

Universidade de Brasília
Departamento de Desenho Industrial
Diplomação
Professor Orientador: Rogério Câmara
Orientando: Vicente Gomes da Silva Neto - 110021126

Desenvolvendo um Instrumento Sintetizador Virtual

Mojo Bass v1.0



FIL ENV

ATT DEC SUS REL

11 32 0 9



AMP ENV

ATT DEC SUS REL

10 37 0 9



OSC 1

TUNE SEMI FINE

-24 0 0

PHASE

WAVE

SAW

SYNC

0

OSC 2

TUNE SEMI FINE

-24 0 -0.1024

PHASE

WAVE

SAW

SYNC

0

PHASE 1+2

FILTER

LP24

CUTOFF RESO

38 0

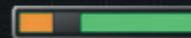
ENV AMT. KEYTRK

1 0

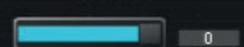
SCOPE



MIXER



OUTPUT



Sumário

| | |
|---|----|
| 1- Introdução | 4 |
| 2- Objetivos..... | 4 |
| 3- Glossário..... | 5 |
| 4- Primeira fase de desenvolvimento: Plugin de efeitos..... | 9 |
| 4.1- Design de conceito | 9 |
| 4.2- Design da GUI | 13 |
| 5- Segunda fase de desenvolvimento: Sintetizador | 16 |
| 5.1- Definindo o produto | 16 |
| 5.2- Conceituando | 18 |
| 5.3- O Desenvolvimento | 20 |
| 5.3.1- Reaktor | 20 |
| 5.3.2- Elementos gráficos | 29 |
| 6- Considerações finais | 39 |
| 7- Bibliografia e referências | 40 |

Resumo

Este relatório trata do desenvolvimento de um plugin de áudio virtual, no caso, um sintetizador virtual, um instrumento para criação musical que é utilizado através de um computador. O relatório descreve o projeto em suas diferentes etapas, desde a conceituação até a elaboração do software e sua parte gráfica, assim como os problemas encontrados e as soluções desenvolvidas. Explica-se brevemente sobre as ferramentas utilizadas para o desenvolvimento: Knobman e Reaktor 6.

Palavras-Chaves

Áudio - Plugin - Sintetizador - GUI - Knobman - Reaktor 6

1 - Introdução

Apresenta-se neste relatório o desenvolvimento de um plugin de áudio virtual, o que se configura como um sintetizador virtual. Há poucas publicações sobre desenvolvimento de plugins de áudio, e nada foi encontrado sobre desenvolvimento de Interfaces Gráficas com esse tipo de produto. Há, no entanto, bibliografia sobre aspectos técnicos de engenharia de áudio, que serviu como base teórica para o desenvolvimento do produto.

Durante o desenvolvimento do projeto recorreremos à usuários de produtos equivalentes e a blogs e fóruns online dedicados ao desenvolvimento de plugins de áudio. Grande parte do embasamento técnico e teórico se deu de forma empírica, pois foi através de minhas experiências e contato com profissionais da área que pude compreender e corrigir os problemas identificados no percurso de desenvolvimento do projeto. .

Inicialmente, a proposta de projeto era desenvolver a GUI (graphical user interface) de um plugin já em desenvolvimento. Entrei em contato com profissionais da área que estavam desenvolvendo um plugin de phasing e precisavam de um profissional para desenvolver a interface. Me candidatei, mas logo percebi que era um projeto conduzido em ritmo de “hobby”, as pessoas demoravam a responder mensagens e quando eu dava alguma palpite técnico era como conversar com uma parede. Abandonei o projeto e optei por desenvolver um plugin integralmente, isto é todas as funções e aspectos do programa e interface gráfica. Mudança drástica, pois acrescentou número considerável de camadas de complexidade ao projeto e alterou a abordagem metodológica.

Durante o segundo semestre de 2017, cheguei a estruturar a proposta de criar um rack de efeitos, isto é, uma ferramenta para modular o som, processo que será descrito neste relatório. No primeiro semestre de 2018, após conversa com um amigo, percebi que existia uma lacuna que eu poderia preencher para um nicho específico de música eletrônica: o trance. O processo de síntese sonora de linhas de baixo para trance e psy trance é executado de maneira bem específica. Procurei desenvolver um produto que pudesse otimizar aquele processo, que auxiliaria produtores e iniciantes.

O desenvolvimento do sintetizador foi realizado na plataforma Reaktor, da Native Instruments, da Alemanha. Para o desenvolvimento da GUI, utilizei os softwares Adobe Illustrator, Photoshop e um software chamado Knobman.

Utilize os arquivos disponíveis em: <http://bit.do/eoMfn> para acompanhar o relatório quando for mencionado algum arquivo de referência, já que alguns exemplos de coisas ditas aqui são melhor compreendidos com uma referência sonora.

2 - Objetivos

O objetivo deste projeto é criar um produto que seja utilizado como uma ferramenta útil no processo de criação de linhas de baixo para psytrance. O sintetizador segue uma série de conceitos pré-determinados em sua concepção. A intenção é a de criar um sintetizador que seja:

- 1- fácil de usar, no sentido de permitir ao usuário a criação rápida de sons
- 2- Leve no uso de processamento (CPU) do computador
- 3- Deve soar tecnicamente bem, sem problemas de clip ou cancelamento de fase

Deseja-se contribuir com o cenário de produção de música eletrônica independente no Brasil, fornecendo uma ferramenta de uso fácil e design moderno.

3 - Glossário

Há um extenso conjunto de terminologias que não possui tradução direta para o português, muitos dos termos podem ser considerados “jargão” e por isso a necessidade de se caracterizar essa terminologia na parte inicial deste relatório. Como referências bibliográficas para estas definições foram utilizados: Sintetizadores à Brasileira, de Francisco Edson de Souza Pereira (2003), o livro Fundamentals of Digital Audio, de Alan P. Kefauver e David Patschke (2007) e o glossário site da empresa Recording Connection (<https://www.recordingconnection.com/glossary/>).

DAW: Sigla para Digital Audio Workstation (Estação de trabalho de áudio digital, ou estação digital de trabalho de áudio). São dispositivos eletrônicos ou peças de software computacional utilizados para a gravação, edição e produção de arquivos de áudio, músicas, fala ou efeitos especiais. DAWs vêm em grande variedade de configurações, desde um único programa de computador até suítes integradas complexas. Independente de suas configurações, as DAWs modernas possuem interface central que permite ao usuário alterar e misturar múltiplas faixas sonoras em uma única peça final. Uma DAW, no contexto da produção de música, seria o equivalente a, por exemplo, o Photoshop no mundo do Design Gráfico e as faixas de áudio dentro de uma DAW são o equivalente às camadas no Photoshop. Exemplos de DAWs: Ableton Live, Logic Pro, Cubase, Reason, Reaper, Audacity.

Plugin: Componente de software utilizado como extensão para alterar ou acrescentar sinais relacionados à áudio dentro de um programa anfitrião (DAW). Existem inúmeros plugins compatíveis com as mais variadas DAWs, cada um desses plugins possuindo sua funcionalidade única, expandindo ainda mais a variedade de sons e manipulações possíveis. Plugins podem ser classificados em 3 tipos: Sintetizadores, Samplers e Efeitos. Exemplos de plugins de áudio de efeitos: Waves H-Delay, Fab-Filter Saturn, Xfer Records LFO Tools. Exemplos de plugins de sintetizador: Xfer Records Serum, NI Massive. Exemplos de plugins samplers: Ableton Simpler, Izotope Breaktweaker, NI Battery, Apple Ultrabeat.

Regulador: Reguladores são os parâmetros de comando utilizados para controlar as diferentes funções de um plugin. Podem se apresentar de diversas maneiras: Botões, Faders, Sliders, Knobs, Switches e Pads, por exemplo. Detalhando:

Knob: Um *knob* é um botão de controle rotativo, onde se determina o valor desejado rotacionando o botão. Exemplos práticos são os botões giratórios que controlam o volume em sistemas de som, potenciômetros de rádio.

Fader: um *Fader* é um botão de controle em que o valor desejado é determinado arrastando o botão ao longo de um trilho (potenciômetro) até o valor determinado. Alguns exemplos são os botões de equalização em alguns sistemas de som de alta definição, controles de volume em consoles de mixagem e mesas de som.

Switch: É um componente que permite ao usuário escolher uma de uma seleção predefinida de opções em uma interface. Um exemplo prático de um switch que todos conhecem é o interruptor de luz: o usuário pode escolher entre 2 opções, luz ligada ou luz apagada. Num plugin de áudio isso pode ser utilizado de diversas formas, como por exemplo para escolher a curva de um corte de EQ. Alguns *switches* podem possuir múltiplos estados, fazendo com que seja mais prático que a informação seja mostrada de outras maneiras, como por exemplo um *menu drop-down* que mostra todas os diferentes estados daquela *switch*.

Pad: Uma área pré determinada em que o usuário pode definir um parâmetro deslocando um ponto no eixo x,y desta área pré determinada. Usado em funções que necessitam de duas variáveis para gerar uma saída.

GUI: Sigla para Graphical User Interface, ou Interface Gráfica de Usuário. É um tipo de interface de usuário que permite aos usuários interagirem com dispositivos eletrônicos através de

ícones gráficos e indicadores visuais. GUIs foram introduzidas em resposta à dificuldade de se aprender a controlar dispositivos eletrônicos através de linhas de comando.

Equalização, ou EQ: é o processo de ajustar o balanço entre diferentes bandas de frequência sonora dentro de um sinal elétrico. Esses dispositivos são utilizados para aumentar ou diminuir a energia de bandas sonoras específicas no espectro sonoro através de filtros. Equalizadores podem ser gráficos ou paramétricos.

Filtro ou Filter: É um dispositivo que remove frequências de uma banda desejada. É um tipo de processamento de sinal elétrico, em que se remove completa ou parcialmente algum aspecto do sinal de áudio. Para não se confundir com a equalização, entenda que um equalizador possui vários filtros dentro de si os quais são utilizados para filtrar as frequências indesejadas. A banda de frequência na qual o filtro agirá é determinada pelo **cutoff**. Filtros são uma parte complexa e fundamental no desenvolvimento de meu plugin e por isso irei definir mais precisamente alguns parâmetros, assim como os diferentes tipos de filtro que podem ser utilizados em diferentes aplicações:

Cutoff: É o ponto de frequência a partir do qual o filtro deixa passar o sinal de áudio.

Low-pass(ou High-cut) Filter: É um filtro que rejeita as bandas agudas e deixa as frequências graves passarem.

High-Pass(ou Low-cut) Filter: É um filtro que rejeita as bandas graves e deixa as frequências agudas passarem.

Band-Pass Filter: É um filtro que rejeita todas as frequências que não se encontram dentro da banda determinada pelo cutoff, e deixa passar o sinal que se encontra dentro da banda determinada.

Notch ou Band-reject Filter: É um filtro que rejeita as bandas selecionadas pelo ponto de cutoff, e deixa passar as que estão de fora. Basicamente o oposto do Band-Pass.

Comb Filter: Um filtro que possui múltiplas bandas estreitas espaçadas regularmente, o que dá uma aparência de pente de cabelo ao filtro, por isso o nome comb (pente).

- **Ganho/gain:** É a medida de um circuito, geralmente de amplificação, que pode ser utilizado para aumentar/diminuir a energia ou amplitude de um sinal de áudio. É o equivalente a um botão de controle de volume em um sistema de som, no entanto, pode ser utilizado tanto para aumentar quanto diminuir a amplitude de um sinal antes que ele alcance a saída final, permitindo um controle melhor do sinal.

