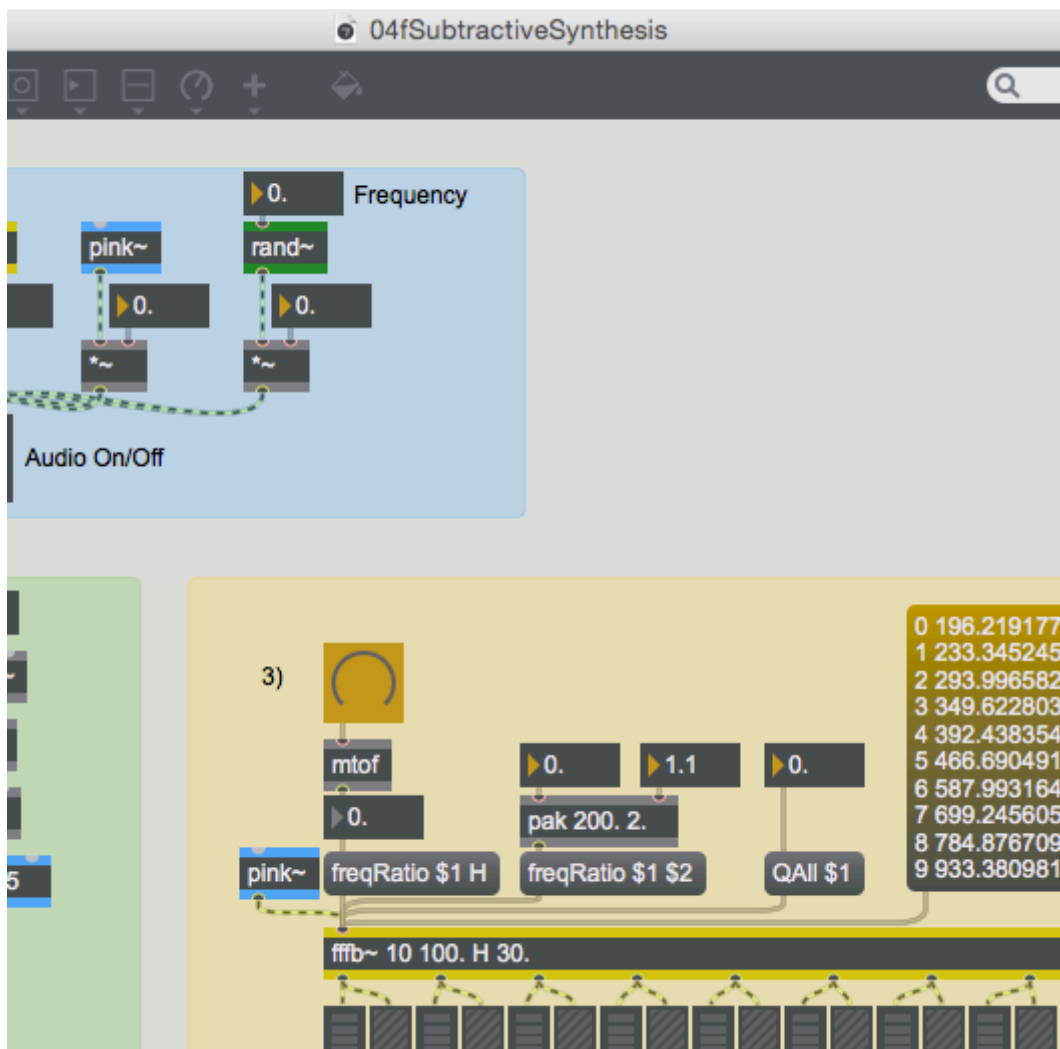


Kontrolloppgave til øvelsen *Subtractive Synthesis*

I denne øvelsen skal vi blant annet se på kombinasjonen av støykilder og filterbanker.

Gå gjennom øvelsen *Subtractive 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 54

Start med en lydfilavspilling som går i løkke.

Spill lyden gjennom 20 parallelle resonante båndpassfilter der den første frekvensen er 100 Hz og de resterende filterne utgjør en harmonisk serie til 100 Hz. Disse 20 parallelle resonante båndpassfilterne skal løses med ett objekt.

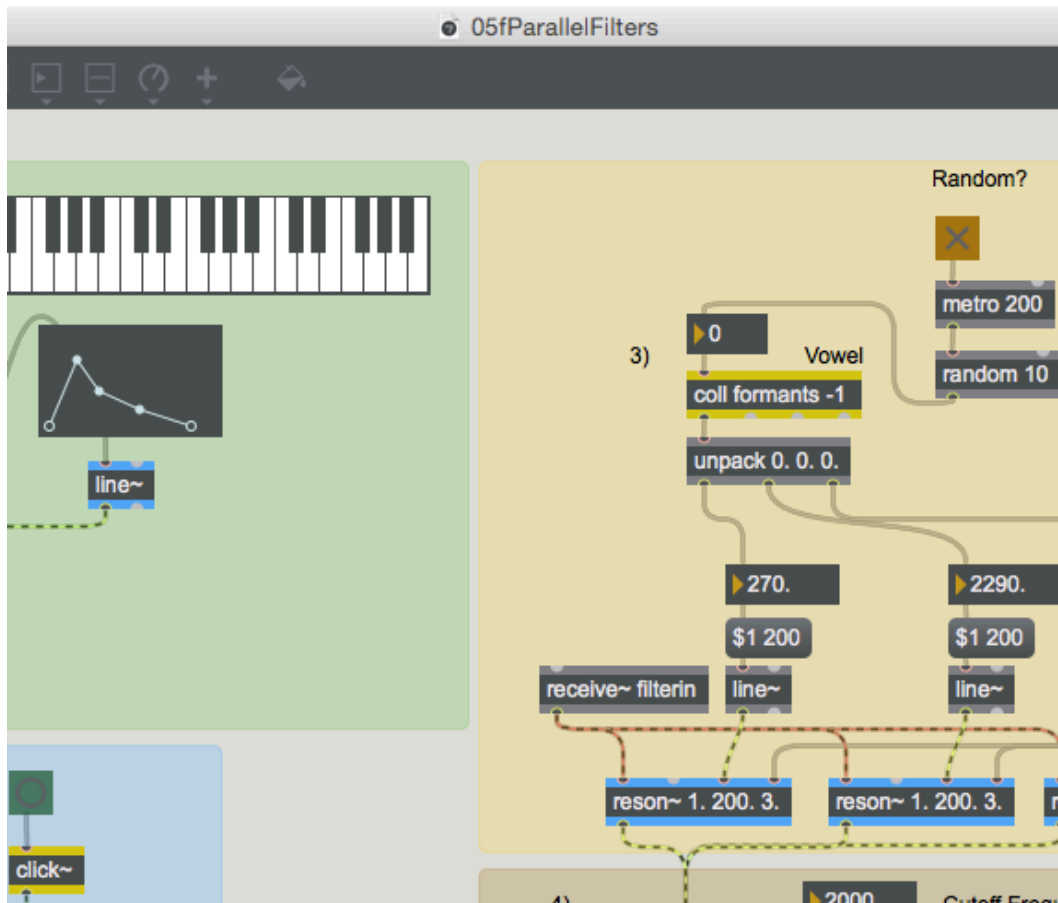
Nye objekter introdusert i denne øvelsen:



Kontrolloppgave til øvelsen *Parallel and Serial Filters*

I denne øvelsen skal vi se på mer komplekse kombinasjoner av ulike filtertyper.

Gå gjennom øvelsen *Parallel and serial 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 55

Ta utgangspunkt i formantfilteret fra øvelsen du nettopp har gjennomgått. Velg ut fire sterke formanter som minner tydelig om menneskestemmen, for eksempel *i*, *e*, *a*, *æ*, og lag en ny liste i *coll*-objektet for disse fire formantene. La nye tilfeldige formanter trigges med et sekunds mellomrom. For hver gang skal det være en formant som ikke ble brukt forrige gang. For å få til dette må du bruke *urn-jb*-objektet.

For å gjøre effekten klarere skal du bruke høy Q-faktor (resonans) i filteret og hvit støy som lydkilde.

Nye objekter introdusert i denne øvelsen:

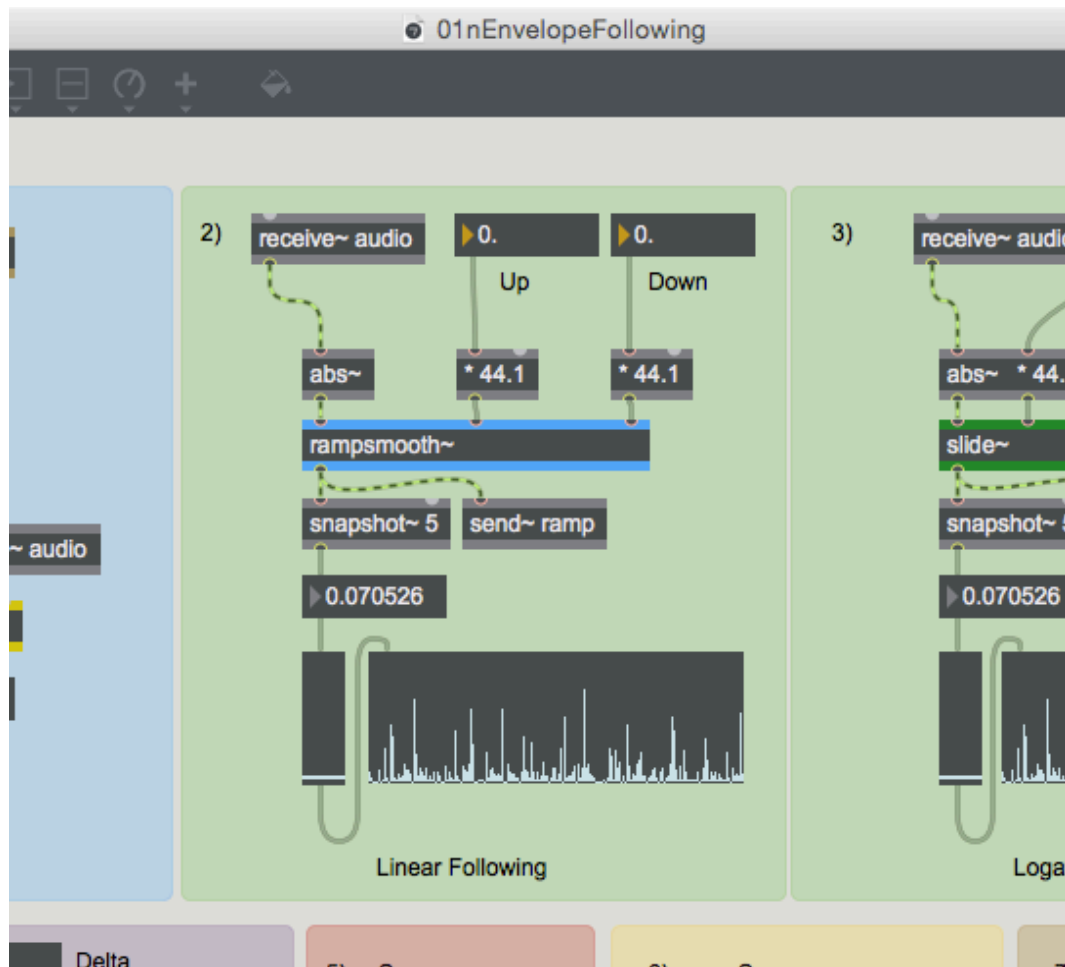
click~



Kontrolløppgave til øvelsen *Envelope Following*

I denne øvelsen skal vi se på måter å jobbe med dynamikk i lydsignaler, og gå nærmere inn på teknikken omhyllingskurvefølging (*Envelope Following*).

Gå gjennom øvelsen *Envelope Following* 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 56

Start med en lydfilavspilling som går i løkke.

