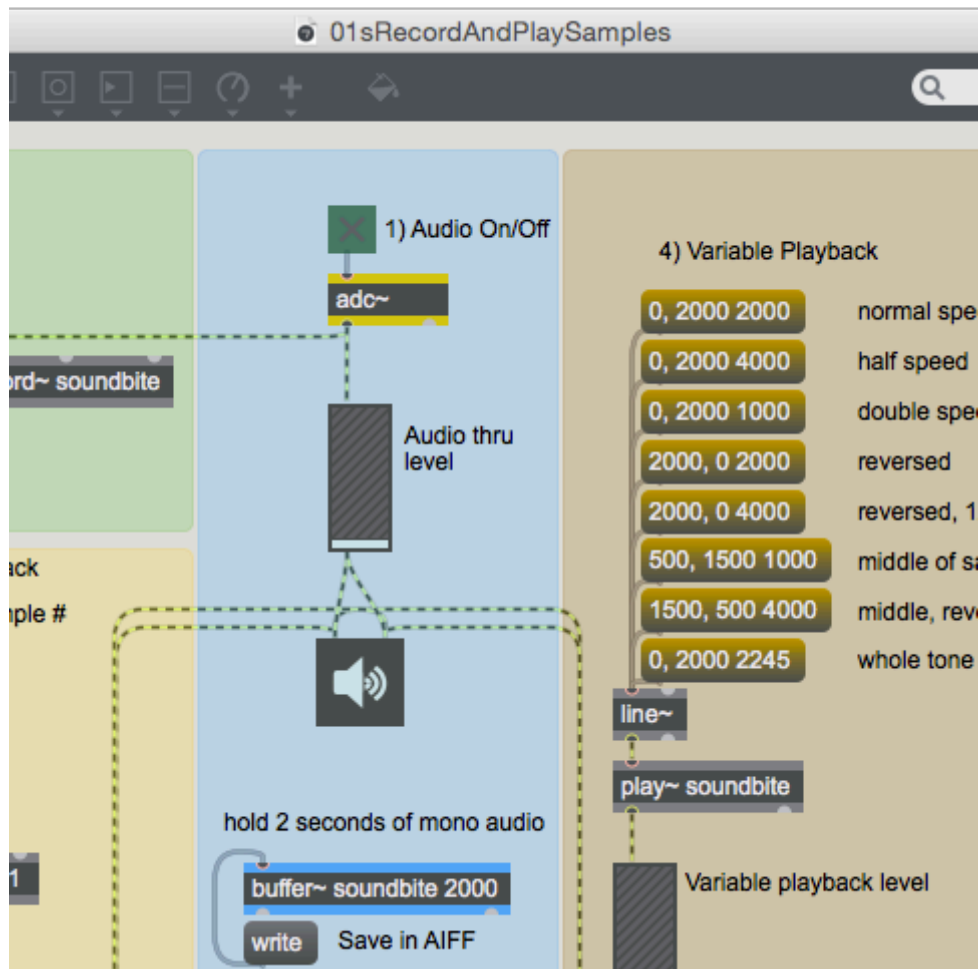


Kontrolloppgave til øvelsen *Record and Play Samples*

Denne øvelsen er den første MSP-øvelsen der vi arbeider med innspilt lyd (Hammer s. 40-46 og Roads s. 117). Vi skal se på hvordan vi lagrer lyden i en buffer og hvordan vi kan manipulere denne lyden etterpå.

Gå gjennom øvelsen *Record and Play Samples* og den tilhørende øvelsespatchen. Om det er noen punkter du er usikker på, gå tilbake og repeter det som er uklart. Om det er et objekt du ikke er helt sikker på, les referansen eller hjelpefilen til objektet.



Gå igjennom de ulike delene av patchen og vær sikker på at du skjønner hvordan alle delene fungerer. Når alt er klart, gå videre til oppgaven.

Oppgave 44

44.1 Lag en patch der du kan spille inn lyd fra en mikrofon, og spille den av igjen etterpå. Bruk *play~*-objektet og *index~*-objektet til å spille av lydfilen.

44.2 Legg inn en knapp som starter avspillingen av lyden 44100 samples inn i lyden. Bruk *index~*-objektet for å gjøre dette.

44.3 Legg inn en knapp til som starter avspillingen av lyden etter et halvt sekund. Bruk *index~*-objektet her også.

44.4 Ved hjelp av *play~*-objektet, legg inn knapper som utløser følgende seks avspillingsvarianter:

Normal hastighet

¼ hastighet

fire ganger normal hastighet

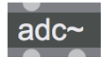
baklengs i ¼ hastighet

spill av fra 250 ms til slutten av lydfilen i normal hastighet

spill av fra 1000 ms baklengs til begynnelsen av lydfilen i halv hastighet.

Nye objekter introdusert i denne øvelsen:

adc~



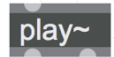
ezdac~



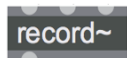
index~



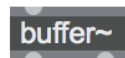
play~



record~



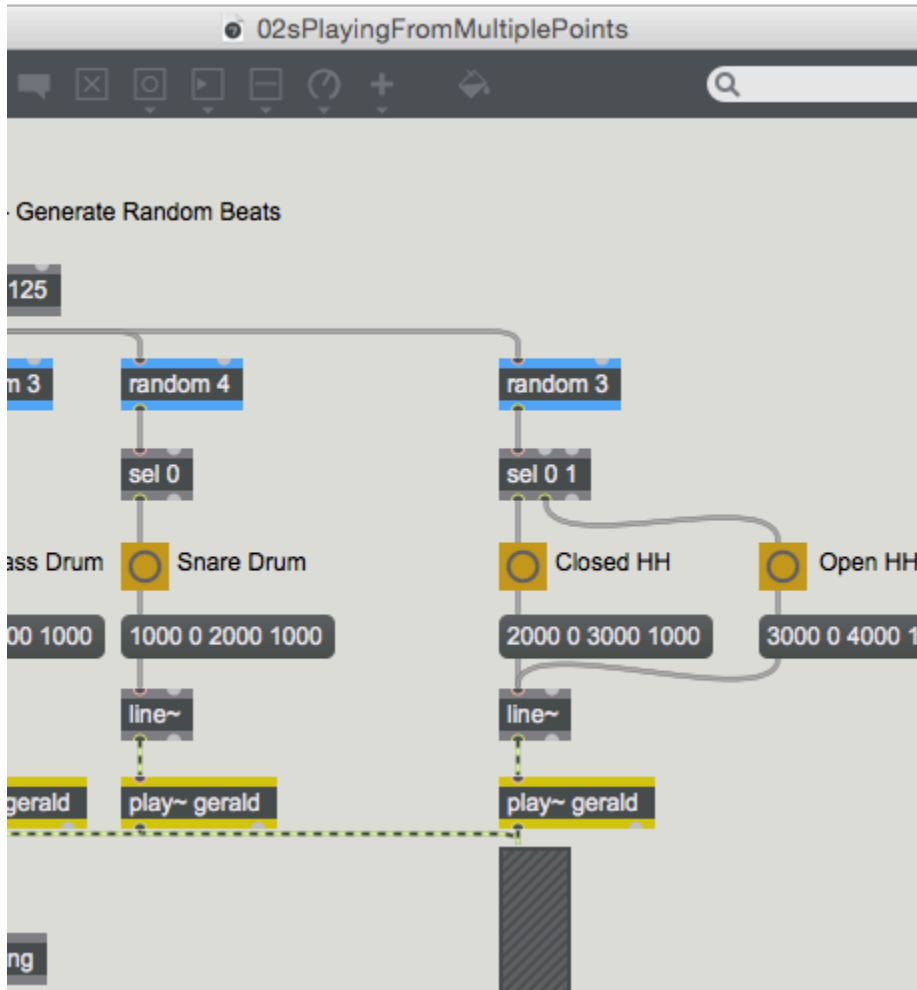
buffer~



Kontrolloppgave til øvelsen *Playing From Multiple Sample Points*

I denne øvelsen skal vi blant annet se på hvordan vi kan bruke flere objekter til å lese fra den samme bufferen.

Gå gjennom øvelsen *Playing From Multiple Sample Points* og den tilhørende øvelsespatchen. Om det er noen punkter du er usikker på, gå tilbake og repeter det som er uklart. Om det er et objekt du ikke er helt sikker på, les referansen eller hjelpefilen til objektet.



Gå igjennom de ulike delene av patchen og vær sikker på at du skjønner hvordan alle delene fungerer. Når alt er klart, gå videre til oppgaven.

Oppgave 45

Ta utgangspunkt i øvelsen du nettopp har gjennomgått.

45.1 Som du hører starter ikke avspillingspunktene i øvelsen på riktig sted i slaget. Forandre startstedet slik at de starter med en fjerdedels mellomrom.

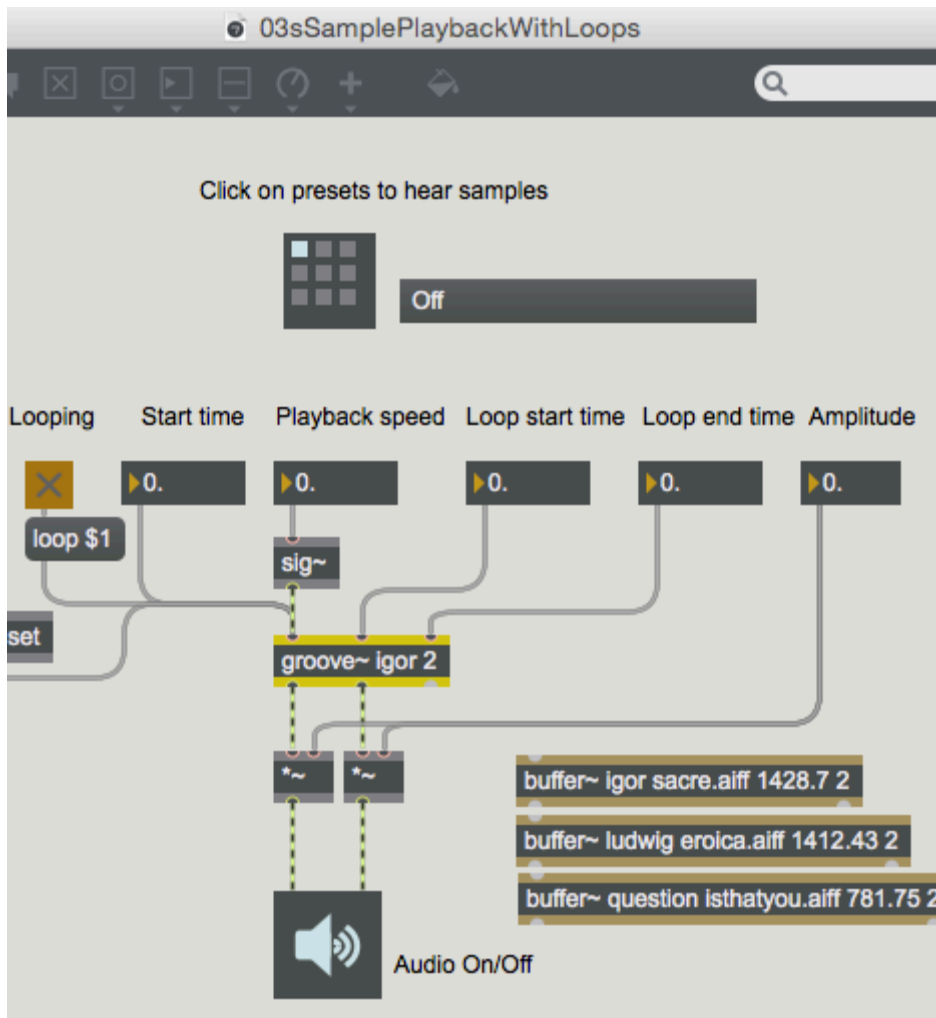
45.2 La alle fire variantene spille av fra samme *play~*-objekt. Still inn hastigheten på *metro*-objektet slik at det spiller av samples i et jevnt 4/4 tempo.

45.3 For hver gang et samplepunkt trigges skal det være fra et nytt tilfeldig sted som ikke ble brukt forrige gang. For å få til dette må du bruke *urn-jb*-objektet.

Kontrolloppgave til øvelsen *Sample Playback with Loops*

Denne øvelsen går videre med lydfilavspilling. Denne gangen skal vi blant annet se på ulike fleksible måter vi kan spille av en buffer i en løkke (loop) (Roads s. 121).