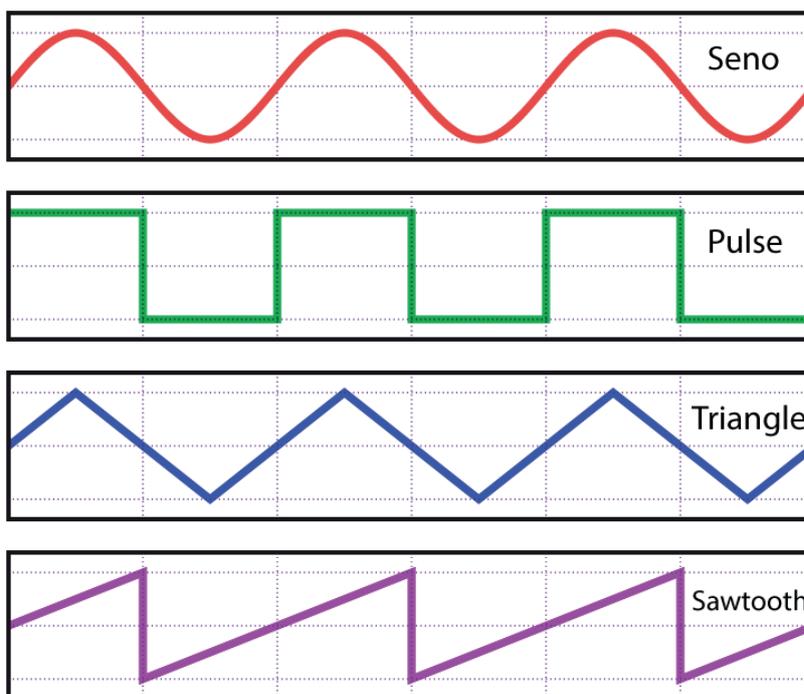
- **Distorção/Distortion:** É a alteração da forma inicial da onda de sinal de áudio. Quando um sinal ultrapassa o limite físico do meio que o emite, o sinal sofre distorção. Por exemplo, se a sua caixa de som possui um limite físico de +10dB, ao tentar emitir um sinal de +11dB através daquela caixa o som será distorcido. A distorção acrescenta uma série de bandas harmônicas ao sinal original e pode muitas vezes ser indesejada. No entanto, há casos em que a distorção pode ser desejável. No heavy metal e rock'n roll, por exemplo, as guitarras têm seu sinal distorcido pesadamente para que seja obtido uma timbragem característica. Em plug-ins de áudio temos a opção de distorcer o sinal de áudio de maneira sutil (no intuito de dar brilho ou corpo a um timbre) ou exagerada (no intuito de destruir o sinal original por qualquer motivo que seja). Existem, também, diferentes modos de se distorcer um sinal. Para que possamos compreender melhor o que é a distorção em si, precisamos entender primeiramente de onde ela veio.

Existem diferentes tipos de distorção, que emulam distorção em diversos tipos de equipamento.. Alguns desses modos de distorção são: Tape distortion, Overdrive, Tube distortion, Rectify, Sine Fold, Waveshaping.

- **LFO:** Em inglês, Low Frequency oscillator, ou Oscilador de Frequências Baixas, é um oscilador que gera uma onda geralmente abaixo de 20Hz, ou seja, no campo inaudível do espectro sonoro. O sinal produzido pelo LFO pode ser utilizado como fonte de modulação para outros

parâmetros dentro de um plugin. Por exemplo, posso ordenar que o LFO module o volume do filtro, e se o LFO estiver oscilando em uma frequência de 100hz, o volume do filtro irá subir e descer na frequência de 100hz. Exemplo de funcionamento: <https://www.youtube.com/watch?v=kpjkQDBHtjU>

- **Oscilador:** É um circuito eletrônico (no nosso caso virtual) que produz um sinal eletrônico periódico e oscilante. Um oscilador de áudio produz frequências entre 16 Hz e 20kHz geralmente. Osciladores geram ondas eletrônicas, ou pulsos, e a frequência com que são gerados esses pulsos determina a nota gerada. Existem vários tipos de ondas, mas para este projeto cobrimos apenas 4: Onda seno (sine), Sawtooth, Pulse e Triangle. Cada onda pode ser ouvida em seu respectivo arquivo de referência dentro da pasta Ondas Sonoras. Observe a imagem abaixo:



- **Osc Resync:** Faz com que sempre a mesma fase de uma onda seja ativada ao se enviar voltagem para o oscilador. Sem o osc resync ativado, a fase é dessincronizada do pressionar de teclas do teclado.

- **Tune:** Tune se refere a qualquer tipo de parâmetro que altere a característica do tom. Fine-tuning é um parâmetro encontrado no sintetizador, também conhecido como microafinação.

- **Envelope:** Quando um instrumento acústico produz som, o volume e características sonoras mudam de acordo com o tempo de maneiras que variam de acordo com o instrumento. Por exemplo, quando se toca uma guitarra com palheta ou uma nota em um piano, cada um deles irá emitir um som que se inicia, se mantém por um período de tempo e depois desaparece. Sintetizadores podem se utilizar de envelopes para controlar esse tipo de característica em seu processo de síntese sonora. Envelopes podem ser encontrados nos mais variados formatos e métodos. Para esse sintetizador nos aprofundaremos em um tipo específico de envelope, o ADSR, ou Attack, Decay, Sustain, Release envelope.

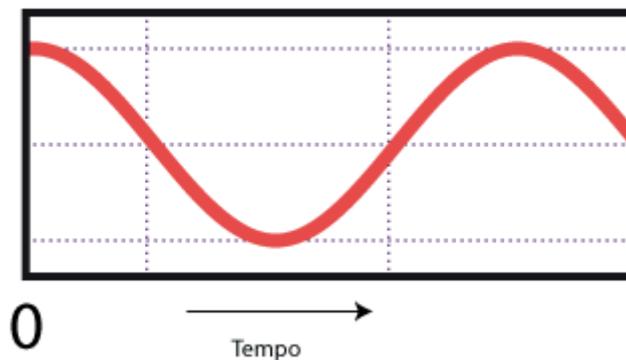
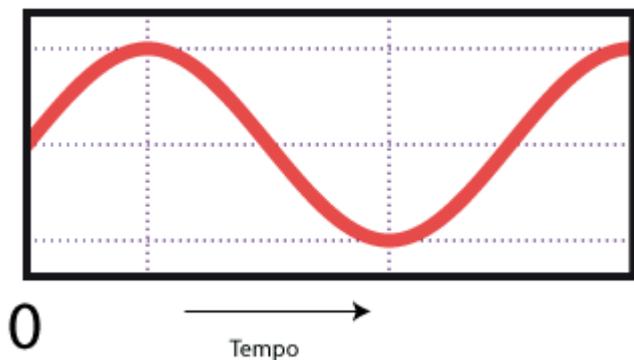
- **Attack** é o tempo que leva ao som para sair de 0 volume ao pico máximo quando a tecla é pressionada.

- **Decay** é o tempo que leva ao som para que subsequentemente o volume decaia ao valor definido pelo sustain.

- **Sustain** é o nível da duração principal do som, isso é enquanto a tecla estiver pressionada, o nível de sustain será o emitido.

- **Release** é o tempo que leva para que o som decaia do nível de sustain até zero volume, contado a partir do momento que a tecla é solta.

- **Phase/Fase:** É a posição da onda em relação ao tempo. Veja o exemplo abaixo, a primeira imagem mostra uma onda seno e a segunda mostra a mesma onda quando a sua fase é deslocada.



3- A primeira fase do projeto: desenvolvendo o rack de efeitos

Inicialmente, conforme já foi dito na introdução, seria desenvolvido apenas uma interface gráfica de usuário para o plugin, em seguida cogitou-se desenvolver um rack de efeitos, por fim, no início de 2018 o projeto foi alterado para o desenvolvimento do sintetizador.

A primeira versão do projeto seria realizado em parceria com profissionais que conheci na internet que estavam desenvolvendo um plugin open source, que seria disponibilizado gratuitamente no futuro. O plugin era de baixa complexidade, com apenas 4 reguladores e de tremolo (exemplo de tremolo: https://www.youtube.com/watch?v=_tLC7wiV7TI). Devido à dificuldades de comunicação e distância física, logo pude perceber que seria impossível conduzir o projeto no ritmo que eu desejava. Assim, cheguei à conclusão que seria melhor desenvolver um plugin “imaginário” e uma interface para tal plugin.

No entanto, o desenvolvimento deveria acontecer em duas etapas:

1 : desenvolvimento do conceito gráfico, identidade visual e elementos animados.

2: desenvolvimento da GUI em si, ou seja, programar cada elemento em seu devido local com suas devidas animações e funções.

A primeira etapa demanda majoritariamente de conhecimentos de design gráfico e suas aplicações no desenvolvimento de GUIs. Já a segunda demanda variados conhecimentos de programação. Em minha pesquisa logo descobri algumas SDKs que poderiam me ajudar a desenvolver o projeto. A que mais interessou inicialmente foi a plataforma JUCE. Essa plataforma possui interfaces que permitem a criação de GUIs de maneira mais simplificada que apenas linhas de código. No entanto, eu ainda necessitaria aprender o básico de C++, uma linguagem de programação extremamente complexa, para que pudesse ter um trabalho satisfatório, o que envolvia grande carga de trabalho.

Em 2018 procurei explorar melhor o potencial da plataforma Reaktor, desenvolvida pela Native Instruments, na Alemanha; o que permitiu desenvolver o projeto de maneira mais fácil.

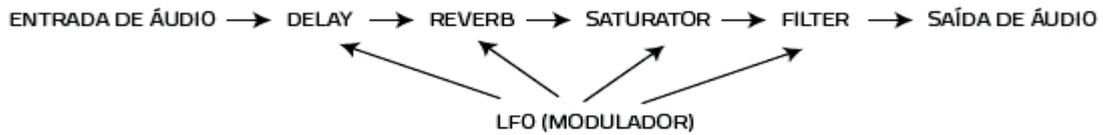
A seguir, falarei separadamente dessas frentes neste relatório, sendo uma frente de Design Gráfico per se e a outra de Design de Conceito. Começaremos pela parte de conceituação do produto.

3.1- Design de Conceito

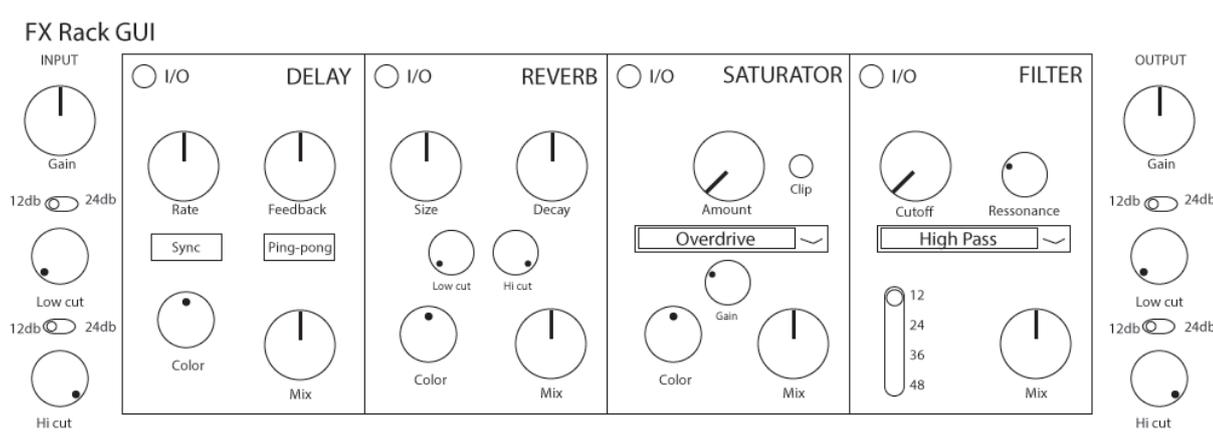
Como já foi dito, existem diferentes tipos de plugins de áudio, eles podem ser sintetizadores, samplers ou módulos de efeito. No princípio do projeto a intenção era de fazer um plugin de efeitos. Imaginar e conceituar um plugin de efeitos, não foi uma tarefa muito complexa. Através de conversas e debates com colegas produtores e engenheiros de áudio, além de minha experiência empírica com produção, pude perceber que havia espaço no desenvolvimento de um tipo específico de plugin: um rack de efeitos. O plugin seria utilizado como ferramenta de melhoramento sonoro, timbragem criativa e correção de problemas. Primeiramente, pensei no plugin como entidade à parte das DAWs, como se fosse um pedaço de hardware que existisse e quais seriam os elementos essenciais em um rack de efeitos com essa função. Nas conversas que tive com músicos, engenheiros e produtores, percebi uma predominância no uso de 4 efeitos em especial: Delay, Reverb, Filtros em geral e Saturadores (distorcedores de sinal). Pretende-se unir esses 4 módulos de efeitos diferentes em uma peça só; por isso chamo de rack (prateleira), que seria como uma mesa que possui vários efeitos em uma só.

Para tanto, definiu-se quais parâmetros devem ser controláveis para cada efeito em meu plugin, pois para que eu possa desenvolver a interface preciso de algumas informações essenciais: quantos reguladores cada módulo terá? Quais serão os tipos de reguladores em cada módulo?

Inicialmente se pensou como seria o caminho do sinal de áudio que passaria. Dito isso, foi esquematizado brevemente um “signal flow”, observe:



Segundo esse esquema o plugin deveria conter 6 módulos. Na sequência deveria se pensar quantos reguladores e quais funções teriam cada um desses módulos. Para isso, foi realizado um wireframe levando em conta minha experiência prévia com áudio e módulos já existentes.



Chamaremos essa versão de “Alpha 0.1”. Input e Output são os módulos de entrada de áudio e saída de áudio, respectivamente. Para justificar as escolhas de reguladores, entraremos em maior profundidade nos módulos.