Gjør en lineær omhyllingskurvefølging (*Envelope Following*) av denne lyden med en «ramp up»-tid på 1 ms og en «ramp down»-tid på 1 ms. La denne omhyllingskurven (envelope) styre frekvensen til en triangelosillator. Omhyllingskurven skal ha et utslag på 0 Hz til 10 kHz.

Nye objekter introdusert i denne øvelsen:

abs~

abs~

snapshot~

snapshot~

rampsmooth~

rampsmooth~

slide~

slide~

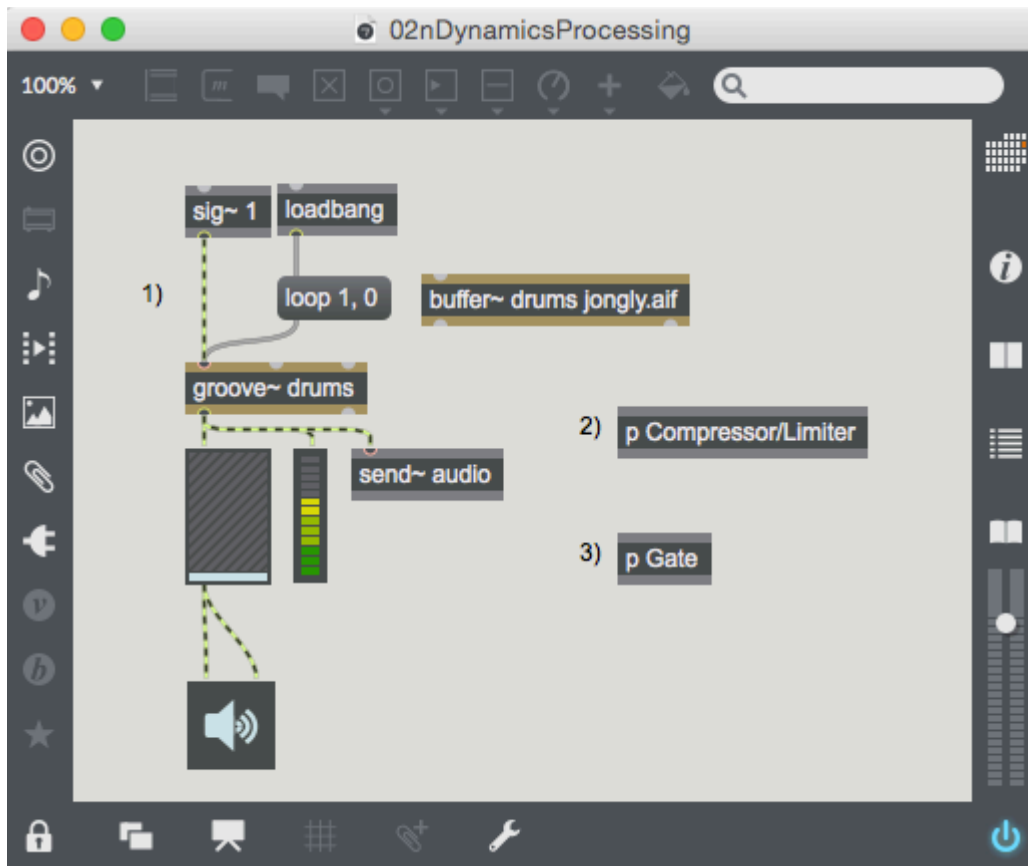
deltaclip~

deltaclip~

Kontrolloppgave til øvelsen *Dynamics Processing*

I denne øvelsen skal vi lage de mest vanlige dynamiske prosessorene; kompressor, limiter og gate.

Gå gjennom øvelsen *Dynamics Processing* 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 57

Trigging av en kompressor med en basstromme er en mye brukt mikseteknikk i ulike typer musikk. Denne teknikken kalles side-kjede komprimering, eller side-chain compression. I denne øvelsen skal vi bruke denne teknikken til å la et signal styre komprimeringen og gatingen av et annet signal.

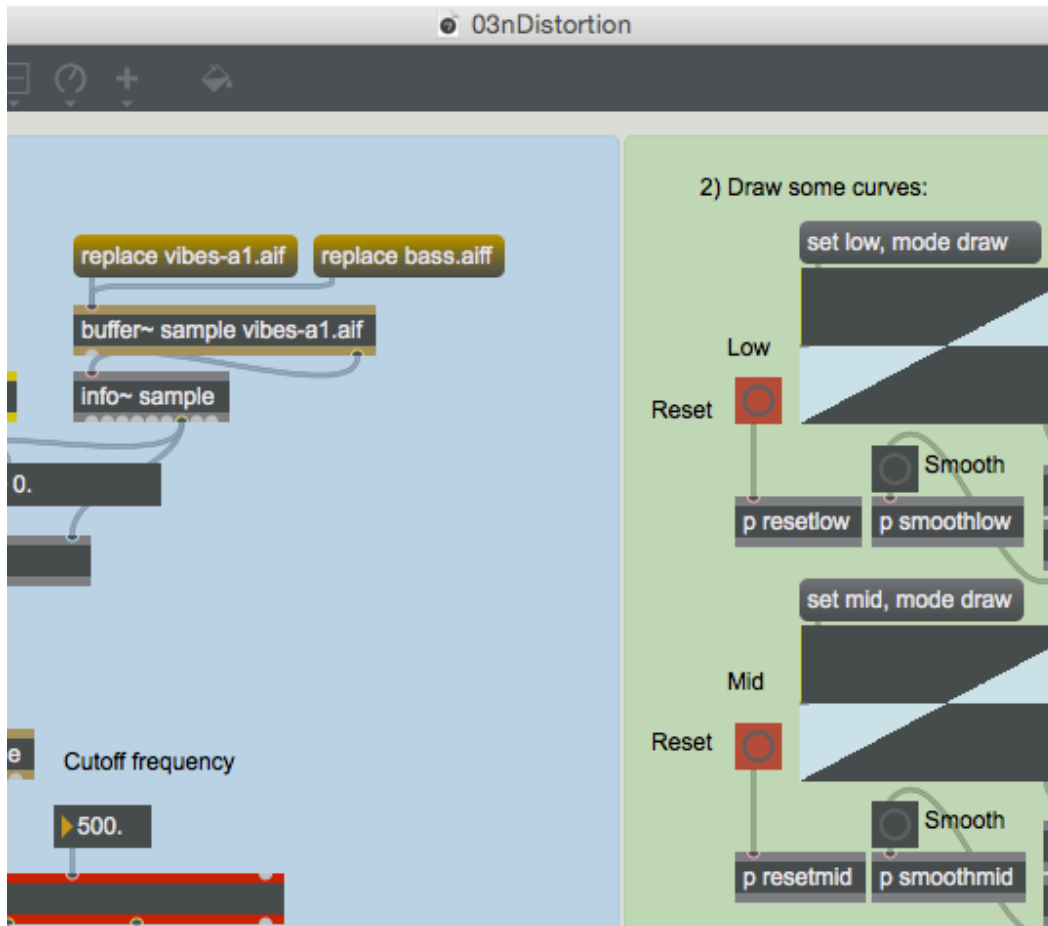
57.1 Ta utgangspunkt i øvelsen du nettopp har gjennomgått. Still inn kompressoren slik at den komprimerer hardt, og still inn Noise Gaten slik at den klipper ganske kontant. Lagre disse innstillingene i *preset*-objekter som aktiveres ved åpning av patchen.

