

reason-04.doc

氏名: クラス:

★Sampler ってなんだ？

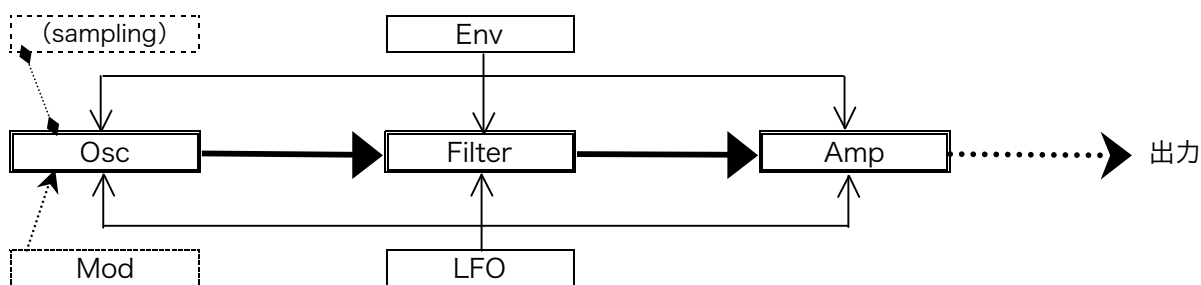
アナログタイプのシンセサイザーSubTractor をマスターしたとこで、次はサンプラー。

まず、アナログシンセとサンプラーとの違いってなんなんだ？とゆーと、
主要3セクション(①Osc, ②Filter, ③Amp)の内

アナログシンセの oscillator ⇒ [電子回路/演算]による周期的な波形
サンプラーの oscillator ⇒ [録音]されたあらゆる波形

よーするに違いは、①Osc が持つ波形だけ。ってこと

SubTractor の場合は、Osc に搭載されている32種類の基本波形 wave form を基に音を創り込んでったわけだが、
Sampler の場合には、録音された音をなんでも oscillator として使えるんで、その基本波形は無限にあるってこと。だ



上図のようにシンセもサンプラーも全く同じ構造だけど、サンプラーの場合には録音部にあたる(sampling)が Osc に基本波形 wave form を供給するんで、まずは録音しないと、サンプラーはまったく音を出せないってことね。

★NN-XT のマウント、の前に、、、

Reason には2種類のサンプラーデバイスが用意されているが、その内の1台「NN-19」は、機能が劣るだけで何もいいことがないんで、この授業では『NN-XT』のみを使う。

NN-XT のメインパネル



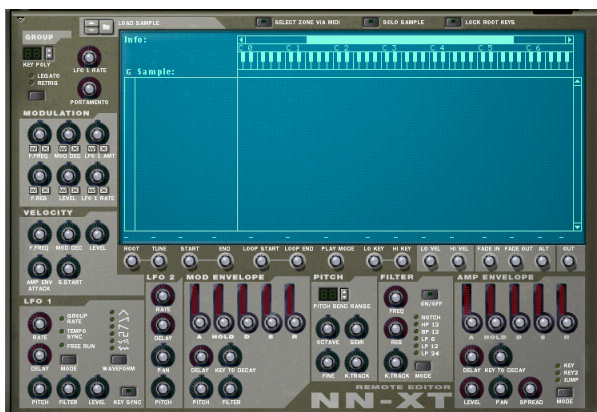
さあそれでは、とっととマウントして音を、ってわけに、サンプラーはいかない、なぜなら、サンプラーの Osc には wave form がなにひとつ入ってないんだから。

よって、まずは

- ・ Wave (.wav)
- ・ AIFF (.aif)
- ・ Sound Font(.sf2)
- ・ REX(.rex2, rex, rcy)

上記フォーマットの何れかで録音するわけだが、Reason では録音できない。

NN-XT のリモートエディターパネル



いちばん上の『hardware interface』は、MIDI IN と AUDIO OUT が合体したデバイスで、それぞれ外部の midi I/O と audio I/O に接続されている。これは