3.1.1- Input e Output

Os módulos são o que se pode chamar de “corretores”. Observe que, na prática, são iguais. A única diferença entre ambos é o posicionamento na cadeia de processamento. O módulo de input serve para processar o sinal de áudio antes que ele passe pelo módulo central de efeitos, isto é, para que se possam fazer ajustes de ganho (gain) e equalização no sinal antes que ele passe pelos efeitos em si. Por que isso é necessário?

Para explicar um pouco melhor, temos que entender os reguladores que coloquei nos módulos. Novamente, como referências bibliográficas para estas definições foram utilizados: Sintetizadores à Brasileira, de Francisco Edson de Souza Pereira (2003), o livro Fundamentals of Digital Audio, de Alan P. Kefauver e David Patschke (2007) e o glossário site da empresa Recording Connection (<https://www.recordingconnection.com/glossary/>).

Gain: Este módulo foi colocado no input para que o produtor possa ajustar a amplitude (volume) do sinal antes que ele passe pelo módulo central. Essa ferramenta serve tanto para que não ocorra distorção indesejada no sinal durante o processamento (caso o sinal esteja muito forte) quanto para que se possa aumentar a energia de sinais baixos demais.

Low Cut (High Pass): O low cut pode, também, ser chamado de High-Pass Filter. É uma ferramenta de equalização utilizada para se remover frequências graves. Por que é necessário remover frequências graves? Muitas vezes, frequências graves são indesejadas na maioria dos timbres, a não ser em percussões e baixos. Frequências graves que tocam ao mesmo tempo tendem a gerar problemas como cancelamento de fase e ou “muddiness” (sujeira), um termo utilizado para denotar um som confuso e “sujo”. Se o sinal possui frequências graves, quase sempre será desejável removê-las, e por isso o

regulador de Low-cut neste módulo. Observe que, abaixo do knob, está um switch em que o usuário pode selecionar entre dois modos: 12dB e 24dB. Esses números representam a curva de corte no ponto de cutoff, isto é, quantos decibéis por oitava serão removidos no cutoff. Para se remover mais, seleciona-se 24dB e para um som mais natural, pode-se manter a curva de 12dB.

High Cut (low Pass): Basicamente a mesma função do knob de low cut, no entanto, este serve para que se removam as frequências mais agudas. Excesso de energia nas bandas agudas pode gerar o que engenheiros chamam de “hiss”, um tipo de chiado que suja o som e inclusive pode fazer com que o som machuque o ouvido de quem o escuta em determinados volumes.

3.1.2- Delay

O efeito Delay é dos mais versáteis para a produção de música. É um efeito que grava e armazena um sinal de áudio e então toca esse sinal após um período de tempo. Se chama delay justamente por conta da noção de atraso que o efeito dá a um som. Esse período entre a gravação e o playback do som pode variar em média de 2 milissegundos até em alguns casos 10 segundos, dependendo algoritmo usado. Períodos muito curtos são usados geralmente para dar cor e corpo a um timbre, enquanto períodos longos são geralmente utilizados como uma ferramenta de composição

Explicando os reguladores que escolhi para este módulo:

Rate: Rate é o período de tempo entre a gravação e a reprodução do som que está sendo processado. Esse período geralmente é medido em Hz. É possível programar o tempo do delay independente da BPM do projeto sendo desenvolvido, mas para melhor sonoridade as vezes é melhor sincronizar o tempo do delay com o do projeto musical que está sendo trabalhado.

Sync: Botão que permite ao usuário escolher se quer que o rate do delay seja sincronizado com a BPM do projeto ou não.

Feedback: O feedback é um parâmetro que determina a quantidade do som reproduzido que será reintroduzido (realimentado) no sistema, gerando um loop. Basicamente, se o seu feedback for 0, o delay irá reproduzir o som gravado apenas uma vez. Quanto mais alto o feedback, mais vezes o som é reproduzido. Em geral, se o seu feedback for superior a 52% o som irá se reproduzir infinitamente. Com porcentagens menores o efeito tem a amplitude (volume) reduzida a cada vez que é reintroduzida no sistema, efetivamente gerando um som finito.

Color: Esse knob regula a “cor” do som processado pelo efeito, isto é, sua característica timbral. A definição de color é um tanto complexa e para não complicar essa descrição tentarei ser breve. Pode ser regulado de -50 até +50. Basicamente quanto menor o valor mais “robusto” e “encorpado” o timbre e quanto maior o valor, mais “afiado” e “rasgado”.

Mix: Esse parâmetro determina o quanto do sinal original e do sinal processado sairão do efeito. Ao ser posicionado em 0, tanto o som processado (wet) quanto o som limpo (dry) são ouvidos em intensidade semelhante. Ao mover o knob para o lado positivo (direita) o som processado se sobressai sobre o som limpo e vice-versa. Serve para misturar um som no outro para que se possa encontrar o balanço perfeito.

Ping-Pong: Esse botão ativa a função ping-pong do delay. Essa função faz com que cada vez que o som for reproduzido, o sinal será redirecionado para um dos lados do canal estéreo. Por exemplo, ao tocar uma nota de piano e então passá-la por um ping-pong-delay, a primeira vez que ela for reproduzida será na caixa de som/fone esquerda, a segunda na caixa de som direita, a terceira na esquerda e assim sucessivamente. Esse ato de jogar de um lado para o outro que dá o nome desse parâmetro de ping-pong.

I/O: botão para ligar e desligar o módulo para possibilitar melhora no desempenho do computador caso aquele efeito não esteja sendo utilizado naquele momento.

3.1.3- Reverb

Reverberação é a persistência do som após ele ser produzido. Uma reverberação é criada quando um som emitido é refletido em diversas superfícies de maneira sucessiva até que seja totalmente absorvido. É o clássico exemplo de bater palmas num banheiro, ocorre uma rápida reverberação, como um eco extremamente rápido. Cada ambiente possui uma reverberação diferente. A reverberações dependem de uma série de fatores, desde material das superfícies, frequências predominantes do som emitido, até o posicionamento da fonte sonora em relação a tais superfícies. Na produção musical um reverb pode ser utilizado para dar uma “sala” a um instrumento, dar corpo a um timbre, criar texturas e atmosferas ou até mesmo agir de modo semelhante a um delay dependendo de como for programado. Por ser tão versátil assim, tive de incluir este efeito em meu projeto. Os reguladores:

Size: o tamanho da “sala” simulada pelo efeito. Salas pequenas geram poucas reflexões e salas grandes, como salões de dança, podem gerar grandes reverberações.

Decay: é o regulador que controla o tempo de duração da reverberação. Basicamente é o controle sobre quanto tempo as reverberações demoram para serem absorvidas pelas superfícies.

Low e high cut: como explicado no módulo de input/output, servem para recortar bandas indesejadas, só que, diferentemente do input/output, esses knobs controlam especificamente a reverberação em si e não o som que as gera. Quando se reflete um som podem ocorrer sobreposições auditivas que por sua vez podem gerar artefatos ou cancelamento/construção de fase, o que pode gerar uma certa timbragem indesejada no reverb.

Mix, Color e I/O: função semelhante à descrita no módulo Delay.

3.1.4- Saturator

Um saturator é uma ferramenta para acrescentar distorção a um sinal de áudio. Diferentes algoritmos que simulam diferentes modos de distorção podem ser selecionados para cada necessidade do engenheiro. Ferramenta essencial para a produção musical, um saturator é quase obrigatório em um rack de efeitos. A quantidade de distorção pode ser controlada para que se obtenha o resultado desejado. Os reguladores:

Amount: Quantidade de distorção aplicada ao sinal. Determinada em porcentagem, 0% sendo nada de distorção e 100% um sinal completamente distorcido.

Seletor de distorção: Um menu drop-down de seleção do algoritmo de distorção a ser utilizado. Neste caso, o algoritmo selecionado é o de Overdrive.

Clip: embora pareça com um botão, na verdade este seria apenas um indicador de que o sinal de áudio está ultrapassando o limite físico do software. Quando este limite é ultrapassado, a luz se acende, indicando ao engenheiro que ele deve consertar isto de alguma forma, seja reduzindo o Amount, seja reduzindo o ganho.

Gain: possui função semelhante ao gain do input/output, ou seja, de regular a amplitude do sinal. Coloquei este regulador aqui pois o ato de distorcer um sinal acrescenta muita energia a sua amplitude, fazendo-se necessária uma correção da amplitude do sinal para que não haja uma quebra do limite físico do módulo. Se a luz Clip acender, é sinal de que o ganho deve ser abaixado caso queira-se manter aquela quantidade de sinal distorcido.

Mix, Color e I/O: função semelhante à descrita no módulo Delay.

3.1.5 - Filter

Um filtro é uma ferramenta fundamental para se processar áudio. Eventualmente um produtor pode querer remover uma faixa do espectro de frequências, seja para corrigir alguma falha, seja para dar uma característica diferente ao timbre. Reguladores:

Cutoff: O ponto a partir do qual o filtro age no espectro de frequências.

Resonance: É um aumento na intensidade do som logo após o ponto do cutoff. Quanto maior a ressonância, maior é o aumento dado à esse ponto.

Seletor de filtro: Como dito na sessão de terminologia, existem vários tipos de filtro diferentes, esse seletor é para que o usuário possa escolher o tipo de filtro que melhor vai poder auxiliá-lo em sua necessidade.

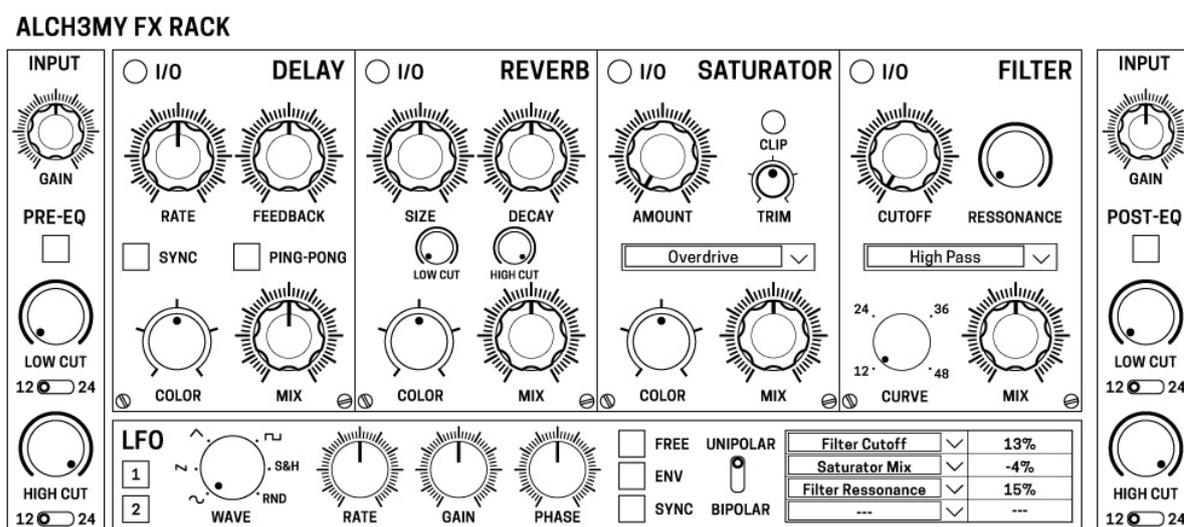
Seletor de curva: Para o usuário poder escolher a curva em decibéis do filtro que está utilizando. Semelhante ao seletor de curvas do input/output, mas com mais opções de curvas para uma maior variedade de timbragens.

Mix, Color e I/O: função semelhante à descrita no módulo Delay.

3.2- Design Gráfico

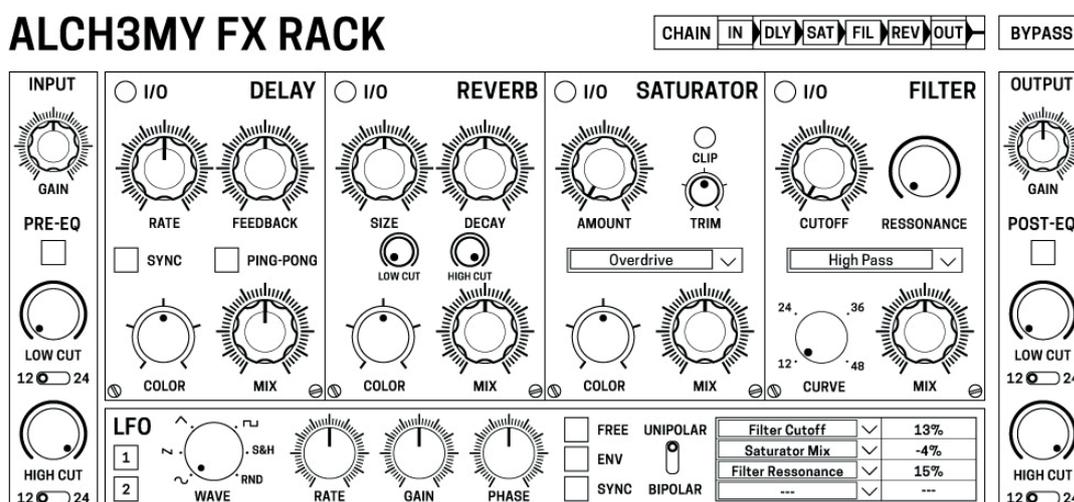
Como o projeto foi conceituado como um rack, pensei em dar-lhe uma aparência de objeto real a partir do rendering O plugin possui muita informação e devido a isso, a tipografia deve ser sólida e de boa legibilidade.