57.2 Bruk trommelooopen som fulgte med øvelsen til å trigge kompressoren og gaten, og la komprimeringen og gatingen av lydfilen «rainstick.aif» som følger med Max bli styrt av trommelooopen.

Kontrolloppgave til øvelsen *Distortion*

I denne øvelsen skal vi se på forvrengning slik den forekommer i blant annet forsterkere.

Gå gjennom øvelsen *Distortion* 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 58

Start med en lydfilavspilling som går i løkke.

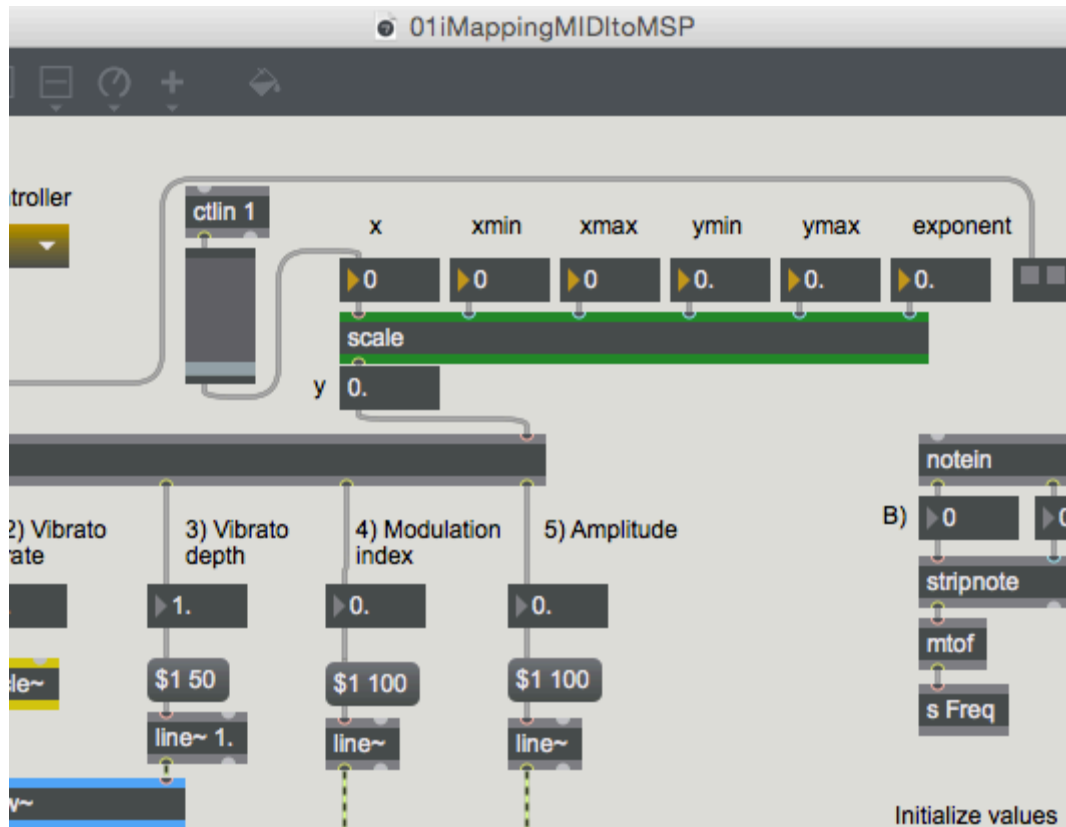
I øvelsen beskrives tre forskjellige overføringsfunksjoner som gir effekter beslektet med rørforsterkere, gitarpedaler og digital distortion. Implementer disse tre. Lagre disse overføringsfunksjonene ved hjelp av en *write*-beskjed til *buffer~*-objektet.

Bruk fire message-bokser koblet til *buffer~*-objektet, med en *replace*-beskjed kombinert med filnavn, for å kunne bytte mellom ingen forvrengning og disse tre forvrengingskarakteristikkene.

Kontrolloppgave til øvelsen *Mapping MIDI to MSP*

Denne øvelsen ser på hvordan man mapper kontrollerdata for MIDI (Roads s. 969) til ulike parametere. Vi skal også se på ulike problemstillinger mellom lineære og ikke-lineære verdier som dukker opp når kontrollerdata blir mappet.

Gå gjennom øvelsen *Mapping MIDI to MSP* 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 59

Ta utgangspunkt i øvelsen du nettopp gjennomgikk.

59.1 Bytt ut verdiene som styres med MIDI kontrollerdata med følgende verdier:

Frekvensomfang: En oktav

Vibratohastighet: 0 100 Hz

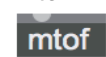
Modulasjonsindeks: 2 til 8

Amplitude: -20 til 0 dB. For å oppgi lydstyrke i dB må du bruke *dbtoa*-objektet.