キーボード等 ⇒ midi in → [Reason] → audio out ⇒ ヘッドフォン等
ってな流れしかないことを示唆している。

よーするに、midi out や audio in は無いのだ。

reason-01.doc 『hardware interface』より

この授業では、CubaseとReasonが常にRewire接続されていることを前提に進める。なので、Cubaseで録音しちゃえ。

【実習-01】

Cubaseのプロジェクト・メニュー > プロジェクト設定を44.1kHz, 24bitにセッティングし、オーディオミックスダウンを使ってSubTractorの鋸歯状波 saw waveを220Hz (A2)の音程でmono録音しなさい。 ※テンポはFIXEDモード120BPM=1小節2秒で作業すること。SubTractorのFilter, Envelope, Mod類を切って、出来るだけ正確な鋸歯状波を得ること。

そもそも波形が周期的でなければ、音程(ピッチ)を認識できないので、ノイズ以外のあらゆる楽音は周期的な波形を持っていると言える。、そこで、

波形の1サイクルのみを生成し、それをループさせることによって音を伸ばさせてたのが…………… → シンセだったのに対し、

短い音をループさせて長い音を創るのも、最初から長い音を録音しといてそのまま使うのも両方あり → サンプラってところがまず重要。

本稿では、まず、サンプラーを理解する為のエチュードとして、サンプラーをアナログシンセ的に扱う。よって、アナログシンセ風の1サイクルが必要。なんだが、Reasonでdestructive(破壊的)な波形編集はできない。なので、Oscの基本波形 wave form 抽出にもCubaseを使う。

【実習-02】

Cubaseのサンプルエディタを使って、録音した鋸歯状波から1サイクルを切り抜きなさい。ゼロクロスポイントにスナップを使って、正確な1サイクルを得ること。

【実習-03】

Cubaseのオーディオ・メニュー > プロセッシングを使って、「DCオフセット除去」と「ノーマライズ(フルビット)」を実行しなさい。

【実習-04】

上記実習で得た(鋸歯状波, 220Hz, 1サイクル, フルビット)の独立ファイルを作成し、ファイル名をsaw.wavとしなさい。再生し、微かなピッチ感の正体はなんなのか考えよ。



【実習-05】 saw.wavについて、下記の括弧を埋めなさい。

プロジェクト設定が44.1kHzなので、saw.wavのsampling rateも() Hz、1秒=() サンプルとなる。よって、1サンプル約() msecとなる。

ヒント→ `$ echo "scale=10; 1000/44100" | bc`

※OSXのターミナルは何かと使えるんで、今後たまに登場する。echoは標準出力への書出し、scaleは小数点以下の桁数、bcは電卓。上向き矢印キーで前に打ったコマンドラインを呼び出せる。

220Hzの鋸歯状波は1サイクル約() msec=約() サンプルだ。

ヒント→ `$ echo "scale=10; 1000/220" | bc`

ヒント→ `$ echo "scale=10; (1000/220)/(1000/44100)" | bc`

プロジェクト設定が24bitなので、saw.wavのquantization bit rate 量子化ビット数も() bit、つまり、1つのサンプルが() 通りの値を取り得る。

ヒント→ `$ echo "2^24" | bc`

●実習-05の検証例

実習-05で得た答えをサンプルエディタウィンドウにて検証してみよう。

まず、[⌘+a]で saw.wav のすべてを選択した時

【横軸】: サンプルエディタの「選択範囲」数は、220Hz のサンプル数(正の整数, 自然数)と一致する。

そして、鉛筆ツールで編集可能な

【縦軸】: 振り幅(プラス 100%〜マイナス 100%)の解像度は、量子化ビット数と一致する。

おまけに、プロジェクトウィンドウに戻って、saw.wav の開始位置を小節番号位置 1.1.1.0 にセットし、[p]キーにて「左右ロケーターを選択範囲に設定」し、[パッド 2]キーにて「右ロケーター位置に移動」した時、トランスポートウィンドウのタイムディスプレイに表示される“秒” & “サンプル数”は、220Hz, 1 サイクルの計算結果と一致するはずだ。その秒数はプールの情報欄等でも確認できる。