A fonte escolhida, para este projeto, foi a Copper Hewitt por sua definição e aspecto estético dos tipos em caixa alta, o que ocorre na maioria dos textos aplicados na interface.



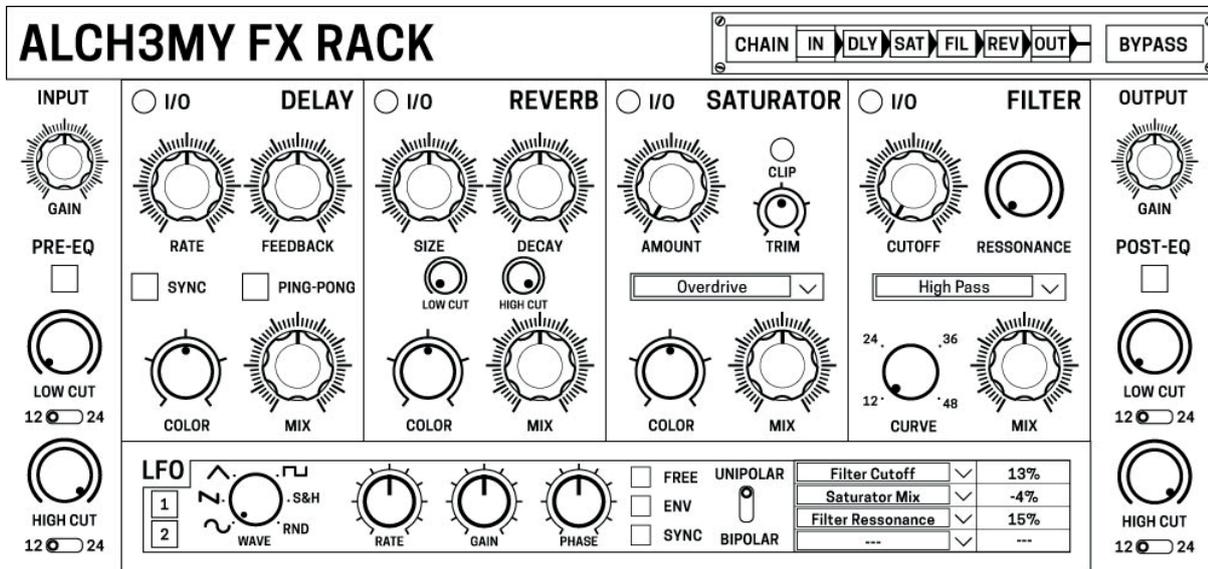
A ilustração acima reproduz a primeira versão de interface cujo nome pertence a uma futura gravadora e estúdio que pretendo abrir com um sócio. Observe, também, que há um sétimo módulo nesta versão, o módulo LFO, mas ele foi introduzido num período avançado do desenvolvimento do projeto. Falarei mais adiante sobre este módulo.

Nesta versão os Knobs já possuem formas e a fonte escolhida já mostra que serve o propósito.



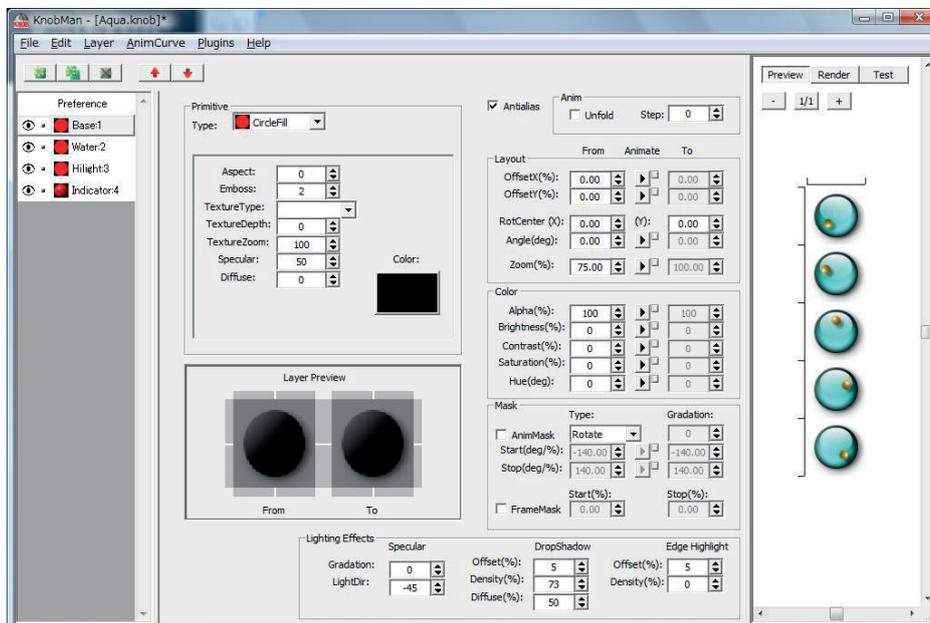
Em novo layout (figura anterior) foram alterados o aspecto visual de alguns botões, a interface do LFO, além do acréscimo no topo direito de um ordenador de processamento, isto é, um seletor da ordem pela qual o sinal passa e é processado.

Na versão final do wireframe (Figura abaixo) removeu-se os parafusos que estavam na interface. A intenção inicial era de dar um ar de "real" mas concluí que eram poluentes visuais e atrapalhavam o layout em geral. Deixei alguns, no entanto, no ordenador de processamento. O LFO foi alterado para que o seu design se encaixasse melhor com possível versão "real" do produto.



3.2.1- O processo de rendering

Iniciado o processo de rendering, logo me deparei com um problema ao desenvolver os knobs. O knob necessita de ser animado girando, o que complica quando se leva em conta que, para parecer real, a iluminação tem que ser fixa enquanto o objeto se move. Para a resolução dos reguladores são necessários 101 frames para alguns tipos e 127 para outros. Isso quer dizer que fazer manualmente cada peça com a iluminação correta é inviável. Felizmente, pude encontrar um software dedicado exclusivamente para esse fim. O software KnobMan, do desenvolvedor g200kg possui tudo que é necessário para animar e gerar o arquivo final que será lido pelo programa de desenvolvimento da GUI. Irei me aprofundar a respeito do software na parte que se refere ao processo de desenvolvimento da parte gráfica do projeto.



A GUI, em sua forma final deverá ser composta por várias camadas diferentes de elementos:

- Textura
- Fundo
- Serigrafia
- Luz
- Sombra
- Camada de elementos animados

Abaixo a camada de fundo, textura, luz e sombra desenvolvido :



A versão final, com os botões e reguladores, ainda não possui nenhuma animação. Essa etapa acabou sendo deixada para desenvolvimento posterior e não foi executada devido ao abandono dessa frente de projeto.

3.3- Conclusão do projeto Rack de efeitos

A pesquisa que desenvolvida no processo de elaborar o rack de efeitos foi de suma importância para que se encontra-se alternativas para a realização do projeto. A parte gráfica precisa evoluir, principalmente na parte de animação, realizada de maneira insatisfatória pela minha falta de conhecimento mais aprofundado. .

Percebi ao longo dos meses que o produto provavelmente seria pouco utilizado por uma série de razões. O plugin de áudio deixou muito a desejar no conceito. O grosso do trabalho foi feito, mas a dificuldade em materializar o produto, deixou a impressão de que eu jamais conseguiria ultrapassar a qualidade de plugins que já existem no mercado fizeram com que eu abandonasse a ideia do Rack de efeitos no início de 2018.

4 - O desenvolvimento do sintetizador

Com isso, mudei quase que completamente o escopo do projeto no início de 2018. De um plugin de áudio de efeitos passei ao projeto de um sintetizador de baixo para produtores de psytrance. Esse nicho tão específico pode ser melhor explicado levando em conta meu histórico como profissional nesta “cena”, e por isso falarei brevemente sobre o assunto ao especificar mais sobre a definição de escopo e produto.

4.1 - A definição do produto

Sou produtor de festas, DJ e produtor de psytrance há 7 anos. Em todos esses anos conheci um número enorme de produtores e músicos da área, pude presenciar diversos produtores de diversas nacionalidades produzindo suas músicas quando os trazia para algum evento que eu participava, por exemplo. Sempre tive um grupo de amigos que fazem música e trocamos conhecimento, vamos nos estúdios uns dos outros, desenvolvemos novas técnicas de síntese e mixagem e fazemos música juntos.

Para pessoas que não são acostumadas com o gênero musical, Psytrance pode soar muito repetitivo e é difícil para perceber detalhes na mixagem e produção das músicas. No entanto, psytrance é um gênero que possui uma variedade incalculável de sub-gêneros, cada qual com suas características timbrais específicas que por si requerem técnicas específicas para mixagem e melhor qualidade sonora. O processo de síntese sonora de um baixo de psytrance consiste basicamente em passar uma onda sawtooth por um filtro em modo low-pass com um rápido envelope, ajustado de acordo com o gosto do produtor.

Abaixo, explico esse processo através de imagens de um osciloscópio acompanhadas de áudio. Para a visualização do processo, utilizarei um osciloscópio.

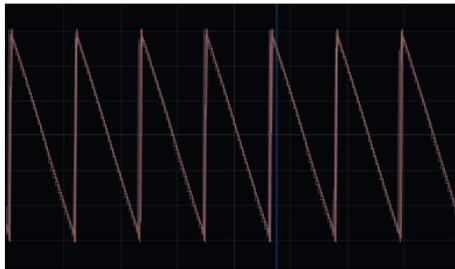
Passo 01:

Selecionar uma onda sawtooth em seu oscilador (OSC A).

Utilizarei o sintetizador desenvolvido neste projeto como exemplo.

Onda sawtooth, tocada na nota C2, sem filtro:

Arquivo de áudio: 01



Passo 02:

Aplicar o filtro com a curva desejada.

Neste caso, utilizaremos a curva de 24dB no menu localizado na seção do filtro (FILTER).

Mesma onda, tocada na nota C2, com o filtro aplicado:

Arquivo de áudio: 02



Passo 03:

Aplicar o envelope no cutoff do filtro. Neste sintetizador isso é feito através do knob chamado Envelope Amount, (ENV AMT.) e regular os parâmetros de attack, decay, sustain e release do envelope do filtro (FIL ENV) de acordo com a timbragem desejada.

Arquivo de áudio: 03 (a nota é pressionada 4 vezes)



Passo 04:

É feito o sequenciamento e arranjo das notas, juntas ao Kickdrum, peça fundamental na construção da linha de baixo. Após isso, se obtém uma linha de baixo básica, que pode ser trabalhada e processada para que se obtenha a característica sonora desejada.

Arquivo de áudio: 04 (o baixo sequenciado é repetido 4 vezes, seguido do mesmo baixo acompanhado de kick).

Baixo sem o kick:



Baixo acompanhado de kick.

Observe que, nos pontos onde o kick encontra o baixo, ocorre uma alteração no sinal de áudio. Isso se deve devido a um fenômeno chamado cancelamento de fase. Isso ocorre devido às frequências graves do kick e do baixo estarem desalinhadas. Para que isso seja corrigido, é necessário ter controle sobre a fase do oscilador que cria a nota do baixo. Isso foi um dos fatores determinantes para uma série de ferramentas que acrescentei ao sintetizador no final: os knobs de controle de fase.



Essa é a clássica “rolling bassline” de psytrance, e pode ser escutada em várias músicas do gênero, como por exemplo Ajja - Bazinga, Ace Ventura - Dark Matter.

O sintetizador criado tem como objetivo ser uma ferramenta que possibilite ao usuário realizar todo esse processo descrito acima com facilidade e rapidez. Quanto menos CPU consumisse o sintetizador, melhor, pois significa que o usuário pode dedicar a sua CPU a outros sintetizadores mais intensos em seus projetos musicais.

4.2 - A conceituação

Após definir o que eu tinha como escopo para meu instrumento, parti para a fase de determinar como seria de fato o sintetizador, ou seja, quais parâmetros de controle iria incluir, quantos osciladores, quantos envelopes, qual tipo de envelope, quais tipos de onda, quais tipos de filtro.