Gå gjennom øvelsen *Sample Playback with Loops* og den tilhørende øvelsespatchen. Om det er noen punkter du er usikker på, gå tilbake og repeter det som er uklart. Om det er et objekt du ikke er helt sikker på, les referansen eller hjelpefilen til objektet.



Gå gjennom de ulike delene av patchen og vær sikker på at du skjønner hvordan alle delene fungerer. Når alt er klart, gå videre til oppgaven.

Oppgave 46

Bruk det medfølgende lydopptaket som inneholder frasen «is that you?»:

Lag tre løkker der ordene «is», «that» og «you» repeteres.

Bruk de samme tre løkkene en gang til, men denne gangen skal «is», «that» og «you» spilles av baklengs.

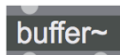
Lagre alle seks løkkene i et eget *preset*-objekt.

Viktige objekter:

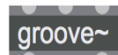
sig~



buffer~



groove~



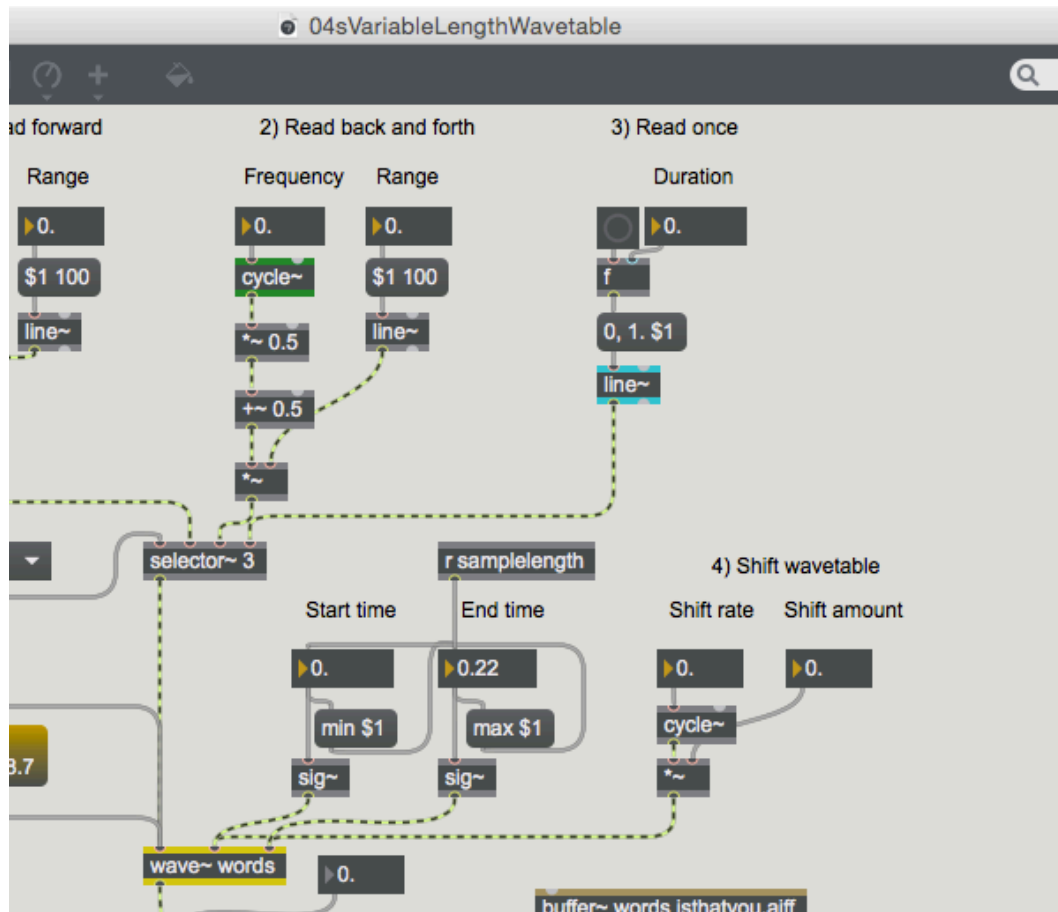
info~



Kontrolloppgave til øvelsen *Variable Length Wavetables*

I denne øvelsen går vi inn på mer dynamiske måter å lese bølgetabeller på.

Gå gjennom øvelsen *Variable Length Wavetables* og den tilhørende øvelsespatchen. Om det er noen punkter du er usikker på, gå tilbake og repeter det som er uklart. Om det er et objekt du ikke er helt sikker på, les referansen eller hjelpefilen til objektet.



Gå igjennom de ulike delene av patchen og vær sikker på at du skjønner hvordan alle delene fungerer. Når alt er klart, gå videre til oppgaven.

Oppgave 47

47.1 Lag en kurvetabell som inneholder lydfilen «isthatyou.aiff»:

47.2 Les denne tabellen fra 102 ms til 109 ms.

47.3 Les den med et *cycle~*-objekt (en oscillator som bruker cosinus). Oscillatoren skal ha en frekvens på 110 Hz.

47.4 Amplituden på oscillatoren skal varieres over tid. I løpet av ett sekund skal den gå fra amplitude 0.12 til amplitude 0.01. på denne måte varierer vi spekterkvaliteten fra rikt spekter til overtonefattig spekter.

Patchen skal altså inneholde følgende verdier:

- Lydfil i buffer: «isthatyou.aiff»
- Loop start-tid: 102 ms
- Loop slutt-tid: 109 ms
- Frekvens: 110 Hz

- Lengde på amplitude-omhyllingskurven: 1 sekund
- Amplitude på omhyllingskurvens startverdi: 0.12
- Amplitude på omhyllingskurvens sluttverdi: 0.01

