Tutoriel SuperCollider

Chapitre 6

Par Céleste Hutchins

2005

[www.celesteh.com](http://www.celesteh.com/)

Licence Creative Commons : paternité uniquement

Synthèse additive

La synthèse additive est l'ajout de tonalités sinusoïdales, généralement dans une série harmonique, à

créer des tons compliqués. Nous avons déjà vu la synthèse additive. Le dernier

exemple du chapitre 2, où nous avons joué les quatre premières harmoniques de 100 Hz était un

exemple de synthèse additive. Tout comme le projet associé au précédent

chapitre.

Certains UGens sont utiles pour ajouter des tonalités sinusoïdales. Vous pouvez mélanger plusieurs signaux en les additionnant tous et en divisant par le nombre de

signaux. Par exemple:

### résultat = (sine1 + sinus2 + sinus3 + sinus4 + sinus5) / 5 ;

Vous pouvez également utiliser un UGen appelé Mix qui prend un ensemble de signaux et,

de manière assez appropriée, les mélange ensemble. C'est comme Pan à l'envers.

### résultat = Mix.ar([sine1, sine2, sine3, sine4, sine5]);

La synthèse additive utilise des tons sinusoïdaux car les tons sinusoïdaux, comme ceux produits par

SinOsc, sont purs. Ils n’ont aucune connotation. Ce sont seulement leur fréquence et

n'avoir aucun autre contenu de pitch. Tout autre ton plus compliqué que vous entendez est

en fait une collection de tons sinusoïdaux. Lorsque vous entendez un ton avec deux harmoniques,

vous entendez une onde sinusoïdale de la fondamentale et les ondes sinusoïdales des deux

des connotations. Les ondes périodiques telles que les triangles, les impulsions et les dents de scie peuvent toutes être

décrit en termes de quels tons sinusoïdaux les composent. Nous y reviendrons

plus tard.

Les sons compliqués du monde réel sont constitués de très nombreuses tonalités sinusoïdales.

Théoriquement, nous pouvons synthétiser n'importe quel son à partir d'une combinaison de composants

tons sinusoïdaux. En pratique, cela peut s'avérer trop compliqué pour certaines tonalités bruyantes.

La synthèse additive est utilisée dans les organes. L'ajout de « arrêts », ou tuyaux, est le

ajout de tons supplémentaires pour créer un timbre complexe. Les orgues électriques, comme le

Hammond B3 utilise également la synthèse additive.

## Modulation en anneau

Sur un synthétiseur analogique, comme ceux que Nicole collectionne, chaque oscillateur

a son propre circuit. Faire de la synthèse additive aurait nécessité une banque de oscillateurs accordables. Cela aurait coûté cher. Les orgues à tuyaux n'ont pas

une façon de varier ce qui sort d'un tuyau. Mais les concepteurs de synthétiseurs ont inventé

façons pour les compositeurs de moduler leurs oscillateurs. Une des premières méthodes de

la modulation est appelée modulation en anneau.

La modulation en anneau doit son nom au fait que lorsque les gens ont commencé à construire

matériel pour ce faire, un tas de composants ont été disposés en cercle,

c'était donc un modulateur "en anneau". Le résultat est un son bruyant. Les modulateurs en anneau fonctionnent

en multipliant le signal porteur par le signal du modulateur. RM est exprimé

arithmétiquement avec multiplication : porteur \* modulateur. Notez que ce sont les

les formes d'onde que nous multiplions et non les fréquences.

### modulateur = SinOsc.ar(modulator\_freq, mul: modulateur\_amp); transporteur = SinOsc.ar(carrier\_freq, mul: carrier\_amp); résultat = modulateur \* porteuse ;

Rappelez­vous cependant que nous avons un argument mul pour SinOsc

### modulateur = SinOsc.ar(modulator\_freq, mul: modulateur\_amp); result = SinOsc.ar(carrier\_freq, mul: modulateur \* carrier\_amp);

Machine Translated by Google

Cela crée exactement le même résultat. Parce que le porteur (c'est­à­dire le résultat) est le

dernier oscillateur d'une chaîne, la variable contrôlant son amplitude est généralement simplement

appelé « ampli ».

Si nous écrivons ceci dans un affichage et utilisons des ControlValues prédéfinies pour le

modulator\_freq et Carrier\_freq, tous deux sont uniquement dans la plage audio.