A grande justificativa para as escolhas apresentadas no sintetizador foram por necessidades e dificuldades que encontrei em meus experimentos e experiências ao longo desses anos como produtor de psytrance. Acessei inúmeros tutoriais tanto de cursos pagos online de síntese sonora quanto de youtube, conversei com vários artistas estabelecidos na cena a respeito do processo de síntese de suas linhas de baixo.

Abaixo o produto final com todos os parâmetros e a GUI aplicada:



Irei separar o sintetizador em suas diferentes partes e justificarei as minhas escolhas para cada parâmetro. Basicamente, o sintetizador pode ser dividido em 5 partes:

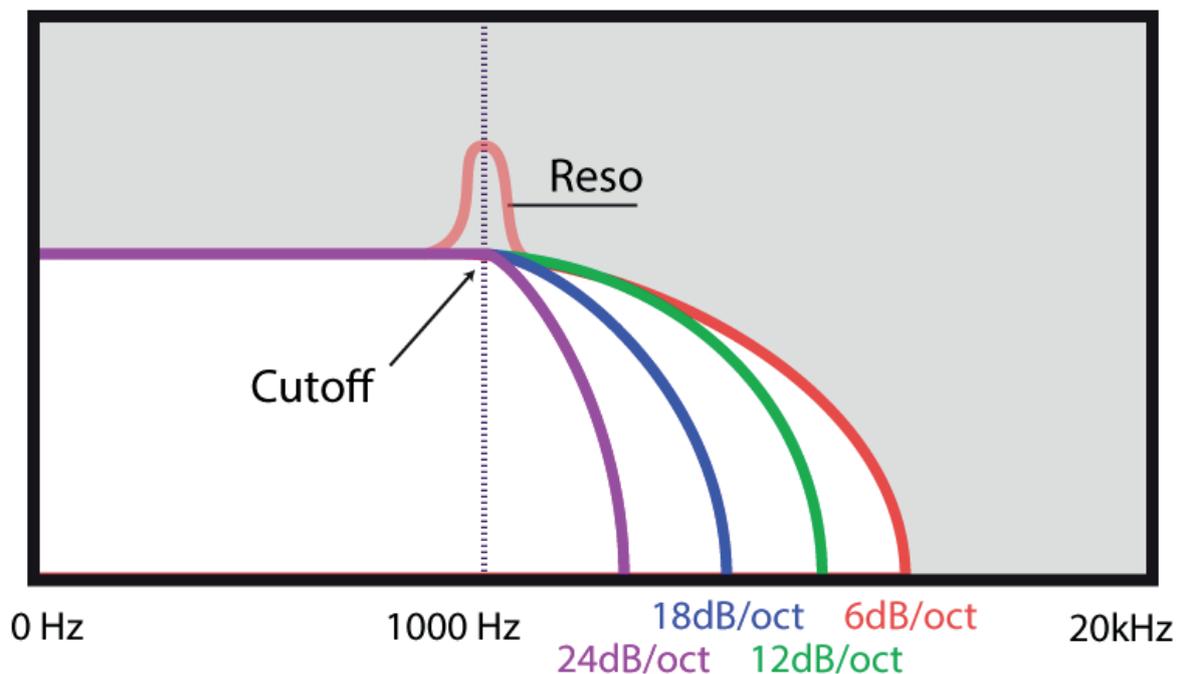
Envelopes: a seção que possui 2 envelopes ADSR, um para o controle do volume do timbre ao longo do tempo (AMP ENV), e o outro para o controle do cutoff do filtro ao longo do tempo (FIL ENV). Cada envelope possui seus respectivos controladores para os parâmetros (attack - ATT, decay - DEC, etc) e um display gráfico do comportamento daquele envelope, que se molda à medida que os parâmetros são alterados

Osciladores: o sintetizador possui 2 osciladores. Cada um deles pode oscilar uma dentre 4 ondas sonoras diferentes (sine, sawtooth, triangle e pulse). Cada oscilador possui um menu para a seleção da onda desejada (WAVE), um botão para ativar o Oscillator Resync (SYNC) um controlador para a fase inicial da onda (PHASE) e 3 controladores para o Tom, cada um com uma precisão diferente. TUNE altera o tom do timbre de

maneira mais abrangente, SEMI altera de semitom em semitom e FINE faz o ajuste microtonal do timbre. Observe que entre os dois osciladores existe um controlador denominado PHASE 1+2. Esse controlador altera a fase dos dois osciladores ao mesmo tempo, o que permite que o usuário possa fazer o controle de fase em cada um dos osciladores separadamente através de seus respectivos knobs e depois um terceiro ajuste de fase que altera os dois osciladores de maneira sincronizada. Alterar a fase de um oscilador é essencial para linhas de baixo de psytrance pois o alinhamento de fase entre kickdrum e baixo é essencial para que se soe bem em sistemas de som de grande porte.

Filtro: como este é um sintetizador de baixo, utilizei apenas curvas de Low-Pass. Para aumentar a variedade timbral e tonal, coloquei 5 curvas diferentes de Low-Pass: 6dB/oitava, 12dB/oitava, 18dB/oitava, 24dB/oitava e o Low-Pass Pro-52 (curva de 24dB/oitava utilizada no sintetizador virtual Pro-52). O CUTOFF determina o ponto de corte no espectro de frequência (de 0Hz a 20kHz) em que o filtro irá cortar as frequências agudas (120=20kHz). RESO controla o valor do ganho dado ao ponto em que ocorre o corte da frequência. ENV AMT. controla a intensidade da modulação exercida pelo envelope no cutoff do filtro.

Observe na imagem abaixo, uma representação do funcionamento do filtro. O cutoff do filtro abaixo se encontra em 1000Hz, e as diferentes curvas e o tanto de som que elas deixam passar são representados nas diferentes cores. A ressonância, controlada pelo knob RESO, é um ganho dado logo no ponto de cutoff, se estiver em 0, não existe ganho.



Osciloscópio (SCOPE): Ferramenta que mostra o sinal de áudio que é emitido pelo sintetizador de maneira gráfica. Utilizado como ferramenta para auxiliar no monitoramento do sinal. O controlador dessa seção determina o tamanho da onda mostrada no display.

Mixer/output: é a seção do sintetizador que controla o volume de saída (OUTPUT) e qual dos osciladores está sendo usado (MIXER). O fader de mixer pode ser posicionado arbitrariamente entre os dois extremos, fazendo assim uma mescla do som emitido pelos 2 osciladores. O fader de volume permite que o usuário possa regular a altura do som emitido. O controle vai de -60dB a 0dB, sendo 0dB o volume máximo de saída.

4.3 - O desenvolvimento

Para elaborar o sintetizador, me utilizei, como previamente citado, da plataforma Reaktor, da Native Instruments. Reaktor é um estúdio musical modular gráfico. Permite que usuários possam criar e desenvolver seus próprios instrumentos e ferramentas de design sonoro. Possui uma série de efeitos e instrumentos já prontos que acompanham o software, de sintetizadores clássicos até ferramentas de áudio futuristas. Além disso, possui mais de 3000 instrumentos e efeitos criados por usuários em sua biblioteca que é constantemente atualizada e retroalimentada. Um dos pontos principais do Reaktor é que qualquer instrumento que seja criado na ferramenta pode ser examinado através de engenharia reversa. De fato, a ferramenta encoraja essa atitude. Usuários aprendem com os sistemas desenvolvidos por outros e assim a comunidade se expande. O Reaktor Player é uma ferramenta gratuita que permite que qualquer usuário possa utilizar os instrumentos desenvolvidos pela ferramenta, mas não permite a edição ou engenharia reversa dos instrumentos.

O programa possui uma interface gráfica default dada aos parâmetros do instrumento que um usuário cria, no entanto, também permite ao usuário edição, ainda que com algumas limitações da parte gráfica. O processo de desenvolvimento do instrumento dentro da plataforma Reaktor será coberto parcialmente neste relatório, pois muitas coisas fogem do escopo desse projeto e se encaixam mais em fatores técnicos de engenharia de som. Tudo aquilo que for relevante será descrito.

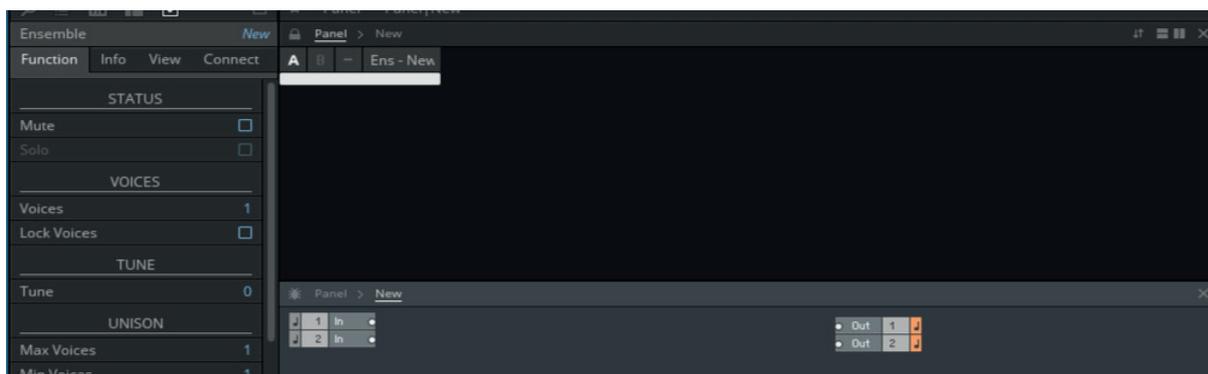
4.3.1 - Desenvolvendo no Reaktor

Reaktor permite ao usuário criar uma sequência de módulos que podem ser interconectados e transmitir sinais entre si, como em um circuito elétrico. Esses módulos podem ser colocados dentro de macros, que são basicamente “caixas” virtuais para melhor organização de seu circuito. Para este sintetizador, utilizei-me de diversos módulos nativos do Reaktor. Observe a interface do programa na imagem a seguir.

Na esquerda, existe uma barra lateral com 4 abas diferentes: Function, Info, View, Connect. Na janela principal, no meio do programa, é onde desenvolvemos o sintetizador e conectamos os módulos desejados. A barra superior mostra a visualização do painel do instrumento desenvolvido e a barra inferior mostra o “circuito” virtual do projeto, com todos os seus módulos conectados. No centro estão as duas abas onde o projeto é desenvolvido de fato. A barra superior mostra o painel, ou interface gráfica, do instrumento que está sendo criado, e a barra inferior o seu respectivo circuito, ou o “motor” do instrumento.

Podemos criar objetos a partir de uma enorme pré seleção de módulos já existentes no programa ou podemos, também, utilizar módulos criados por usuários. Para este projeto, utilizamos apenas os módulos nativos do programa.

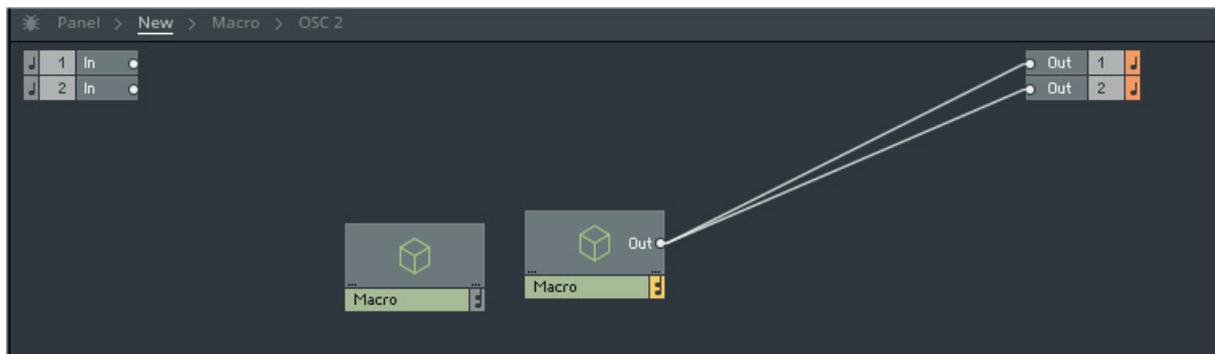
O Reaktor trabalha basicamente com 2 tipos de “sinal”. O sinal elétrico, que seria o equivalente ao sinal elétrico que gera o som que escutamos na saída de áudio, representado pelas conexões brancas (“fios” brancos). E temos o sinal de controle, ou sinal lógico, que seria o sinal elétrico responsável pelas modulações em cima do sinal de áudio; neste caso é representado pelas conexões amarelas. As entradas brancas recebem tanto sinal elétrico branco quanto o sinal lógico. As entradas amarelas recebem apenas sinal lógico. Observe melhor nas imagens a seguir.