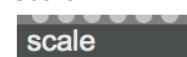
59.2 I tillegg skal du legge til et ekstra menyvalg. Dette siste menyvalget skal være panorering, og panoreringen skal gå fra venstre (minimum) til høyre (maksimum). Dette får du til med å la amplituden i venstre kanal gå fra 1 til 0 samtidig som amplituden i høyre kanal går fra 0 til 1. Om dette blir vanskelig kan du evt. se på *MSP Panning Tutorial 1: Simple panning*.

Nye objekter introdusert i denne øvelsen:

mtof



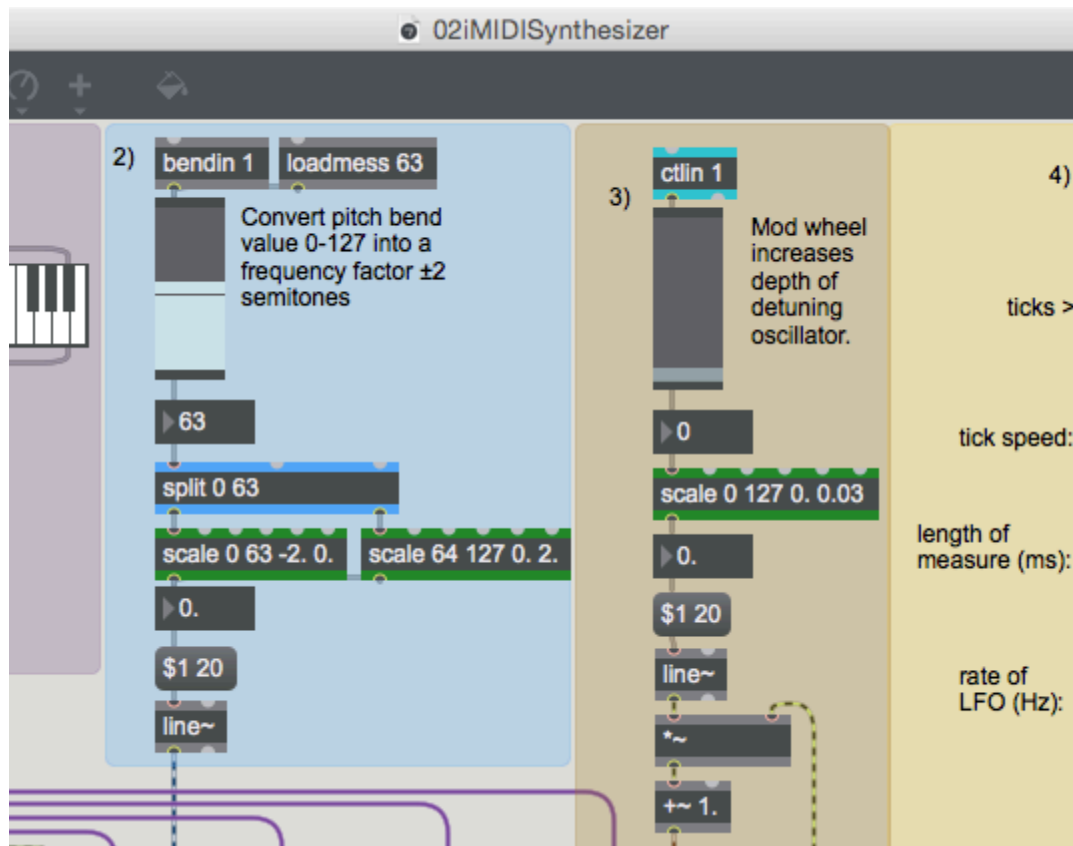
scale



Kontrolloppgave til øvelsen *MIDI Synthesizer*

Denne øvelsen ser på hvordan man bruker de mest vanlige kontrollerne i MIDI-formatet (Roads s. 969) til å kontrollere de mest vanlige parameterne i en synthesizer.

Gå gjennom øvelsen *MIDI Synthesizer* 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 60

60.1 Lag en polyfon synthesizer som i øvelsen, men doble polyfonien fra 4 til 8 toner.

60.2 La detuningen av oscillatoren bli styrt av modulasjonshjulet på MIDI-keyboardet, akkurat slik som i øvelsespatchen. Gi detuningen som blir styrt av modulasjonshjulet dobbelt så stort omfang.

Nye objekter introdusert i denne øvelsen:

mtof

mtof

poly

poly

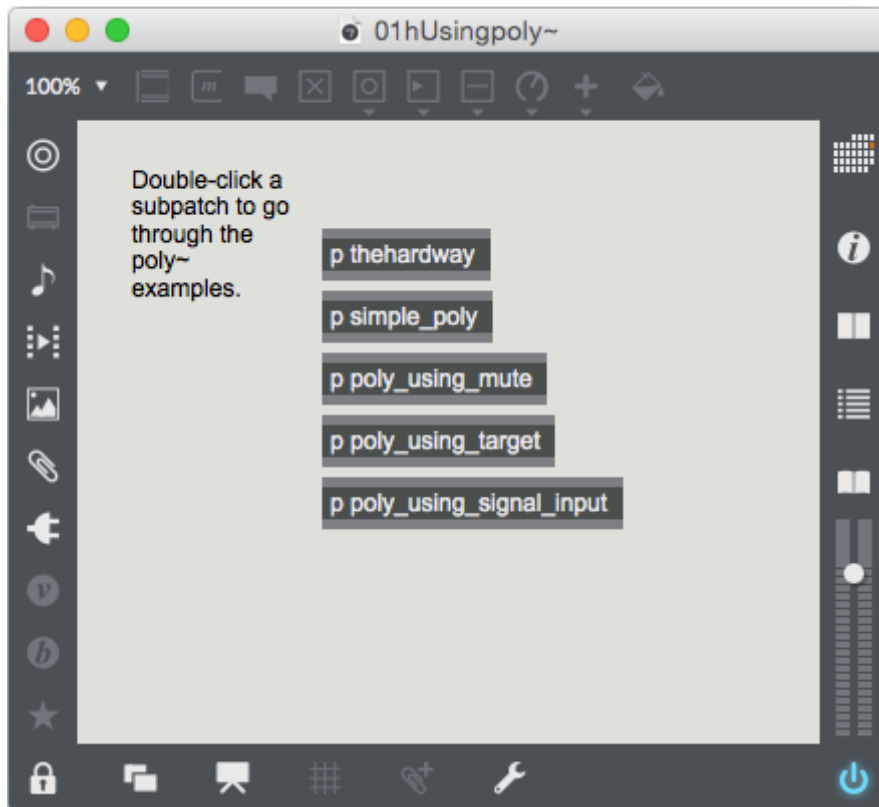
adsr~

adsr~

Kontrolloppgave til øvelsen *Creating Polyphonic Abstractions*

Vi hopper over øvelsen *MIDI Sampler - Creating simple MIDI-compatible samplers in MSP*, og går rett til øvelsen *Creating Polyphonic Abstractions*. Denne øvelsen ser på mer effektive måter å behandle polyfoni på enn de vi har benyttet tidligere.

Gå gjennom øvelsen *Creating Polyphonic Abstractions* 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 61

I øvelsens første eksempel er det implementert en liten synthesizer kalt *littlesynth~*. Denne finner du i subpatchen som heter «*thehardway*».

Lag en tilsvarende synthesizer, men gjør følgende forandringer:

I stedet for en sinusbølge skal den bruke en sagtann (hint: se på objektet *saw~*).

littlesynth~ har en omhyllingskurve (envelope) som bruker 30 ms på både å tone inn og ut. Forandre denne verdien til 500 ms.

Synthen skal kunne spilles på fra et MIDI-klaviatur akkurat som i tutorialen.

Synthen skal lagres som en patch slik at du kan kalle den opp fra *poly~*-objektet. Gi den et passende navn, f.eks. *synth01~* eller noe tilsvarende.

Kall synthen opp fra et *poly~*-objekt med en polyfoni på 5.

Hint: se på *target*-beskjeden til *poly~*-objektet.

Nye objekter introdusert i denne øvelsen:

thispoly~

thispoly~

poly~

poly~

out~

out~

out

out~

in~

in~

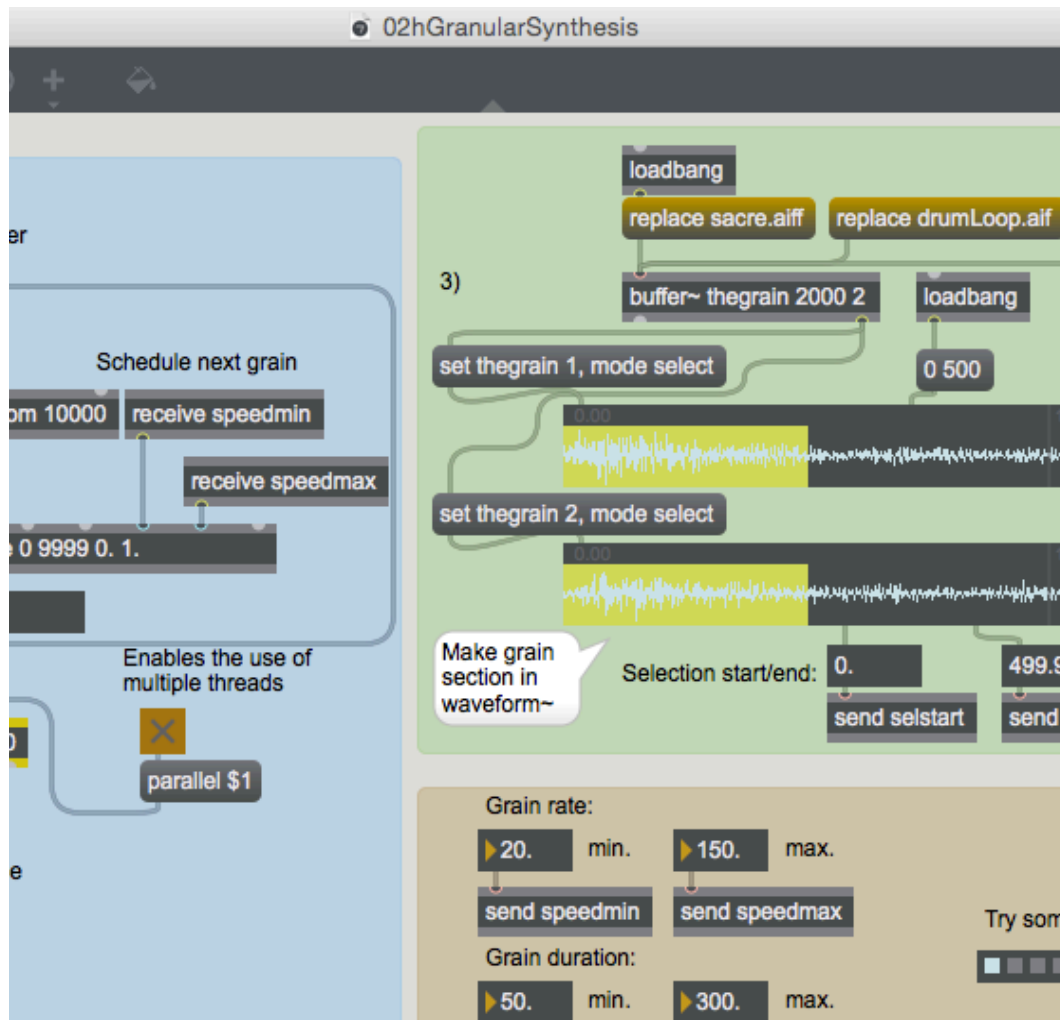
in

in

Kontrolloppgave til øvelsen *Granular Synthesis*

I denne øvelsen skal vi se på kontroll over store mengder lydhendelser, og hvordan vi bruker de i en av de mest brukte teknikkene i eksperimentell lydbruk; granulær syntese.

