

Semesteraufgaben SC3, Teil 2

4. Programmiere einen "Oberton-Sequencer".
Als Klangquelle soll ein **"fetter" Sägezahn** dienen. Erzeuge dazu 2 oder mehr leicht gegeneinander verstimmte Sägezähne und mische sie.
Lege mit Hilfe geeigneter **Demand-rate** UGens und eines Impulsgenerators einen Sequenzer an.
Dieser Sequenzer soll in zufälliger oder sequentieller Reihenfolge Werte generieren, die die Frequenz eines nachfolgenden Resonators so steuert, daß sie immer mit **Harmonischen** (Obertönen) der Sägezahngrundfrequenz übereinstimmt. Geeignet ist ein Bereich zwischen der 3. und 12. Harmonischen.
Die durch den Demand UGen produzierten Resonanzfrequenzen sollen im UGen **Lag** mit 50 ms interpoliert werden.
Die auf diese Weise Oberton-gefilterte Sägezahnmischung soll irregulär zwischen dem linken und rechten Lautsprecher hin- und herwandern.
Tempo der Obertonsequenz sowie Grundfrequenz sollen bei laufendem Synth von außen steuerbar sein.

(4 Punkte)

5. Es soll ein Effektsynth programmiert werden.
Das monophone Inputsignal soll über **9 parallele Delaylines** mit je unterschiedlicher Verzögerungszeit (im Bereich von 1 bis 6 Sekunden) laufen.
Die verzögerten Signale sollen anschließend durch jeweils durch ein anderes **Resonanzfilter** laufen (Resonanzfrequenzen zwischen 500 und 2000 Hz, Filtergüte etwa 50).
Zuletzt sollen die 9 verzögerten und gefilterten Signale gleichmäßig über das **Stereopanorama** verteilt werden (von ganz links bis ganz rechts). Das Ausgangssignal soll also stereophon sein.

Beachte, daß der Out-UGen nur eindimensionale Arrays richtig verarbeiten kann!
Mehrdimensionale Arrays, wie sie bei dieser Aufgabe entstehen können, müssen vorher auf ein 1-dimensionales Array mit der gewünschten Anzahl von Elementen bzw. Kanälen auf geeignete Weise reduziert werden.

Teste anschließend diesen Synth mit einem Soundfile.
Erzeuge dazu einen Knoten mit diesem Effektsynth sowie einen Knoten mit einem Bufferplayer und achte auf die **richtige Reihenfolge**.

(5 Punkte)

6. Programmiere unter Zuhilfenahme der am 5. Kurstag (Granularsynthese) besprochenen Techniken (SynthDefs, **Task**) eine **granulare 1-kanalige Textur** aus einem Mono-Soundfile. Dabei soll die Dichte der Grains relativ gering sein (Overlap $\ll 2$).
Alle Grainsynths sollen auf dem Server in einer **Group** laufen - siehe dazu das Helpfile zu Group.

Diese Textur soll anschließend einen **Effektknoten** mit einer Art **"Hall"** aus Kammfiltern durchlaufen.
4 unterschiedliche parallele Kammfilter mit Delayzeiten zwischen 20 und 60 ms und Decays von etwa 1.5 Sekunden.
Die 4 Kammfiltersignale werden gleichmäßig zwischen links und rechts verteilt.

Beachte, daß die Group angelegt werden muss, bevor der Task die Synthknoten mit dieser Group als **Target** starten kann.
Beachte weiterhin, daß der Knoten mit dem Kammfiltereffekt hinter bzw. nach der Graingroup auf dem Server liegen muss.

(5 Punkte)