Nye objekter introdusert i denne øvelsen:

buffer~

buffer~

phasor~

phasor~

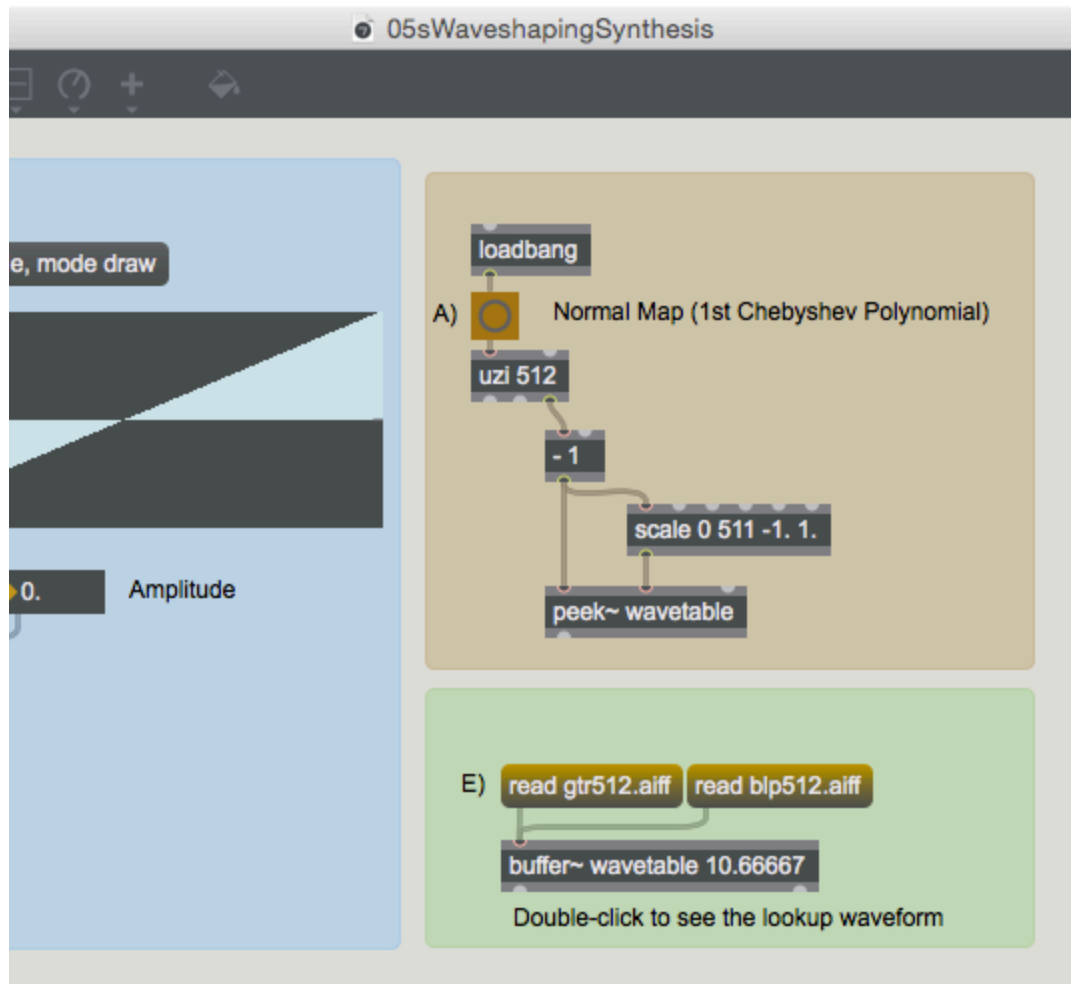
wave~

wave~

Kontrolloppgave til øvelsen *Waveshaping Synthesis*

Kurveformsyntese (Waveshaping) (Hammer s. 48 og 54 og Roads s. 252) er en enkel form for syntese som krever lite regnekraft. I denne øvelsen skal vi se på ulike måter å forme klangen på ved hjelp av denne synteseformen.

Gå gjennom øvelsen *Waveshaping Synthesis* og den tilhørende øvelsespatchen. Om det er noen punkter du er usikker på, gå tilbake og repeter det som er uklart. Om det er et objekt du ikke er helt sikker på, les referansen eller hjelpefilen til objektet.



Gå igjennom de ulike delene av patchen og vær sikker på at du skjønner hvordan alle delene fungerer. Når alt er klart, gå videre til oppgaven.

Oppgave 48

Ta utgangspunkt i øvelsen du nettopp har gått igjennom. I patchen er det eksempel på bruk av lydfilene «gtr512.aiff» og «blp512.aiff» i en buffer. Skift ut lydfilen i bufferen med lydfilen «bd+hh.aiff». Denne lydfilen ligger i den samme mappen som de to andre lydfilene, så du trenger ikke bekymre deg for at programmet skal finne den.

I stedet for å spille av en sinusbølge gjennom bølgeformingen, bruk en lydfil. Du kan bruke en av lydfilavspillingsteknikkene du har gjennomgått tidligere, eller bare dra en lydfil fra *Audio*-menyen til venstre i vinduet (den som er formet som en note). Velg for eksempel «Cherokee». Bølgeformingen fungerer nå som en type forvrenging.

Nye objekter introdusert i denne øvelsen:

lookup~

lookup~

waveform~



peek~

peek~

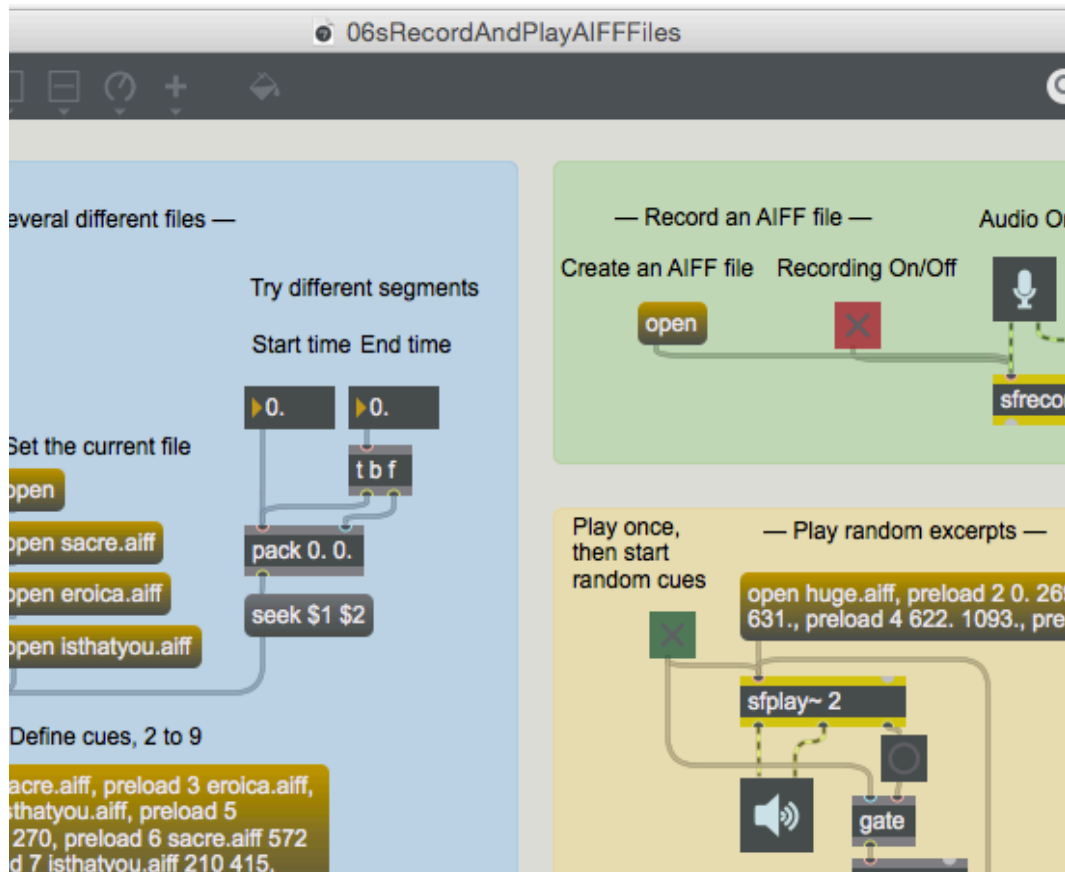
sampstoms~

sampstoms~

Kontrolloppgave til øvelsen *Recording and Playing Soundfiles*

Ved hjelp av objektene *sfplay~* og *sfrecord~* kan vi håndtere lyd inn- og avspilling (Hammer s. 40-46) på en enklere måte enn med bruk av en buffer samtidig som vi ikke legger beslag på store mengder minne.

Gå gjennom øvelsen *Recording and Playing Soundfiles* og den tilhørende øvelsespatchen. Om det er noen punkter du er usikker på, gå tilbake og repeter det som er uklart. Om det er et objekt du ikke er helt sikker på, les referansen eller hjelpefilen til objektet.



Gå igjennom de ulike delene av patchen og vær sikker på at du skjønner hvordan alle delene fungerer. Når alt er klart, gå videre til oppgaven.

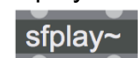
Oppgave 49

49.1 Lagre tre *cues* fra lydfilen «isthatyou.aiff». De tre ulike cuene skal peke til fragmentene «is», «that» og «you?» fra setningen.

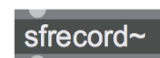
49.2 Ved hjelp av en tilfeldighetsfunksjon skal de tre cuene spilles kontinuerlig og i tilfeldig rekkefølge helt til du skruer dem av.

Nye objekter introdusert i denne øvelsen:

sfplay~



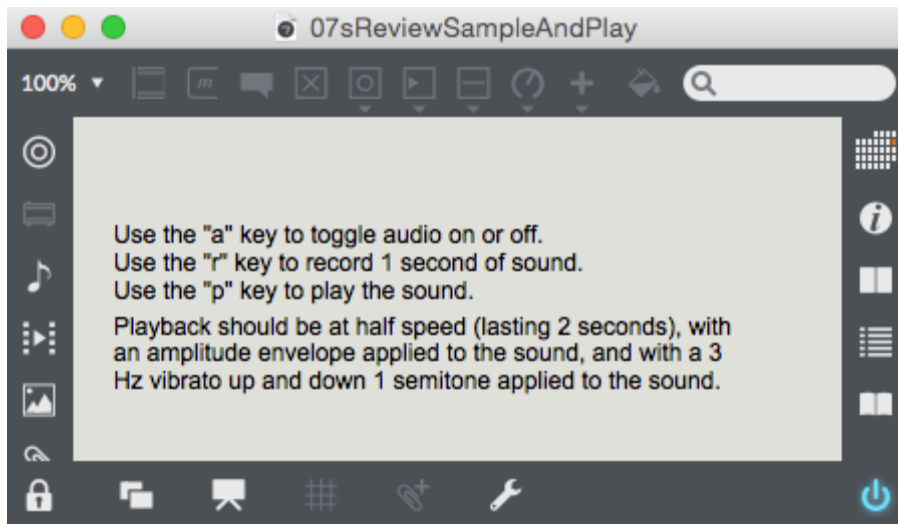
sfrecord~



Kontrolloppgave til øvelsen *Sampling Review*

Denne øvelsen repeterer endel aspekter ved bl.a. lydopptak (Hammer s. 40-46) som vi har gjennomgått i de siste øvelsene.

Gå gjennom øvelsen *Sampling Review* og den tilhørende øvelsespatchen. Om det er noen punkter du er usikker på, gå tilbake og repeter det som er uklart. Om det er et objekt du ikke er helt sikker på, les referansen eller hjelpefilen til objektet.



Gå igjennom de ulike delene av patchen og vær sikker på at du skjønner hvordan alle delene fungerer. Når alt er klart, gå videre til oppgaven.

Oppgave 50

50.1 Lag en patch som tilsvarer den øvelsen du nettopp har gjennomgått, med følgende forandringer: «x»-tasten skal skru audio av og på, «y»-tasten skal spille inn 1 sekund lyd, og «z»-tasten skal spille av lyden.