Gå gjennom øvelsen *Granular 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 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 62

Ta utgangspunkt i granulær syntesen i øvelsen.

62.1 La granulær syntesen ha maksimum 50 korn, såkalte «grains».

62.2 Lag to ulike innstillinger til granulær syntesen. En som består av korte, lyse og ujevne knitrende korn, og en som består av lange, mørke og jevne korn.

Fjern alle delene av patchen som har med innstillinger til granulær syntesen å gjøre, og erstatt disse med to *message*-bokser som sender informasjon ved hjelp av et semikolon. Se *Max Tutorial 16: Remote Messaging*, for mer informasjon om dette.

La disse to *message*-boksene styre hvilke av de to innstillingene man vil lytte til.

Nye objekter introdusert i denne øvelsen:

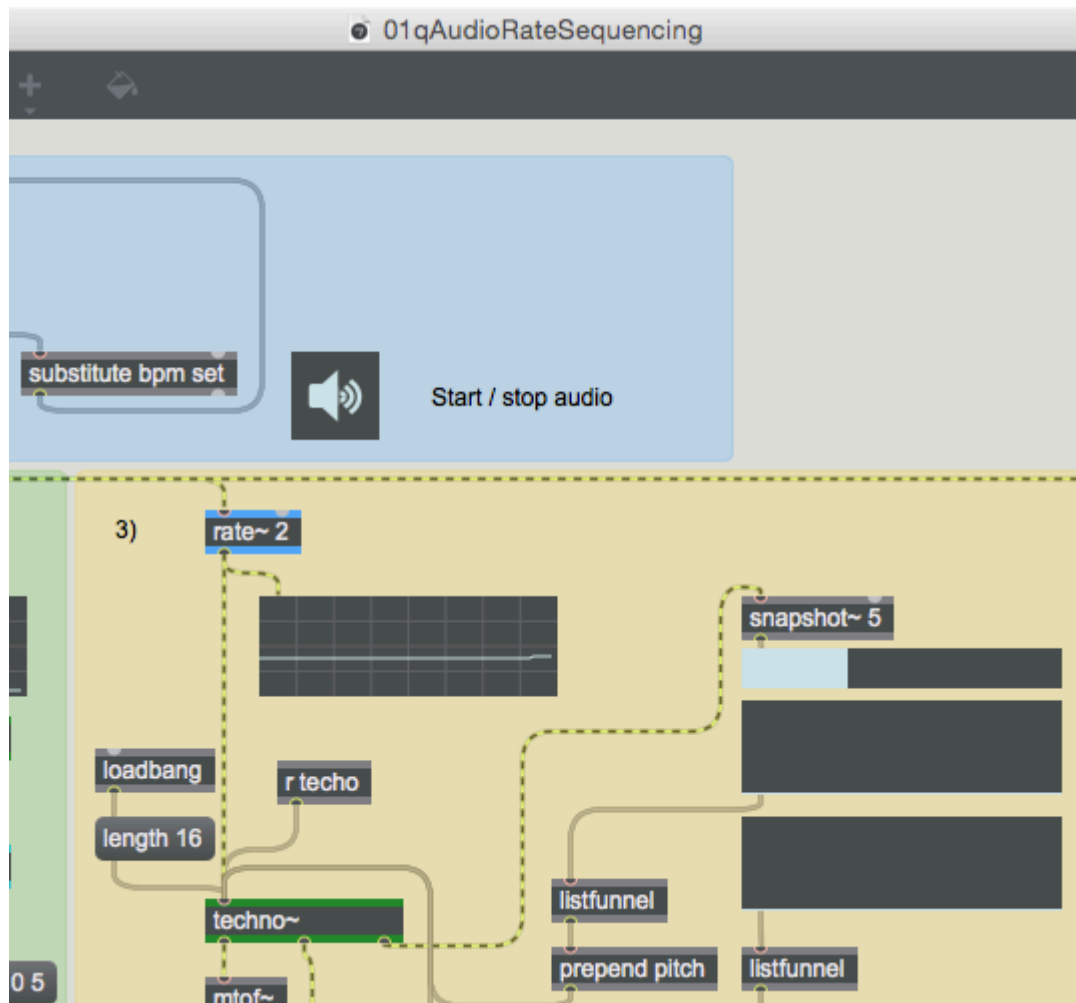
adstatus

adstatus

Kontrolloppgave til øvelsen *Audio-rate Sequencing*

Vi hopper over øvelsen *Audio-rate Control Data*. Selv om det kan være nyttig å bruke MSP-objekter for kontroll av data, har vi ikke bruk for det nå. I stedet vil vi gå rett til øvelsen *Audio-rate Sequencing*. I denne øvelsen skal vi se på hvordan vi kan kontrollere tidsforløpet i en patch med lydobjekter i MSP i stedet for å bruke kontrollobjekter i MAX. Dette gir oss blant annet bedre og mer nøyaktig kontroll over tidsforløpet.

Gå gjennom øvelsen *Audio-rate sequencing* 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 63

63.1 Ta utgangspunkt i audio-rate sequencing med `techno~`-objektet. Lag en pitch- og amplitudesevens og lagre de i en `message`-boks som trigges ved start av patchen. For hvert steg i sekvensen skal decayen bli kortere og kortere.

63.2 Lag en audio-rate sequencer til med *techno~*-objektet, og la denne trigges av samme *sync~*-objekt som den forrige. Denne sekvensen skal ha en egen pitch- og amplitudesekvens lagret i en *message*-boks som trigges ved start av patchen. Denne sekvensen skal ha en lengde på 8 skritt, men med samme tempo som den andre sekvensen.