Nous voulons pouvoir utiliser le modulateur comme oscillateur basse fréquence

(parfois appelé LFO) pour moduler la porteuse avec des ondes trop lentes

entendre. Le fichier d'aide ControlValue mentionne une autre classe appelée ControlSpec.

Selon son propre fichier d'aide, une ControlSpec est une "spécification pour un contrôle

saisir." Une instance de ControlSpec est un objet qui décrit la plage d'un

instance d’une ControlValue. Le spec\_ setter pour ControlValues peut soit prendre un

symbole qui fait référence à un ControlSpec prédéfini, ou il peut prendre un ControlSpec

défini par le programmeur. Le constructeur ControlSpec est décrit comme

ControlSpec.new (minval, maxval, warp, step, default,unities);

Nous voulons une plage proche de 0 à 20,00 Hz. Puisque notre perception du ton est

exponentielle (chaque octave est le double du Hz de l'octave du dessous), elle veut que le

curseur pour augmenter de façon exponentielle, avec une valeur de pas la plus petite possible.

modulateur\_freq.spec\_(ControlSpec(0.1, 20000, \exp, 0, 1));

Notre affichage pour la modulation en anneau est ci­dessous.

(

Display.make({ arg thisDisplay, carrier\_freq, modulateur\_freq,

ampli, modulateur\_amp, panoramique ;

Machine Translated by Google

carrier\_freq.spec\_(\freq); modulateur\_freq.spec\_(ControlSpec(0.1, 20000, \exp, 0, 1)); amp.spec\_(\amp);

modulateur\_amp.spec\_(\amp); pan.spec\_(\pan);

thisDisplay.synthDef\_({ arg carrier\_freq, modulateur\_freq,

ampli, modulateur\_amp, panoramique ;

modulateur var , panner, résultat ;

modulateur = SinOsc.ar(modulator\_freq,

mul:modulateur\_amp); résultat = SinOsc.ar(carrier\_freq,

mul : modulateur \* ampli) ; casseroles = Pan2.ar(résultat, casserole); Out.ar(0, casseroles);

}, [\carrier\_freq, carrier\_freq, \modulator\_freq,

modulateur\_freq, \amp, amp, \modulator\_amp, modulateur\_amp,

\pan Pan]);

thisDisplay.name\_(" Modulation en anneau");

}).montrer;

)

Lorsque vous déplacez les deux fréquences dans la plage audio, vous devriez entendre un timbre compliqué. Il s'agit en fait de deux tons. La modulation en anneau produit une somme et des fréquences différentes. Si l'on règle la porteuse sur 400 Hz et le modulateur sur

500 Hz, les fréquences résultantes seront 100 Hz et 900 Hz. (ROBERT)

« Ces produits sont appelés « bandes latérales ». (ORTON/R) Le résultat est simple et simple tant qu'on s'en tient aux ondes sinusoïdales, mais quand on passe à un mode plus forme d'onde compliquée, une avec des harmoniques, les partiels du plus complexe

les vagues créent également des bandes latérales. (ORTON/R) « [Chaque] partiel d’une entrée sera