50.2 Avspilling skal skje ved en fjerdedels hastighet (altså vare 4 sekunder).

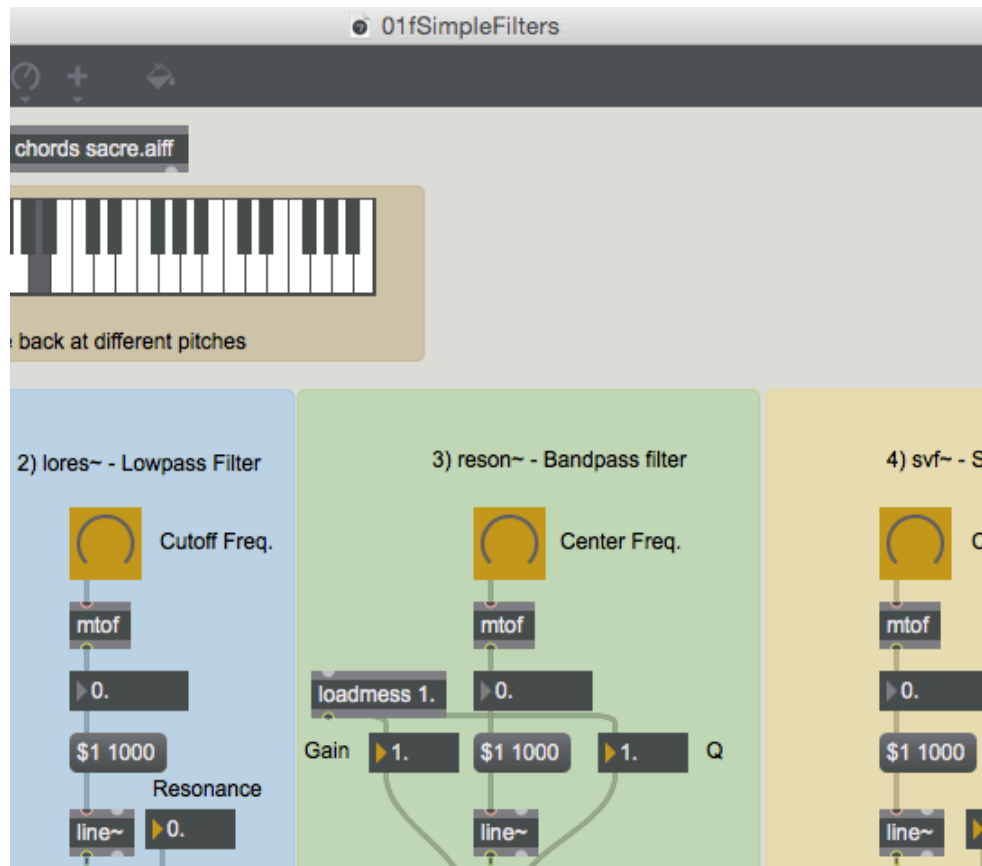
50.3 Det skal være en omhyllingskurve for amplituden.

50.4 Lyden skal også ha en 12 Hz vibrato som går opp og ned 1 halvtone.

Kontrolloppgave til øvelsen *Simple Filters*

I denne øvelsen skal vi se på enkel bruk av filtre i MSP, samt få grunnleggende forståelse av ulike filtertyper.

Gå gjennom øvelsen *Simple Filters* og den tilhørende øvelsespatchen. Om det er noen punkter du er usikker på, gå tilbake og repeter det som er uklart. Om det er et objekt du ikke er helt sikker på, les referansen eller hjelpefilen til objektet.



Gå igjennom de ulike delene av patchen og vær sikker på at du skjønner hvordan alle delene fungerer. Når alt er klart, gå videre til oppgaven.

Oppgave 51

Start med en lydfilavspilling som går i løkke.

51.1 Spill lyden gjennom et resonant filter der filterfrekvensen oscillerer mellom 50 Hz og 10 kHz 2 ganger i sekundet. Filtertypene lavpass, høypass, båndpass og båndstopp (notch) skal kunne velges fra en meny.

51.2 I dette eksempelet skal vi lage en såkalt filterbank. Spill lyden gjennom 6 parallelle resonante båndpassfilter med frekvensene 100, 200, 300, 400, 500 og 600 Hz. Filtrene skal være så resonante at de har en lang etterklang.

Nye objekter introdusert i denne øvelsen:

lores~

lores~

reson~

reson~

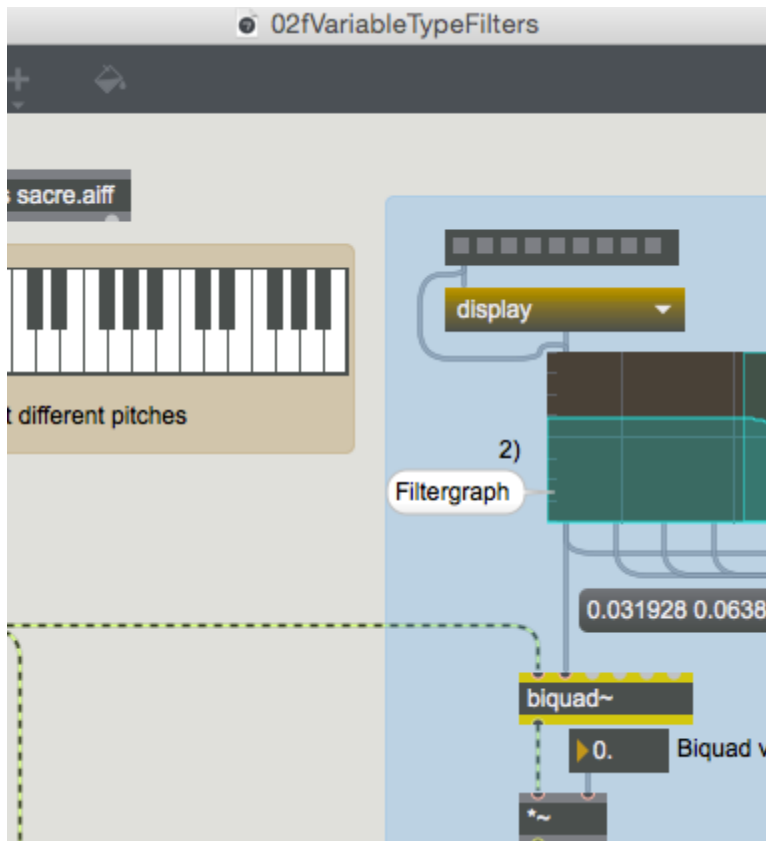
svf~

svf~

Kontrolloppgave til øvelsen *Variable-type Filters*

I denne øvelsen skal vi se på mer komplekse filtertyper. Vi skal også se på hvordan vi kan lage grensesnitt til filteret.

Gå gjennom øvelsen *Variable-type filters* og den tilhørende øvelsespatchen. Om det er noen punkter du er usikker på, gå tilbake og repeter det som er uklart. Om det er et objekt du ikke er helt sikker på, les referansen eller hjelpefilen til objektet.



Gå igjennom de ulike delene av patchen og vær sikker på at du skjønner hvordan alle delene fungerer. Når alt er klart, gå videre til oppgaven.