Abaixo, a visualização do sintetizador criado para este projeto, em uma de suas primeiras versões. No meio, na parte de cima, é visível o painel do sintetizador. A GUI dessa versão do projeto são os elementos padrão do programa. Veja que as conexões que se ligam às saídas de áudio são fios brancos, isso é, sinal de áudio.



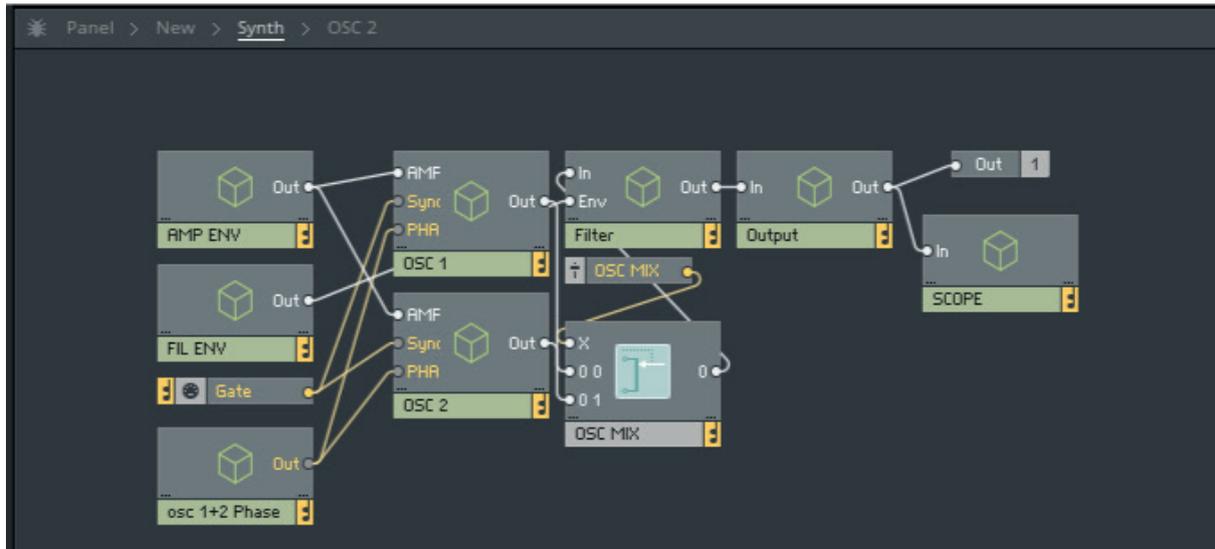
Na barra inferior, é possível ver uma caixa chamada “Macro” que está conectada a 2 módulos de saída, chamados Out. Observe melhor na imagem a seguir.



Os módulos de saída de som são disponibilizados por default pelo programa. Se o instrumento não estiver conectado a eles, é impossível obter um sinal de áudio.

Como mencionado anteriormente, os “macros” são caixas que contém elementos de construção do sintetizador. Utilizamos macros no Reaktor para organizar o circuito. Podemos clicar duas vezes em um Macro para acessarmos seu conteúdo e consequentemente conseguirmos editar tal conteúdo.

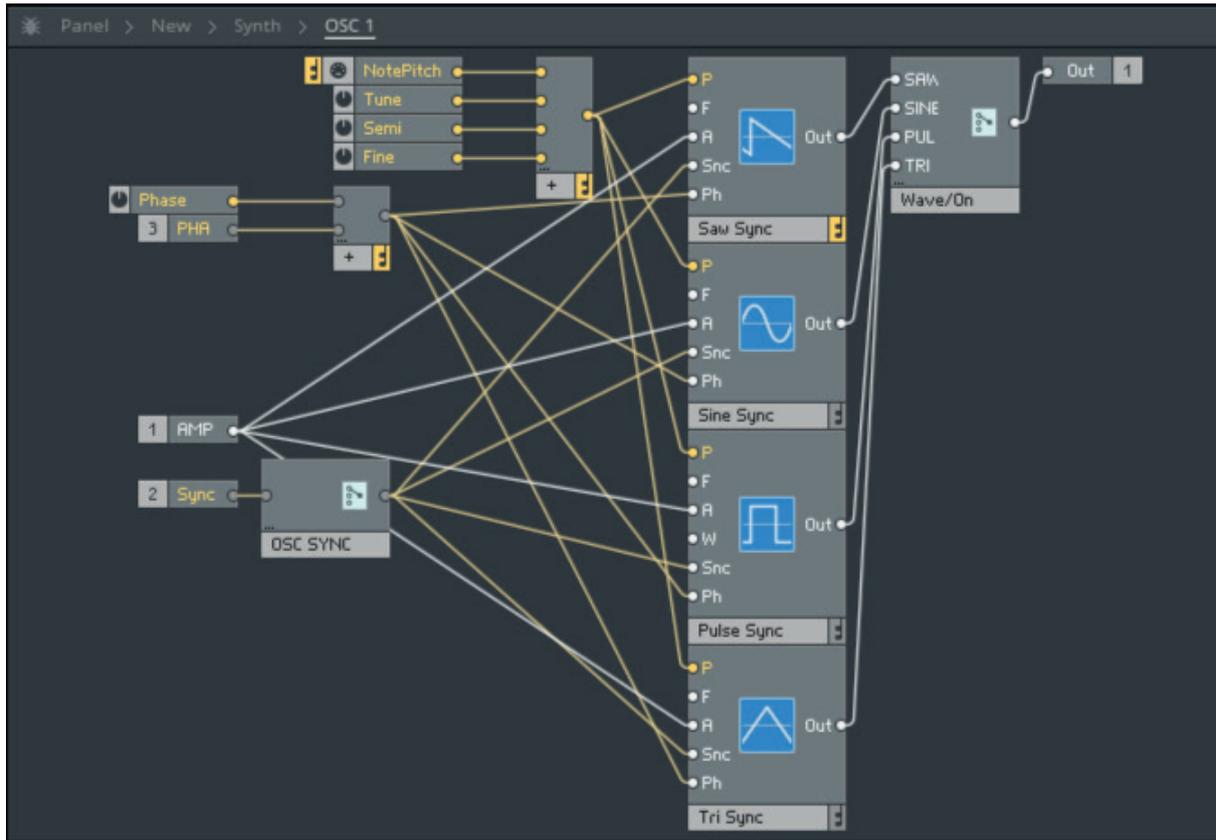
Ao se clicar duas vezes no objeto "Macro" que está conectado às duas saídas, podemos acessá-lo e ver o que o compõe.



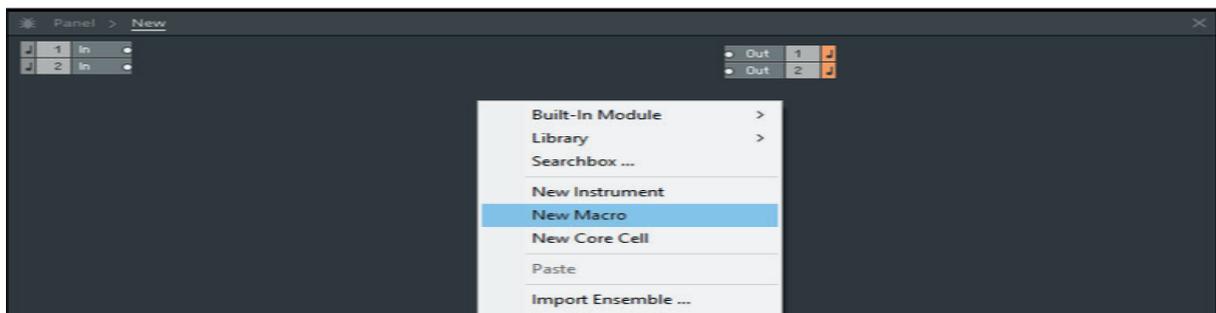
Dentro daquele macro existem outros 8, cada um interconectado a outros macros. No Reaktor, a caixa que possui a barra inferior verde (onde está o nome) é um macro e a caixa que possui a barra cinza é um módulo. Podemos ver que os macros são denominados de acordo com a função que cada um exerce no sintetizador. Observe, também, que agora existem 2 tipos de conexões, as brancas e as amarelas. As conexões amarelas são os sinais de controle, ou seja, sinais que controlam parâmetros dentro dos módulos, por exemplo knobs, faders, botões.

No nosso caso, possuímos 8 Macros: AMP ENV, FIL ENV, OSC 1+2 PHASE, OSC 1, OSC 2, Filter, Output e SCOPE. Existem 3 módulos que estão fora de macros nesse caso: Gate, OSC MIX e Out. O módulo de Out é o de saída; assim como no nível anterior, necessitamos criar uma saída dentro deste macro para que possamos ligar o sistema às saídas padrão de áudio. O módulo Gate é um módulo que recebe os inputs MIDI de seu sistema e os traduz para sinal elétrico, neste caso um sinal de controle, como pode ser observado pelo fio amarelo. Ou seja, quando eu pressiono uma tecla em meu teclado MIDI, o módulo Gate irá converter aquele sinal para um sinal elétrico, que então irá seguir seu rumo através do circuito. O módulo OSC Mix é um módulo que recebe 2 ou mais sinais de entrada mas possui apenas uma saída. No caso deste sintetizador, conectamos um oscilador em cada entrada e utilizamos o controlador X para determinar qual sinal será ouvido na saída. Observe que o módulo possui duas entradas: 00 e 01 que são as entradas de sinal de áudio e a entrada X é uma entrada que recebe sinal de controle. No caso deste sintetizador, criamos um elemento de Fader cujo sinal lógico vai de 0 a 1. Quando o fader está em 0, apenas o sinal ligado à entrada 0 0 será ouvido na saída. Se o fader estiver em 1, apenas o sinal da entrada 0 1 será ouvido. Se o fader se encontra em 0,5, cada sinal poderá ser escutado com o volume igual.

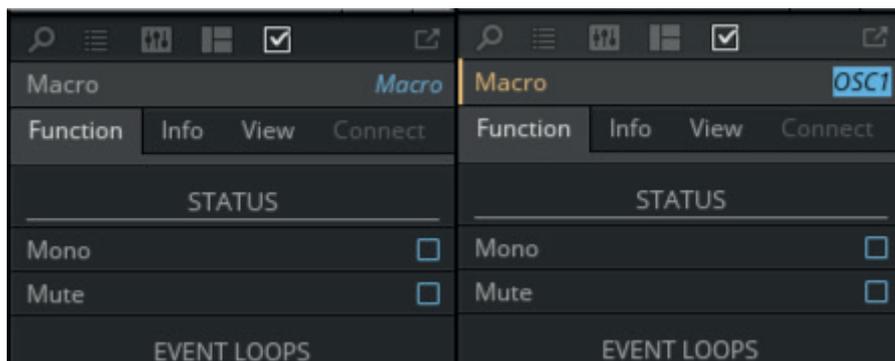
Vamos acessar o macro denominado OSC 1 e ver o que o compõe. Observe a imagem à seguir.



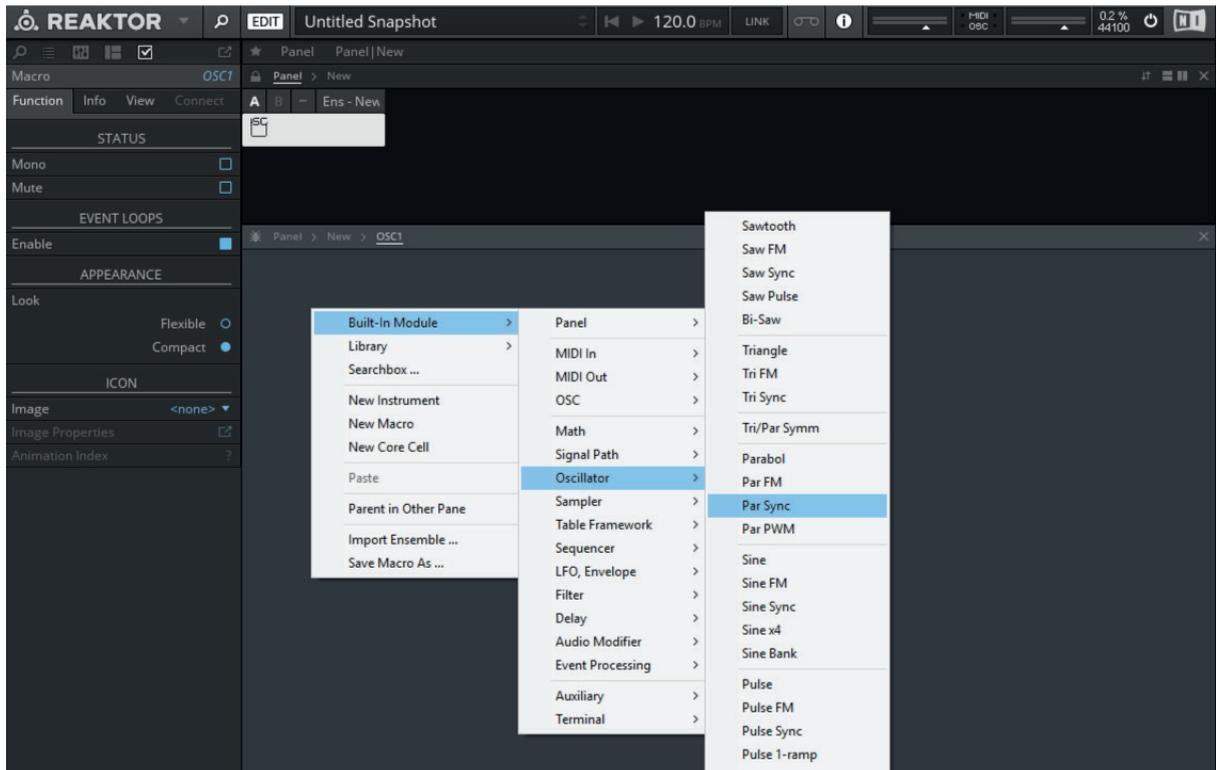
Como exemplo de desenvolvimento do projeto, irei demonstrar como foi gerado o macro do oscilador 1 (OSC 1) através do Reaktor. Primeiramente, iremos criar um novo macro que irá conter a nossa função de Oscilador. Isso pode ser feito clicando com o botão esquerdo na barra inferior e selecionando a opção "New Macro".