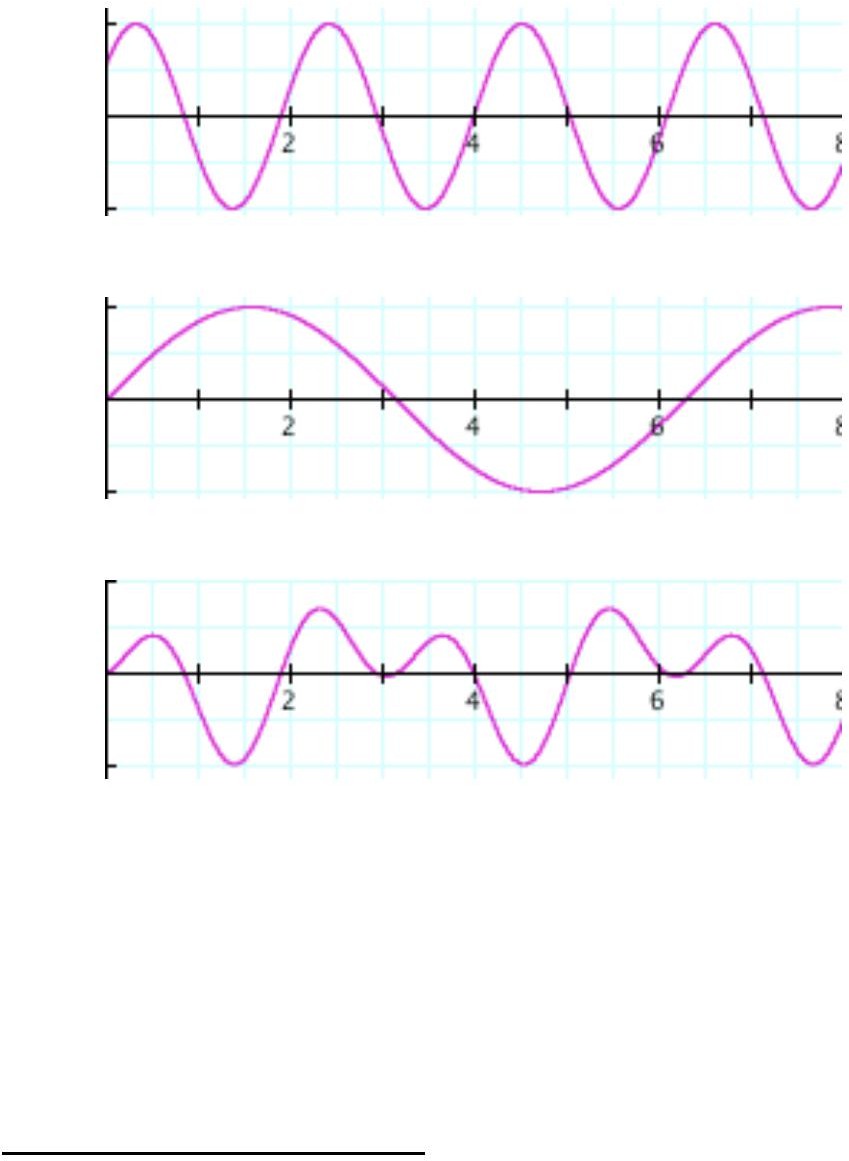
être ajouté et soustrait de chaque partie de l'autre. » (ROBERTS) Ces

les bandes latérales ne sont « pas liées à la série harmonique ». (ORTON/R) Cette création

des partiels enharmoniques est ce qui donne à la modulation en anneau sa réputation bruyante.

En modulation en anneau, la fréquence de modulation va entre 1 et ­1. Ce signifie que lorsque le modulateur et la porteuse sont négatifs, le résultat est positif.

Lorsque ces deux fréquences sont modulées en anneau :

et

Le résultat est

Notez que lorsque les deux fréquences sont négatives, le résultat est positif.

La modulation d'amplitude

La modulation en anneau est très étroitement liée à la modulation d'amplitude. Ils sont

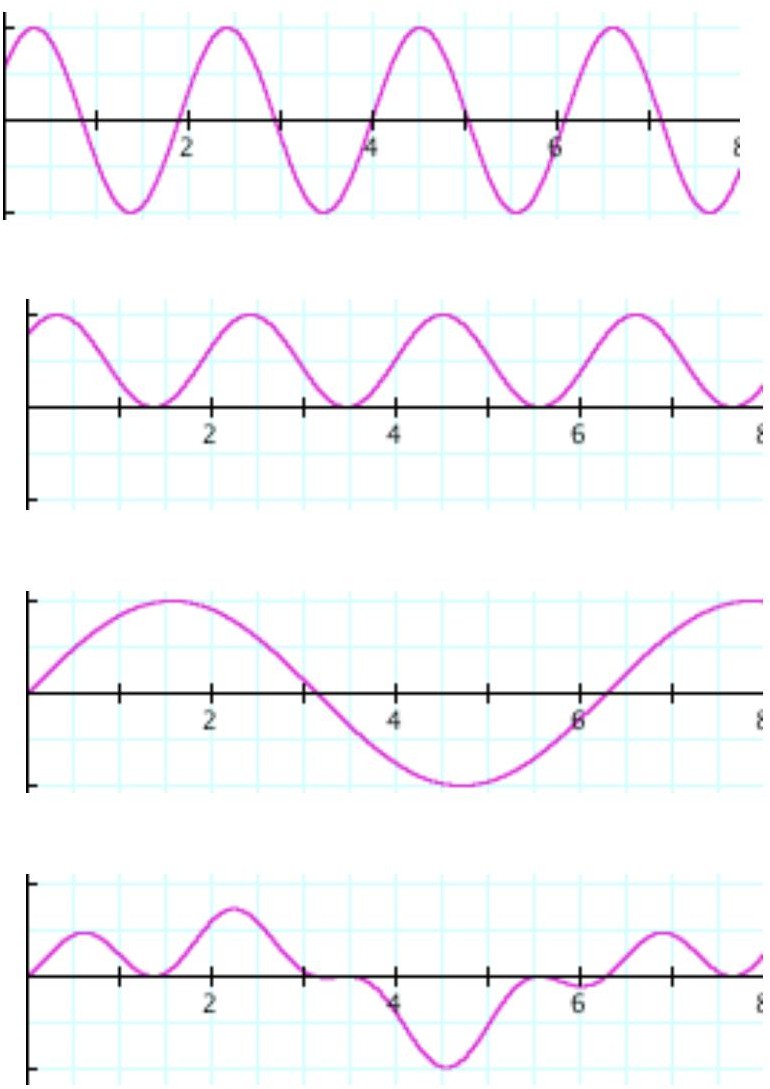
exactement pareil sauf qu'en modulation AM, le modulateur va entre 1 et 0 au lieu d'entre 1 et ­1. Il n'y a jamais de circonstance où deux