Oppgave 52

52.1 Start med en lydfilavspilling som går i løkke. I dette eksempelet skal vi lage den samme filterbanken som i øvelsen *Simple Filters*, men denne gangen skal vi bruke *biquad~*-objektet i stedet. Filterfrekvensene til de 6 parallelle resonante båndpassfilter er 100, 200, 300, 400, 500 og 600 Hz denne gangen også. Filtrene skal være så resonante at de har en lang etterklang. For å enkelt sette filterkoeffisientene til *biquad~*-objektet, bruk *filtercoeff~*-objektet.

52.2 Start med en lydfilavspilling som går i løkke. Lag en samling som har de samme egenskapene som tradisjonelle miksebord equalizere. Det vil si muligheten til å kunne stille på bass, mellomtone og diskant. Dette skal løses med å bruke *cascade~*-objektet.

Nye objekter introdusert i denne øvelsen:

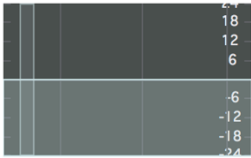
biquad~

filtergraph~

cascade~

filtercoeff~

biquad~



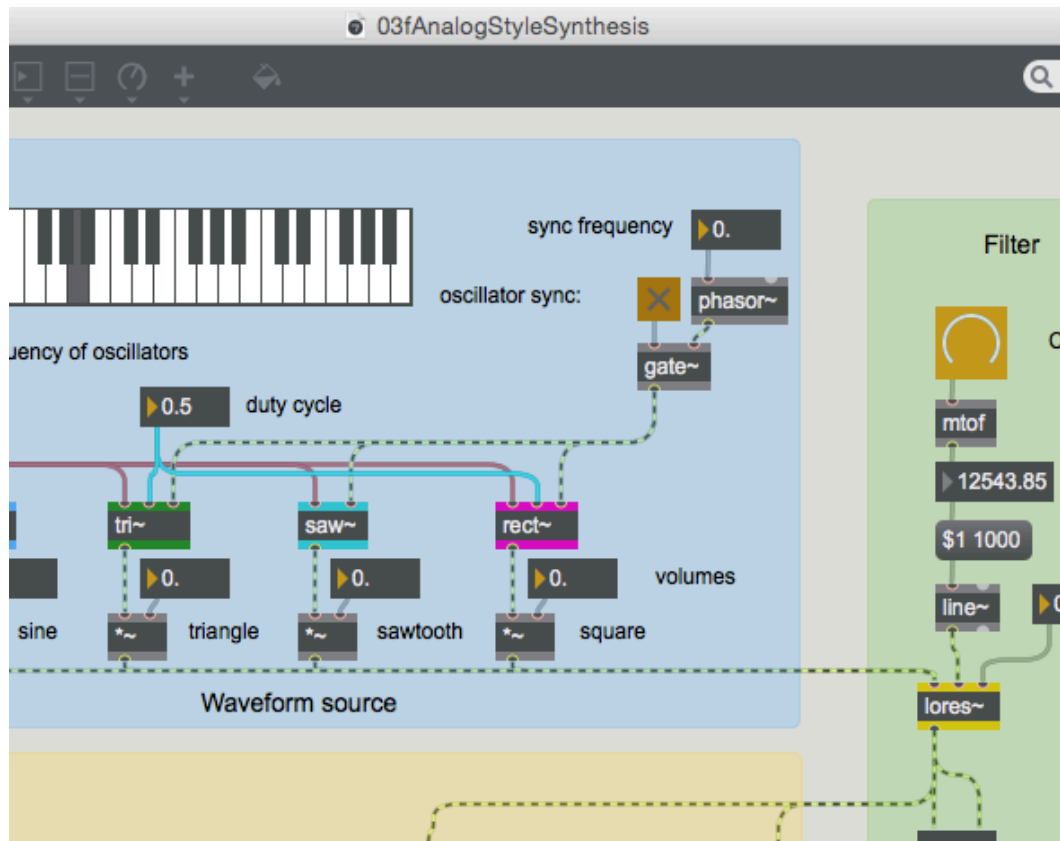
cascade~

filtercoeff~

Kontrolloppgave til øvelsen *Analog-style Synthesis*

I denne øvelsen skal vi jobbe med digital simulering av analog syntese. Vi skal blant annet se på utfordringene med ulike typer digitale oscillatorer.

Gå gjennom øvelsen *Analog-style synthesis* og den tilhørende øvelsespatchen. Om det er noen punkter du er usikker på, gå tilbake og repeter det som er uklart. Om det er et objekt du ikke er helt sikker på, les referansen eller hjelpefilen til objektet.



Gå igjennom de ulike delene av patchen og vær sikker på at du skjønner hvordan alle delene fungerer. Når alt er klart, gå videre til oppgaven.

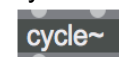
Oppgave 53

53.1 Miks sammen to oscillatorer som begge har en frekvens på 50 Hz. En med en trekantbølge og en med en firkantbølge. La en oscillator med en frekvens på 0.125 Hz styre duty cyclen på oscillatorene slik at den oscillerer mellom 0 og 1.

53.2 Spill lyden gjennom et resonant lavpassfilter der filterfrekvensen oscillerer mellom 0 Hz og 5 kHz med en frekvens på 0.0625 Hz.

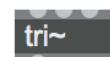
Nye objekter introdusert i denne øvelsen:

cycle~



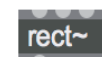
scope~

tri~

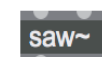


spectroscope~

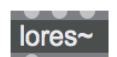
rect~



saw~



lores~





BEAP — mer virtuell analog syntese

La oss fortsette med å utvide synthesizeren fra forrige BEAP-øvelse. Legg til en oscillator til. På denne oscillatoren kobler du CV ut fra sequenceren til CV2 inn i stedet for CV1 slik vi gjorde sist. I kontrollvinduet for CV2 setter vi ikke verdien til 100% men heller 99.5%:

CV2 99.5 %

På denne måten får vi en liten forskjell i stemmingen av oscillatorene, noe som ligner på det som finnes i gamle analoge synthesizere.