Em seguida, clicamos 2 vezes na caixa que aparece e entramos em nosso macro. Podemos alterar o nome do macro na barra lateral esquerda. Como mostrado abaixo, mudamos o nome de nosso Macro para OSC1.

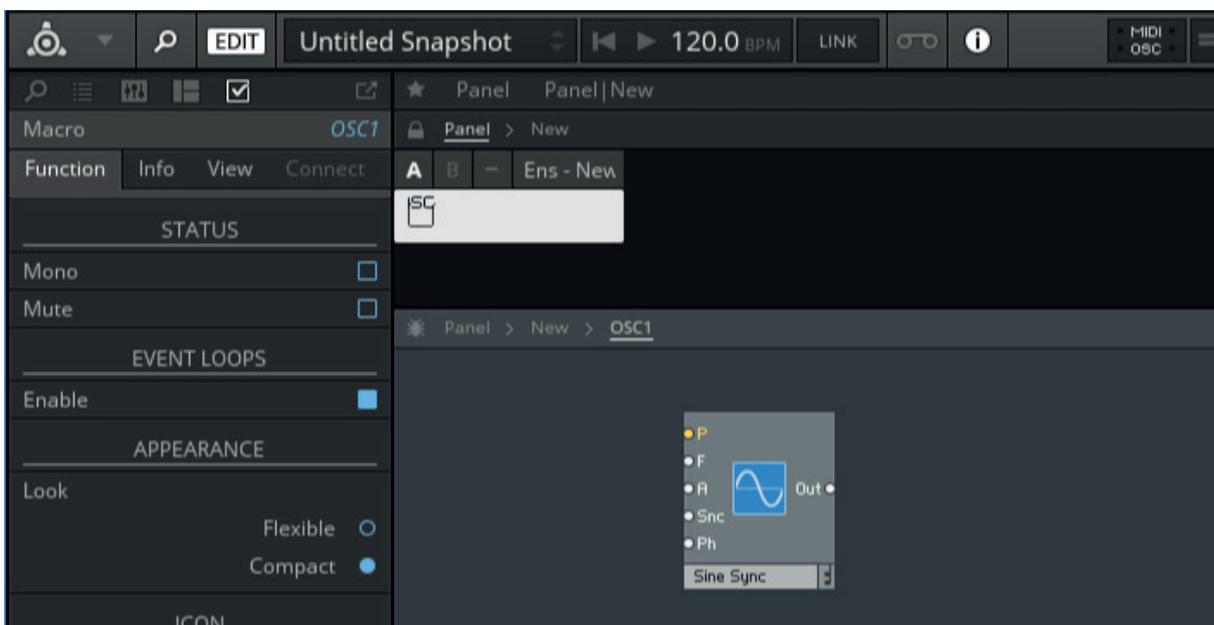


Dentro do macro agora iniciaremos a criação do oscilador. Ao clicarmos com o botão esquerdo na barra inferior, podemos selecionar a opção “Built in Module” e em seguida veremos todos os tipos de módulos nativos que o programa oferece. Selecionamos a opção para osciladores e veremos a enorme variedade de osciladores também oferecidos pelo programa.



Os mais variados tipos de módulos são necessários na construção de um instrumento ou efeito. Alguns módulos emitem sons, como os módulos de oscilador, outros executam funções matemáticas (Math) como soma, subtração. Existem módulos que exercem modulações sobre outros módulos e existem módulos que são objetos que aparecerão no painel, os módulos localizados em Panel. Esses módulos são importantes pois é a partir deles que o usuário pode interagir com o circuito criado. Os módulos Panel são os módulos de interação com o usuário, por exemplo: knobs, botões, faders.

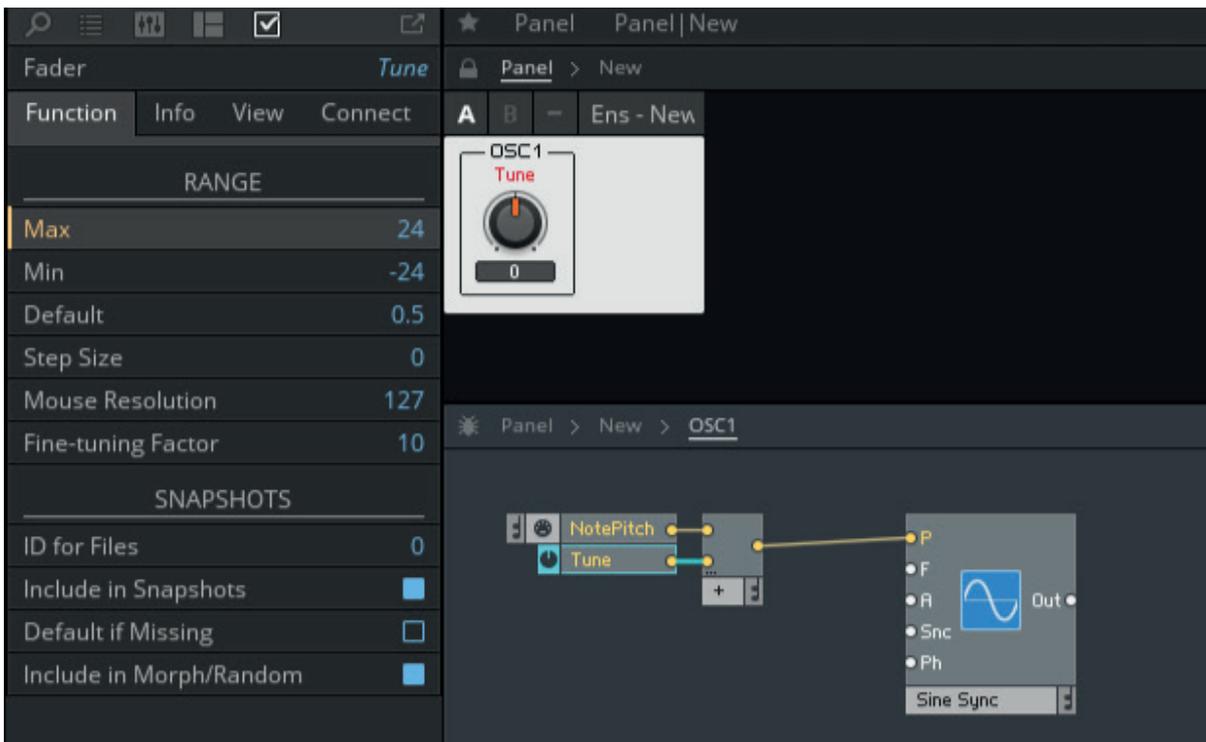
Neste primeiro momento, iremos selecionar a opção de oscilador seno. Para isso selecionaremos Built in Module>Oscillator>Sine Sync. Escolhemos o módulo de Sine Sync pois é um módulo que nos permite ter maior controle sobre o oscilador do que os outros. Veja abaixo.



Observe as entradas e saídas do módulo. O módulo possui 5 entradas: P, F, A, Snc e Ph. Observe que a entrada P é de cor amarela e as outras são de cor branca.

A entrada P controla o tom, ou o pitch do oscilador. Recebe apenas sinal de controle. Essa é a entrada que irá determinar a nota tonal emitida pelo oscilador. Podemos controlar esse sinal de várias maneiras. Nesse caso iremos utilizar a entrada midi de Pitch, que irá fazer com que possamos tocar o oscilador com nosso teclado midi. Selecionamos Built-in Module>Midi In> Note Pitch. Para que possamos afinar nosso oscilador para obter um maior controle timbral, podemos inserir um knob que irá somar o seu sinal ao sinal do Note Pitch utilizando um módulo de soma.

Criamos um módulo de Knob na aba Panel em Built in module>Panel>Knob e criamos um módulo de soma em Built-in module>Math>Add. Veja abaixo.



Observe que utilizamos o módulo de Soma para somar os dois sinais de controle. A entrada P recebe apenas 1 sinal por vez, isso quer dizer que para conectarmos 2 ou mais controles nela precisaremos utilizar um módulo de soma. O knob que criamos pode ter os valores que emite alterados na aba lateral esquerda chamada function. Podemos inserir os valores máximo e mínimo que o knob emite, no nosso caso +24 e -24, pois o módulo trabalha com valores de 1 semitom por unidade. Uma oitava possui 12 semitons, então utilizamos +24 por nos dar um alcance de até 2 oitavas acima e -24 até 2 oitavas abaixo. Veja que na parte superior é visível um knob no painel.

A entrada F não será utilizada nessa breve explicação.

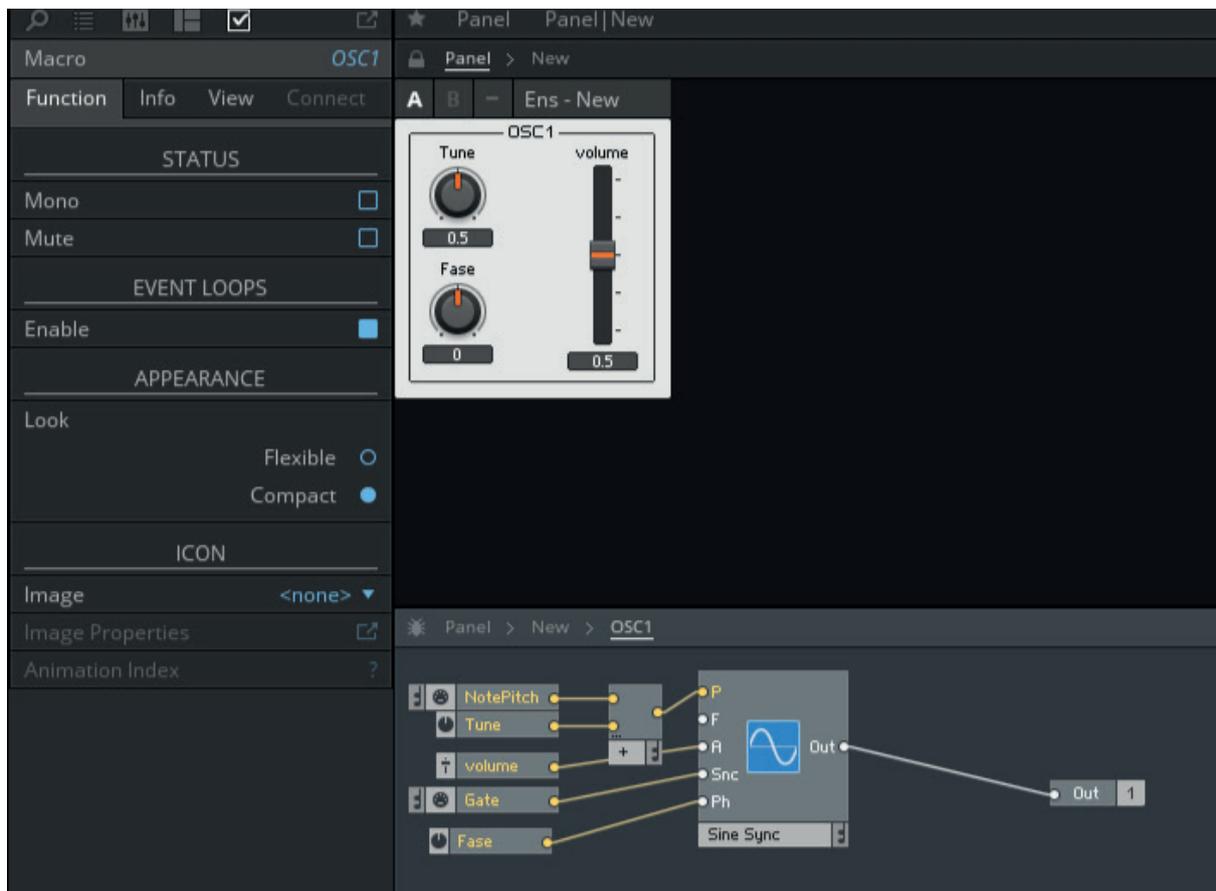
A entrada A controla a amplitude do sinal, ou seja, seu volume. Observe que é uma entrada branca, isso é pode receber tanto sinal elétrico de áudio normal quanto sinal de controle. Para esta explicação, utilizaremos um módulo de fader para controlar o volume do oscilador. Built-in Module>Panel>Fader. Na aba function podemos alterar os valores máximo e mínimo do fader para 1 e 0 respectivamente, sendo 1 o valor máximo de volume e 0 sendo sem nenhum áudio.