les négatifs sont multipliés ensemble.

On divise le signal du modulateur par deux pour diviser par deux son amplitude. Ensuite on ajoute 0,5

pour qu'il soit centré sur 0,5 au lieu de zéro.

Cela transforme

à

quand tu multiplies ça par

vous obtenez

Remarquez en quoi cela diffère du graphique de modulation en anneau de la section précédente.

Mathématiquement, la transformation est :

Machine Translated by Google

### modulateur = (modulateur / 2) + 0,5 ;

Puisque le modulateur est un SinOsc, nous pouvons utiliser le mul et ajouter des arguments.

### modulateur = SinOsc.ar(modulator\_freq, 0, modulator\_amp /2, 0.5);

La dernière mission de Nicole est de créer un écran qui effectue à la fois la modulation en anneau

et modulation d'amplitude. Elle a remarqué que diviser par 2 équivaut à

en multipliant par 0,5. Elle se rend compte que l'utilisateur peut basculer entre RM et AM, si

elle utilise une variable. Elle appelle ça la compensation.

### modulateur = SinOsc.ar(modulator\_freq, 0, modulateur\_amp \* offset,

compenser);

Lorsque le décalage est de 0,5, elle obtient une modulation AM, mais lorsque le décalage passe à zéro,

l'amplitude passe à zéro au lieu de un. Elle a une idée.

### modulateur = SinOsc.ar (modulator\_freq, 0, modulateur\_amp \*

(1 ­ décalage), décalage);

Lorsque le décalage est de 0, l'amplitude est de 1 (fois le modulateur\_amp). Quand le

le décalage est de 0,5, l'amplitude est de moitié. Le décalage ne doit pas nécessairement être simplement de 0 ou 0,5. Il

peut être n’importe quel nombre entre les deux. Elle crée un ControlSpec pour le décalage.

### offset.spec\_(ControlSpec(0, 0.5, \linear, 0, 0.5));

C'est linéaire car vous ne pouvez pas avoir de ControlSpecs exponentiels qui ont zéro valeurs.

L'affichage de Nicole est ci­dessous :

Machine Translated by Google

(

Display.make({ arg thisDisplay, carrier\_freq, modulateur\_freq,

amp, modulateur\_amp, panoramique, décalage ;

carrier\_freq.spec\_(\freq); modulateur\_freq.spec\_(ControlSpec(0.1, 20000, \exp, 0, 1)); amp.spec\_(\amp);

modulateur\_amp.spec\_(\amp); pan.spec\_(\pan);

offset.spec\_(ControlSpec(0, 0.5, \linear, 0, 0.5));

thisDisplay.synthDef\_({ arg carrier\_freq, modulateur\_freq,

amp, modulateur\_amp, panoramique, décalage ;

modulateur var , panner, résultat ;

modulateur = SinOsc.ar (modulator\_freq, 0, modulateur\_amp \* (1 ­ décalage), décalage);

résultat = SinOsc.ar(carrier\_freq,

mul : modulateur \* ampli) ; casseroles = Pan2.ar(résultat, casserole); Out.ar(0, casseroles);

}, [\carrier\_freq, carrier\_freq, \modulator\_freq,

modulateur\_freq, \amp, amp, \modulator\_amp, modulateur\_amp,

\pan, pan, \offset, offset]);

thisDisplay.name\_("Amplitude / Modulation en anneau");