【実習-06】

上記にならない 実習-05 の検証例 を実行し、確認しなさい。

普段使ってるエフェクター類のパラメーター設定に比べると、やたら細かい数字が出てくる。が、デジタル・オーディオの最小単位はサンプルなんで、これより細かいのは無い。ここまで理解できたら、後々極楽。

●実習-01〜05の応用例

サンプルエディタウィンドウでミュージカルモードにした1サイクルは、プロジェクトウィンドウにて選択した1サイクルを [⌘+k] で反復複製させることにより、BPM で正確にピッチを変えられる。純正律ならオーディオテンポの 5/4 倍で M3rd、4/3 倍で P4th、3/2 倍で P5th、2/1 倍でオクターブ、それぞれの逆比でインバージョン。他のインターバルも全て自然倍音列から算出できる。

平均律なら半音数 n につき BPM が

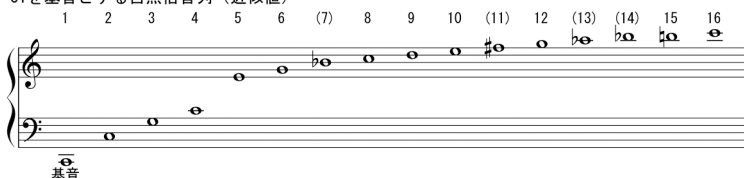
$$f(n)/f(\text{基準音}) = \sqrt[12]{2^n} = 2^{n \times (1/12)} = 2^{n/12}$$

倍ってこと。

ってことは、Cubase のテンポトラックでメロディ鳴らせちゃうってことね。

これが、サンプラーの原理だ。

Cl を基準とする自然倍音列 (近似値)



【実習-07】

トランスポートウィンドウにてテンポを TRACK モードに設定し、プロジェクトメニュー > ブラウザ にてプロジェクトブラウザを表示し、テンポトラックに右図の通り BPM を入力しなさい。

saw.wav をトラックの開始位置 1.1.1.0 に貼付け、独立ファイルを作成しなさい。オーディオファイル名は free1.wav に変更すること。

サンプルエディタウィンドウでミュージカルモードにした free1.wav をプロジェクトウィンドウで選択し、[⌘+k] で開始位置 1.1.1.0 から 1 1 5 0 回反復複製 (共有コピー) しなさい。

プロジェクトメニュー > テンポトラック を確認しながら開始位置 1.1.1.0 から再生し、誰の羊か叫びなさい。

【実習-08】

鉛筆ツールで激しく波形を書き換え誰の羊か叫びなさい。気が済んだら保存しておくこと。

【実習-09】

講師の指示に従って指定のディレクトリに、上記実習で作成した saw.wav と free1.wav をコピーしなさい。

さて、

Reason に戻って、サンプラーの最終準備を。。。

【実習-10】

Reason の Empty Rack に、『Mixer 14:2』『RV-7』『DDL-1』以上3つのデバイスをマウントし、『Mixer 14:2』の Master Out を『hardware interface』の 1,2ch に、エフェクターを『Mixer 14:2』の Send, Return にルーティングしなさい。(今後は毎回そうすること)

【実習-11】

『NN-XT』もマウントし、1/L, 2/Rch を『Mixer 14:2』の 1ch にルーティングしなさい。

ポジション	テンポ	タイプ
1. 1. 1. 0	150.0000	ジャンプ
1. 2. 1. 0	135.0000	ジャンプ
1. 2. 3. 0	120.0000	ジャンプ
1. 3. 2. 0	135.0000	ジャンプ
1. 4. 1. 0	150.0000	ジャンプ
1. 4. 4. 0	300.0000	ジャンプ
2. 1. 1. 0	150.0000	ジャンプ
2. 1. 4. 0	300.0000	ジャンプ
2. 2. 1. 0	150.0000	ジャンプ
2. 3. 3. 0	135.0000	ジャンプ
2. 4. 2. 0	160.0000	ジャンプ
3. 1. 2. 0	150.0000	ジャンプ
3. 2. 3. 0	135.0000	ジャンプ
3. 3. 1. 0	120.0000	ジャンプ