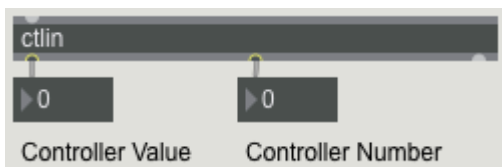


Noe av det fine med modulære synthesizere er de fleksible modulasjonsmulighetene. La oss prøve ut noen av disse. Koble utgangen fra oscillator 1 inn til CV2 i *Ladder*-patchen, og skru gradvis opp verdien for CV2. Skru også verdien for frekvensen du har i utgangspunktet («Freq») litt ned. Koble utgangen fra oscillator 2 inn til *Res (0-5v)*-inngangen i *Ladder*-patchen, og skru gradvis opp CV verdien for *Res (0-5v)* til 50 prosent. Se skjermbildet nederst på denne siden om du er usikker på signalgangen.

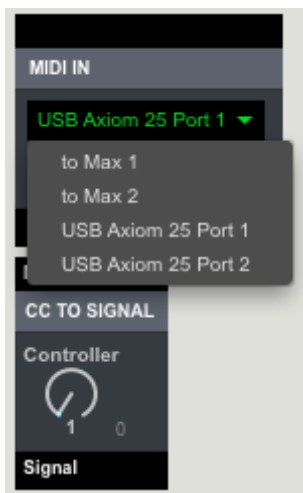
Oscillator 1 modulerer nå filterfrekvensen og oscillator 2 modulerer filterresonansen noe som gir en mer levende klang.

Til slutt skal vi koble til en MIDI-kontroller. For å finne ut hvilket kontrollernummer MIDI-kontrolleren din sender ut, velg maxobjektet «ctlin» og koble utgang nr 1 («Controller Value») fra «ctlin» til et «number»-objekt. Skru på en knapp på MIDI kontrolleren din, for eksempel modulasjonshjulet. «number»-objektet vil nå vise om du mottar data fra MIDI kontrolleren din.

Koble utgang nr 2 («Controller Number») fra «ctlin» til et annet «number»-objekt. Skru på en knapp på MIDI kontrolleren din igjen. «number»-objektet vil nå vise hvilket kontrollernummer denne sender på. Dette nummeret må du velge i *CC to Signal*. Mer informasjon om «ctlin»-objektet finnes i øvelsen *Basics — Getting MIDI input and output*.

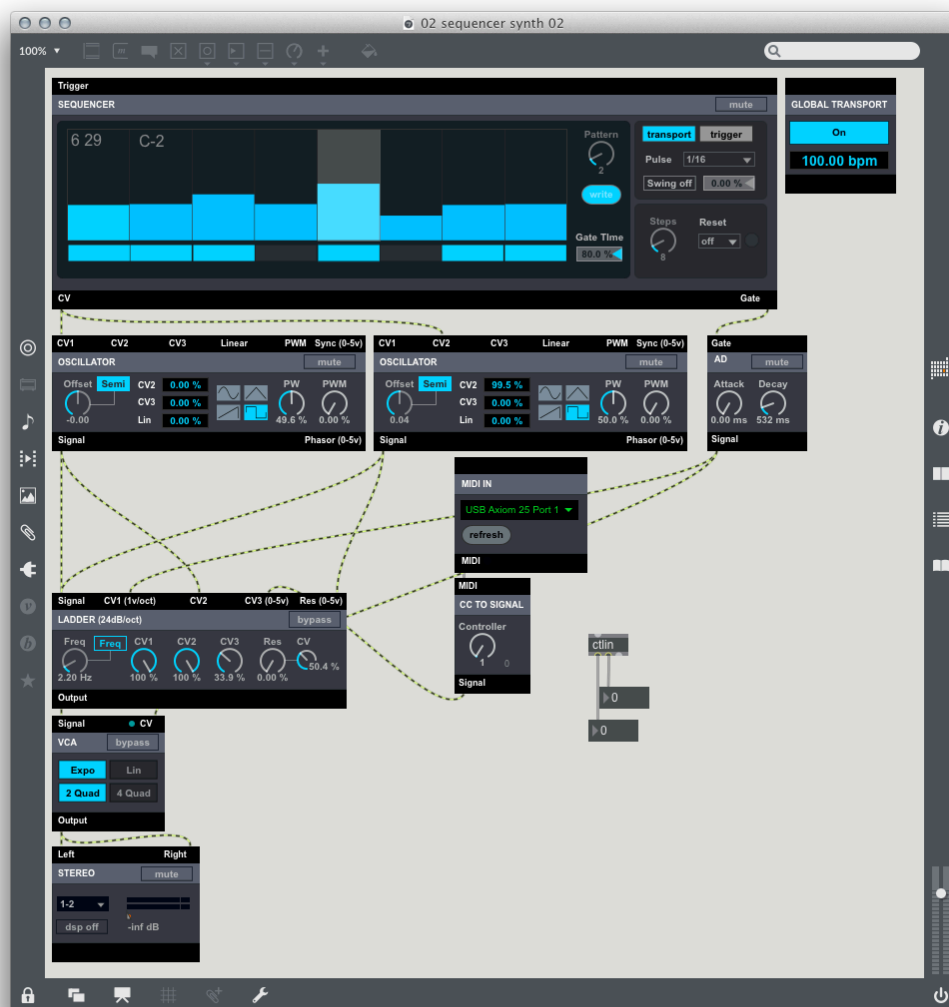


Velg «Midi in»-patchen fra BEAP MIDI-kolonnen. Velg deretter «CC to Signal»-patchen fra den samme MIDI-kolonnen. «Midi in»-patchen tar inn MIDI signaler og «CC to Signal»-patchen gjør MIDI kontrollerdata om til signaler som vi kan bruke i BEAP. Koble *MIDI*-utgangen fra «Midi in»-patchen til *MIDI*-inngangen i «CC to Signal»-patchen. Velg navnet på MIDI-kontrolleren din fra menyen i «Midi in»-patchen, og velg riktig kontrollernummer i «CC to Signal»-patchen.



Til slutt kobler du *Signal* ut fra «CC to Signal»-patchen til CV3 (0-5v)-inngangen i *Ladder*-patchen, og skru opp verdien for CV3 til omtrent 35%. Når du nå skruv på MIDI-kontrolleren din kan du forandre filterfrekvensen.

Husk å velge «Autosave Snapshot» for å lagre innstillingene.



Hele patchen