}).montrer;

)

Machine Translated by Google

Lorsque le décalage est de 0,5, vous entendez une modulation d'amplitude. Quand c'est un 0,

vous entendez une modulation en anneau. En le déplaçant, vous pouvez entendre des sons

entre les deux. Vous devriez remarquer un troisième ton émergeant comme décalage

approche 0,5. La modulation d'amplitude produit les deux mêmes bandes latérales que

modulation en anneau, mais en AM, la fréquence porteuse est également audible. Quand le

le décalage est à 0,5, l'amplitude des bandes latérales est la moitié de l'amplitude de la porteuse

ou moins. (PENDRE)

## Modulation FM et phase

La FM est une idée utilisée à la radio et existe depuis longtemps. Musical

les applications sont un peu plus récentes. Les synthétiseurs analogiques de Nicole peuvent faire de la FM pour

créer des formes d'ondes complexes. Utiliser la FM numérique pour modéliser le physique

Cependant, ces instruments datent des années 1970. « Modulation de fréquence (ou FM)

la synthèse a été découverte et introduite par John Chowning à Stanford vers

1973. » Cette technologie a été concédée sous licence à Yamaha et a servi de base au développement sauvage

le populaire synthétiseur Yamaha DX7 et la carte SoundBlaster dans les PC d'antan.

(« Synthèse FM sur SND »)

En synthèse FM, le signal modulateur module la fréquence porteuse. C'est,

le pitch de la porteuse est modifié en fonction de la forme d'onde du modulateur.

C'est comme le vibrato, où un joueur déplace sa hauteur de haut en bas, la modulant

sur une période régulière. Nous pouvons utiliser les propriétés du modulateur comme unique

moyen de contrôler le transporteur.

modulateur = SinOsc.ar(modulator\_freq, mul:modulator\_amp,

ajouter : modulateur\_add); carrier = SinOsc.ar(modulateur, mul: carrier\_amp);

Machine Translated by Google

### Si vous considérez FM comme un vibrato, modulator\_add est la fréquence centrale autour lequel nous avons un vibrato. modulator\_amp contrôle la profondeur du vibrato

et modulator\_freq contrôle la vitesse du vibrato.

(

Display.make({ arg thisDisplay, modulateur\_freq, modulateur\_amp,

modulateur\_add, ampli, panoramique ;

modulateur\_freq.sp(1, 0, 20000);

modulateur\_amp.sp(1, 0, 20000);

modulateur\_add.sp(0, 0, 20000); amp.spec\_(\amp); pan.spec\_(\pan);

thisDisplay.synthDef\_({ arg modulateur\_freq, amp,

modulateur\_amp, modulateur\_add, panoramique = 0 ;

modulateur var , porteur, panneur ;

modulateur = SinOsc.ar(modulator\_freq, mul:modulator\_amp,

ajouter : modulateur\_add); carrier = SinOsc.ar(modulateur, mul: amp);

casseroles = Pan2.ar(support, casserole); Out.ar(0, casseroles);

}, [\modulator\_freq, modulateur\_freq, \amp, amp,

\modulator\_amp, modulateur\_amp, \modulator\_add, modulateur\_add, \pan, pan]);

thisDisplay.name\_(" Modulation de fréquence");

}).montrer;

Machine Translated by Google

)

Notez que, lorsque le modulator\_freq est dans la plage audio, la modification du modulator\_amp modifie le nombre de bandes latérales. L'effet du modulator\_amp, cependant, varie à mesure que vous déplacez le modulator\_freq.

La modulation de phase est un type de FM. Le SinOsc UGen prend un argument de phase. Nous transmettons le signal du modulateur à l'argument de phase de la porteuse.

modulateur = SinOsc.ar(modulator\_freq, mul: modulateur\_amp); carrier = SinOsc.ar(carrier\_freq, modulateur, amp);

Notez qu'avec la modulation de phase, le modulator\_freq ne modifie pas la

nombre de bandes latérales, uniquement leur fréquence. Le modulateur\_amp est quoi modifie le nombre de bandes latérales.