【実習-12】

Device を Keep したまま『Sequencer』の全トラックを Delete し、Cubase の midi out を『NN-XT』に接続しなさい。(今後は毎回そうすること)

【実習-13】

『NN-XT』のリモートエディタパネルを開き、内部ルーティングと各機能をイメージしながら全パラメーターを凝視しなさい。

サンプラーといえども、シンセの一種なんだから、基本となるアナログタイプの SubTractor をマスターした君達なら、『NN-XT』のシンセパラメーターだってすぐに理解できるはずだ。新しいのはキーマップだけね。

★NN-XT ①キーマップディスプレイ_その壱

『短い音をループさせて長い音を創る』

録音&波形編集したことで、Osc の準備は整った。

SubTractor の場合には、1つの Osc セクションが全ノートナンバー(0~127) & 全ベロシティ(0~127)に対し自動接続されたが、サンプラーは複数の Osc を様々なパラメーターからトリガーすることができる。

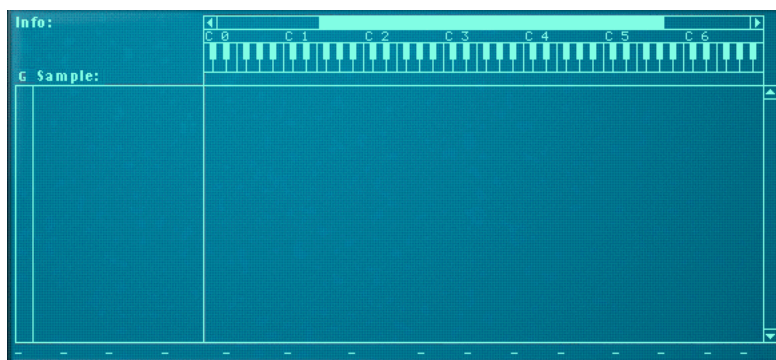
サンプラーの場合、Osc セクションが無限にあるってことね。ってことはサンプラーの場合、Osc の後ろに続く Filter セクションも Amp セクションも、Osc の数だけ無限にある。ってことだ。

そこで、まずは Osc(サンプル)を、任意のノートナンバー&ベロシティ(ゾーン)に配置する。

そこで、SubTractor には無かったキーマップディスプレイなるものが必要となる。

キーマップディスプレイ

まずは、このキーマップディスプレイ中にサンプルを読み込ませる。



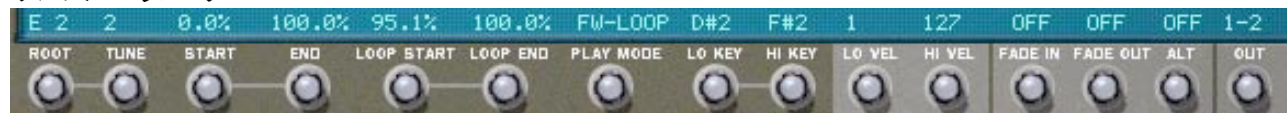
【01】キーマップディスプレイを右クリック (cont + クリック) すると、Edit メニューが出てくるので、“Browse Samples”を選ぶ。

【02】Sample Browser で、任意のサンプルを読み込む。読み込まれたサンプルのことをゾーンってゆう。

【03】ゾーンの Play Mode を設定する。

【04】ゾーンのキーレンジを設定する。そんな時 LOCK ROOT KEYS を有効にしとけばキーレンジを移動してもトランスポートされない。チューニングが怪しければ ROOT にてルートノートを確認、更に TUNE にて cent 値を確認。

サンプルパラメーター



・[左端の ROOT~Hi KEY]は、1ゾーンずつのエディットのみ可能。(シングルアジャストメントパラメーター)

・[LO VEL~右端の OUT]は、選択された全ゾーンを同時にエディット可能。(マルチアジャストメントパラメーター)

Play Mode

- FW (forward) → ループ無し
- FW-LOOP → スタート-エンドポイント間を無限ループ
- FW - BW → 前向き/ 後ろ向きを繰り返しながら無限ループ
- FW-SUS → 鍵盤が押されている間のみ FW-LOOP
- BW (backward) → ループ無しのリバース