Nye objekter introdusert i denne øvelsen:

sync~

sync~

rate~

rate~

delta~

delta~

edge~

edge~

techno~

techno~

seq~

seq~

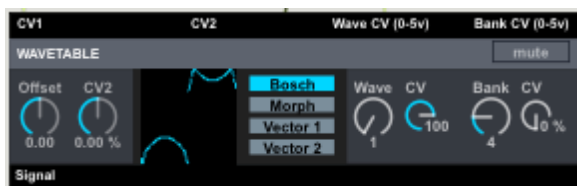
BEAP — bølgetabellsyntese

I denne øvelsen skal vi bytte ut de tradisjonelle oscillatorene med såkalte bølgetabelloscillatorer (wavetable oscillators). Bølgetabellsyntese er en interessant synteseform som blant annet ble populær gjennom PPG Wave synthesizerene på 1980-tallet. I bølgetabellsyntese er bølgeformer lagret digitalt og man velger disse enten manuelt eller automatisk gjennom for eksempel oscillatorer, omhyllingskurver (*envelope*), eller sequencere.

Ta utgangspunkt i den forrige BEAP synthesizer patchen. Ta bort MIDI-kontrolleren. Bytt ut de to oscillatorene med to *Wavetable* oscillatorer. Disse finner du i *Oscillator*-kolonnen.

Koble CV utgangen fra sequenceren til CV1 inngangen på Wavetable oscillator 1 og CV2 inngangen på Wavetable oscillator 2. Still inn CV2 verdien på Wavetable oscillator 2 til 99.5% slik at oscillatorene er litt ulikt stemt.

Skru opp filterfrekvensen for *Ladder* filteret til 129 Hz for å høre mer av klangen i oscillatorene. Du kan velge forskjellige banker med bølgeformer. Velg bank 4 i begge oscillatorene.



Vi kan automatisere valg av bølgeformer. Velg *LFO* BEAP-patchen fra *LFO*-kolonnen. Sett *LFO* frekvensen til 1 Hz.

Velg *Sample and Hold* BEAP-patchen fra *Level*-kolonnen. Koble *LFO Up* utgangen til *S&H Signal* inngangen, *Sequencer Gate* utgangen til *S&H Trigger* inngangen, og *S&H Output* utgangen til *Wave CV (0-5v) input* på både Wavetable oscillator 1 og 2. Skru opp *Wave CV* til 100 på både Wavetable oscillator 1 og 2. Hver gang sequencer BEAP patchen gir fra seg et gate signal vil *Sample and Hold* BEAP-patchen sende ut den verdien den får fra *LFO* BEAP-patchen til *Wave CV (0-5v) input* på både Wavetable oscillator 1 og 2. I praksis betyr dette at oscillatorene får en tilnærmet tilfeldig bølgeform for hver ny note.

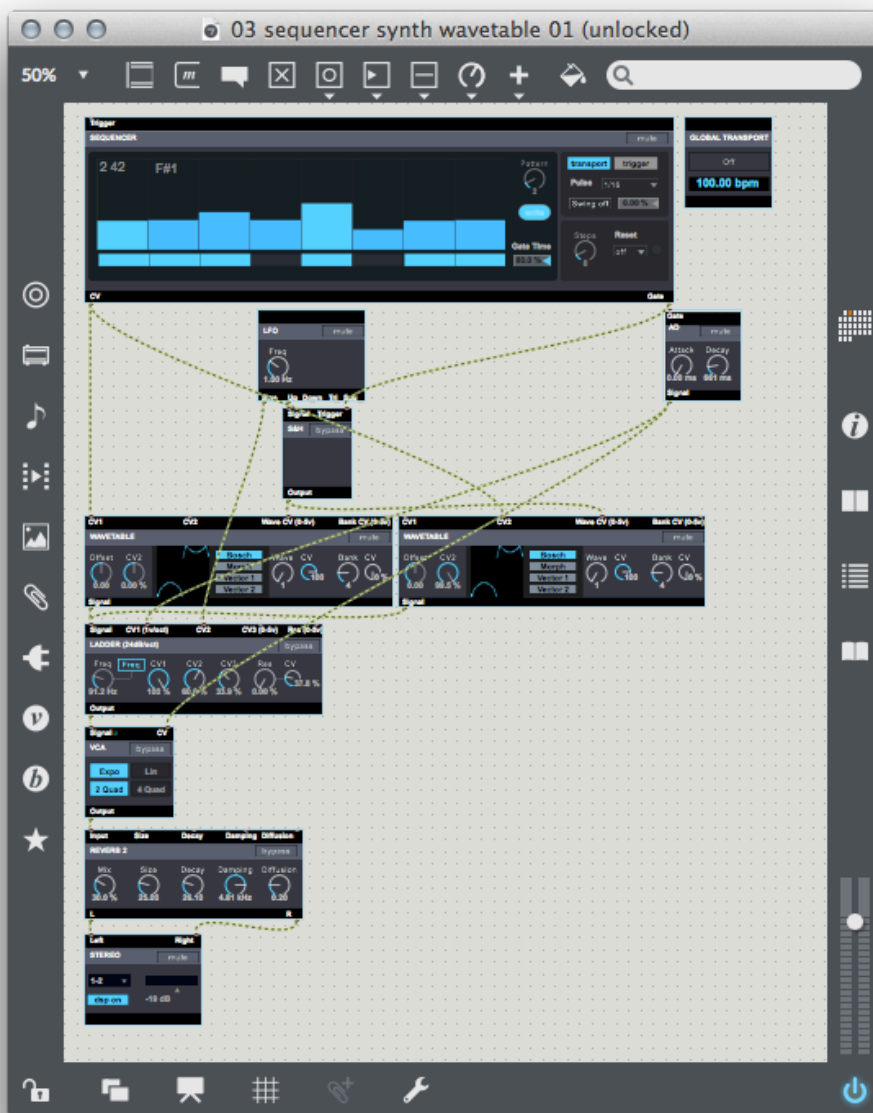


Som en siste modulasjon kan du koble *LFO Sine* utgangen til *Ladder* filter *CV2 input*. Sett filter *CV2* til 60%.

Koble også til en romklang i slutten av signalgangen. Gå til *Effects*-kolonnen og velg *Reverb 2*-patchen. Gi romklangen følgende innstillinger:



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



Hele patchen