(

Display.make({arg thisDisplay, carrier\_freq, modulateur\_freq,

ampli, modulateur\_amp, panoramique ;

carrier\_freq.spec\_(\freq);

modulator\_freq.sp(1, 1, 20000, warp:'exponentiel'); amp.spec\_(\amp);

modulateur\_amp.spec(\amp, 1); pan.spec\_(\pan);

thisDisplay.synthDef\_({ arg carrier\_freq, modulateur\_freq,

ampli, modulateur\_amp, panoramique ;

Machine Translated by Google

modulateur var , porteur, panneur ;

modulateur = SinOsc.ar(modulator\_freq,

mul:modulateur\_amp);

carrier = SinOsc.ar(carrier\_freq, modulateur, amp); casseroles = Pan2.ar(support, casserole); Out.ar(0, casseroles);

}, [\carrier\_freq, carrier\_freq, \modulator\_freq, modulateur\_freq, \amp, amp, \modulator\_amp, modulateur\_amp, \pan, pan]);

thisDisplay.name\_(" Modulation de phase");

}).montrer;

)

Comme l'AM, la FM modifie les timbres en produisant des tonalités supplémentaires, appelées tonalités latérales.

bandes. Grâce à Chowning de Stanford, nous savons comment fonctionnent ces bandes latérales.

Fréquence de bande latérale = fréquence porteuse + (n \* fréquence modulatrice)

n est un nombre entier. Imaginons que nous ayons une porteuse de 400 Hz et un

modulateur de 100 Hz. Le premier nombre entier est 0. 400 + (0 \* 100) = 400.

Alors est 1. 400 + (1 \* 100) = 500. Alors, un nombre entier est également ­1. 400 + (­1

\* 100) = 400­ 100 = 300. Alors 2 et ­2. 400 + (2 \* 100) = 600. 400 + (­2 \*

100) = 200. Théoriquement, n va de moins l’infini à l’infini positif. Comme

nous avons vu que l'amplitude de ces bandes latérales en FM régulière dépend de la l'amplitude du signal de modulation ET le rapport entre la fréquence porteuse

et la fréquence du modulateur. En modulation de phase, l'amplitude des bandes latérales dépend uniquement de l'amplitude du signal de modulation.

« Si le rapport entre les fréquences [du modulateur] et la porteuse est simple (1:1,

2:1, 3:2 etc.) les bandes latérales générées seront en séries harmoniques, et les

les tons complexes produits ressembleront aux structures harmoniques de vrais

instruments." (ROBERTS) À l’inverse, des fréquences non liées entraîneront

des sons bruyants et électroniques. Vous obtenez beaucoup de fréquences avec seulement deux oscillateurs

en FM, alors qu'en RM, AM ou synthèse additive, il faudrait de nombreux

oscillateurs. FM et PM sont des moyens efficaces pour générer des tonalités complexes.

# Autre modulation

La modulation de largeur d'impulsion utilise l'argument kwidth de Pulse.ar

### modulateur = SinOsc.ar(modulator\_freq, mul: modulateur\_amp); carrier = Pulse.ar (carrier\_freq, modulateur, amp);

N'importe quel type d'UGen ou valeur peut être utilisé pour modifier la largeur, comme n'importe quel UGen peut le faire.

être transmis comme (presque) n'importe quel argument à n'importe quel UGen. De nombreux synthétiseurs numériques

ont utilisé la modulation de largeur d'impulsion, souvent avec des ondes carrées modifiant la

largeurs des autres ondes carrées.

# Problèmes

1. Écrivez un affichage selon lequel AM et l'anneau modulent deux oscillateurs qui ne le sont pas.

ondes sinusoïdales.

1. Écrivez un affichage qui utilise un oscillateur non sinusoïdal comme modulateur de fréquence.
2. Écrivez un affichage qui utilise un oscillateur non sinusoïdal comme modulateur de phase.
3. Un patch chaos est un patch dans lequel trois oscillateurs FM se modulent mutuellement.

Écrivez un patch chaos de modulation de phase, afin qu'oscillateur1 modifie le phase de l'oscillateur2, 2 modifie 3 et 3 modifie 1.

1. Concevez votre propre affichage modulant en utilisant n'importe quel nombre d'oscillateurs dans

n'importe quelle configuration. N'oubliez pas d'éviter la polarisation DC.

1. Utilisez des enveloppes pour contrôler les amplitudes des fréquences de modulation. Tu peux

Trouvez un exemple de code pour utiliser des enveloppes dans les affichages au chapitre 5.

# Projet

Écrivez un morceau d'une ou deux minutes en utilisant deux types de modulation différents.