[LO VEL, HI VEL]は、ベロシティによるサンプル切り換えを実行する場合、ゾーンのベロシティレンジ設定に使う。

[FADE IN, FADE OUT]は、ベロシティレンジが重なり合うゾーンを選択した場合の、クロスフェード設定に使う。

[ALT]は、スネアロール等で一発ごとにサンプルを切り換えたい場合、重なり合うゾーンを交互 alternate に再生する。

【実習－14】

『NN-XT』に saw.wav を読み込み、saw.wav ゾーンの PLAY MODE を FW-LOOP にし、「ピッチ変化=再生スピード変化」だててことを実感し、感動せよ。

【実習－15】

PLAY MODE を使って、saw.wav から三角波 Triangle wave を作りなさい。

【実習－16】

『NN-XT』に free1.wav も読み込み、2つのオーディオファイルから複数の音色を創り、キーレンジ・ベロシティレンジに留意しつつ、『NN-XT』1台のみで saw wave music を創りなさい。

★NN-XT ①キーマップディスプレイ_その式

『最初から長い音を録音しといてそのまま使う』

そんぢゃ、

『短い音をループさせて長い音を創る』ってのはやったんで、『最初から長い音を録音しといてそのまま使う』ってのにいってみよー。

録音に際して、心構えを軽く。

リッチサウンド 情報量が多い	↔	チープサウンド 情報量が少ない
リッチな録音技術 原音の情報だけを余すこと無く取り込む	↔	プアーな録音技術 必要な情報が間引されちゃってる

録音に必要なスキルはRec実習にて獲得してもらおうとして、大まかにフルビットに近い録音ほどリソースを有効活用すると言える。ちなみに synthesizer におけるチープサウンドは狙いとして当然アリだ。が、それもこれもリッチな録音技術が前提。現実もしくは美学としての貧乏はアリだけど、貧乏くさいのは絶対許さん。

デジタル録音する際には、必ず「ケイ素」(半導体)と「水晶」(ワードクロック)と「イカの内蔵」(液晶)に感謝してからプラグを突っ込むこと。

【実習－17】

Cubase で録音しなさい。録音方法,ディレクトリ,ファイルフォーマット,ファイル名は講師の指示に従うこと。iMac 内蔵マイクで録音するもよし、tascam の I/O で録音するもよし、まずは声でも具体音でも数種類をハイファイで。

【実習－18】

録音したファイルを「DC オフセット除去」実行後、「ゼロクロスポイントにスナップ」を使って任意の開始位置から切り分け複数の「独立ファイル」を作成し、「ノーマライズ (フルビット)」を実行しなさい。分かりやすいファイル名をつけとくこと。

NN-XT には、個々のサンプルに対応したゾーンの上層構造として、複数のゾーンをまとめたグループってのがある。

よおーするに、あらゆるゾーンはなんらかのグループに属してることだ。

複数のゾーンを選択後、右クリック > Group Selectec Zones にて作成されたグループは、グループコラム(ゾーン名左横の縦線)をクリックすることで、グループ内のゾーンを全て一発で選択することができる。

グループパラメーター



グループパラメーターは、グループ内のみで有効な4つの機能をコントロールする。

①KEY POLY : 同時に演奏できるキーのポリ数。例えばグループ化したハイハット類のポリ数を1にすると、クローズハットが鳴ると同時にオープンハットが消える。

②LEGATO : アサインキーを全て使い切った次のノートで、エンベロープをトリガーしない。結果、モノモード時にはレガートになる。 RETRIG : 通常の設定

③LFO 1 RATE : LFO1 が“Group Rate” モードで使用されている場合のレートコントロール。その場合、LFO1 セクションのレートパラメーターよりこっちが優先される。

④PORTAMENTO : ポルタメントタイムのコントロール。レガートモードでは、レガートノートのみポルタメントがかかる。

【実習－19】

実習17,18で作成したオーディオファイルをゾーンに配置し、『NN-XT』 & Cubase でサンプルリングミュージックを創りなさい。必要であれば複数台の『NN-XT』を使うべし。