A entrada Snc é a entrada que recebe o sinal de sync, ou seja o pulso que reinicia o oscilador a cada vez que é emitido. Para este propósito iremos utilizar o midi gate in.

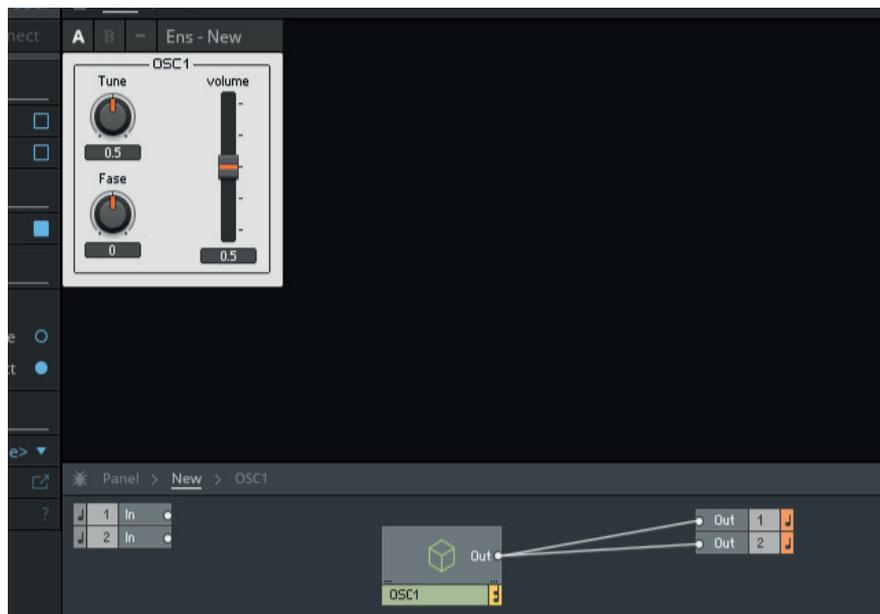
A entrada Ph é a entrada que permite o ajuste de fase do oscilador. Criaremos um knob para que possamos efetuar esse ajuste.

Out é a saída do sinal elétrico emitido pelo oscilador. Vamos conectá-lo à um módulo de saída, Out em Built-In Module>Terminal>Out Port.

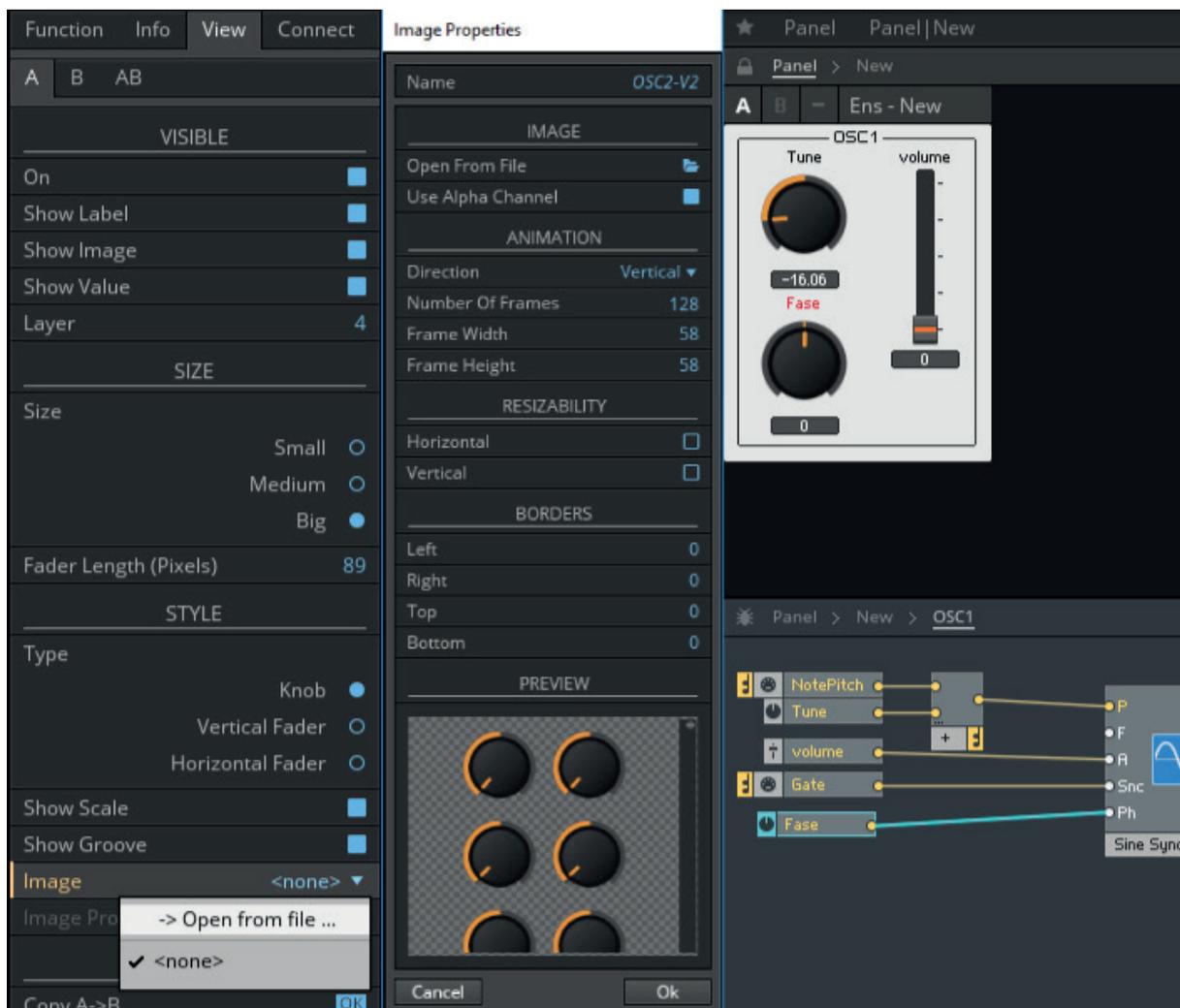
Observe como fica o pequeno sintetizador que criamos.



Possuímos um oscilador senoidal e temos controle sobre seu tom, volume e fase. Voltamos 1 nível, para o nosso macro e conectamos a saída do macro no módulos de saída do instrumento e nosso instrumento já está 100% funcional.

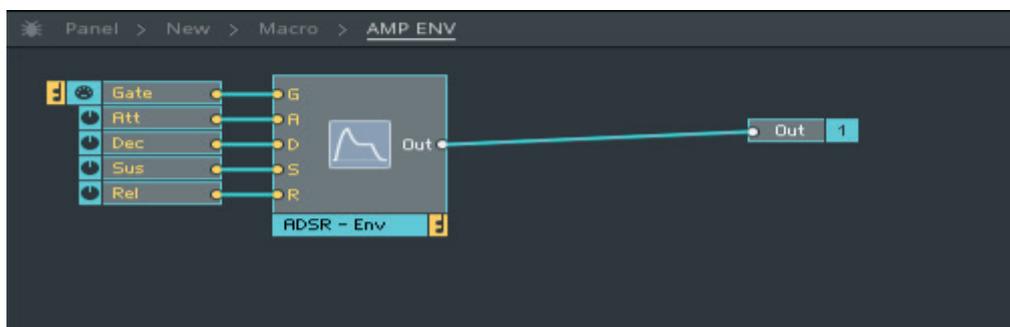


Para alterarmos os gráficos dos elementos, precisamos selecionar um módulo de painel como por exemplo um knob e selecionar o gráfico customizado que desejamos inserir através da aba View na lateral esquerda. Geramos um gráfico de knob utilizando o software Knobman. Observe a imagem a seguir e veja as etapas.



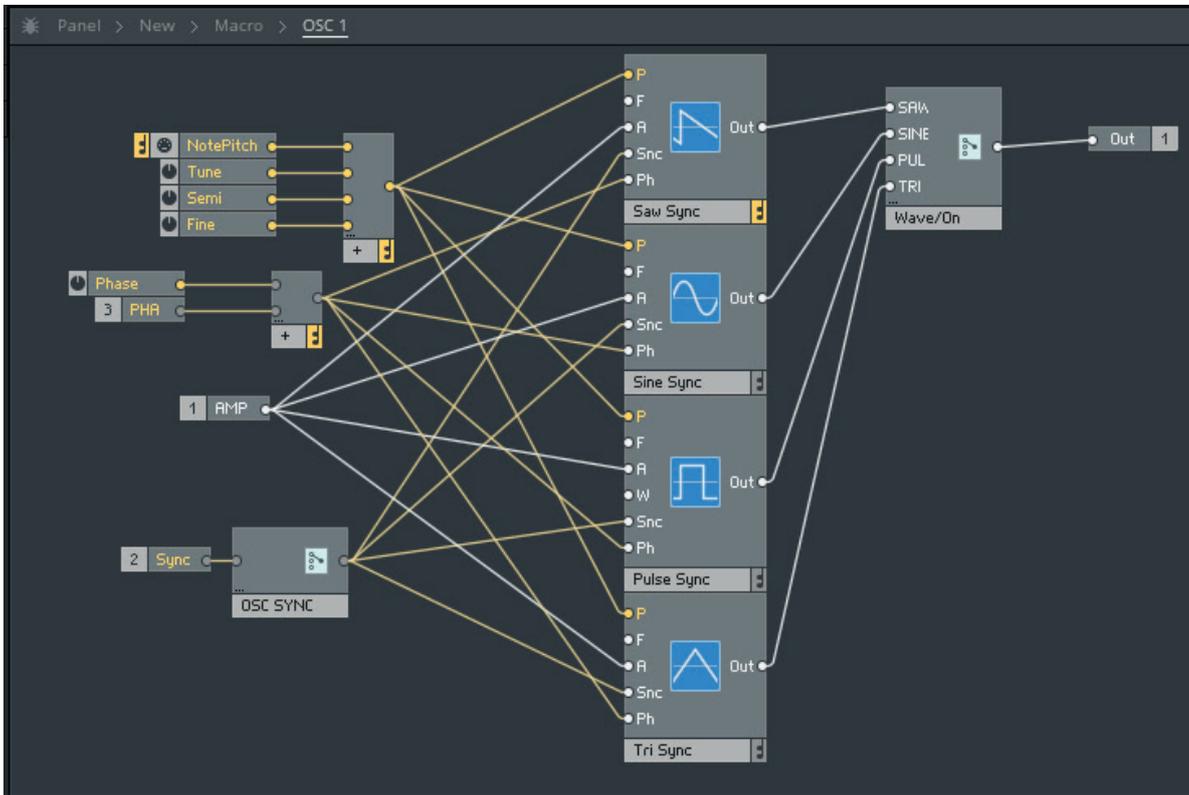
O reaktor aceita imagens animadas em formato strip, isto é, os frames da animação se encontram todos no mesmo bitmap, uma longa imagem, por isso o nome strip. Essa strip é gerada utilizando o software Knobman.

No geral, esse é o método de trabalho no Reaktor. Para explicar de maneira mais aprofundada a elaboração do circuito para o sintetizador desenvolvido neste projeto eu levaria dezenas de páginas e profundas explicações sobre teoria de engenharia de áudio. Para sintetizar irei apenas comentar sobre os macros do circuito desenvolvido.



Na imagem anterior está mostrado o circuito do AMP ENV, isto é, o envelope do amplificador. Esse macro é composto por um módulo de envelope ADSR, o gate é controlado pelo módulo de Midi Gate In e cada parâmetro do envelope é respectivamente controlado por seu knob equivalente.

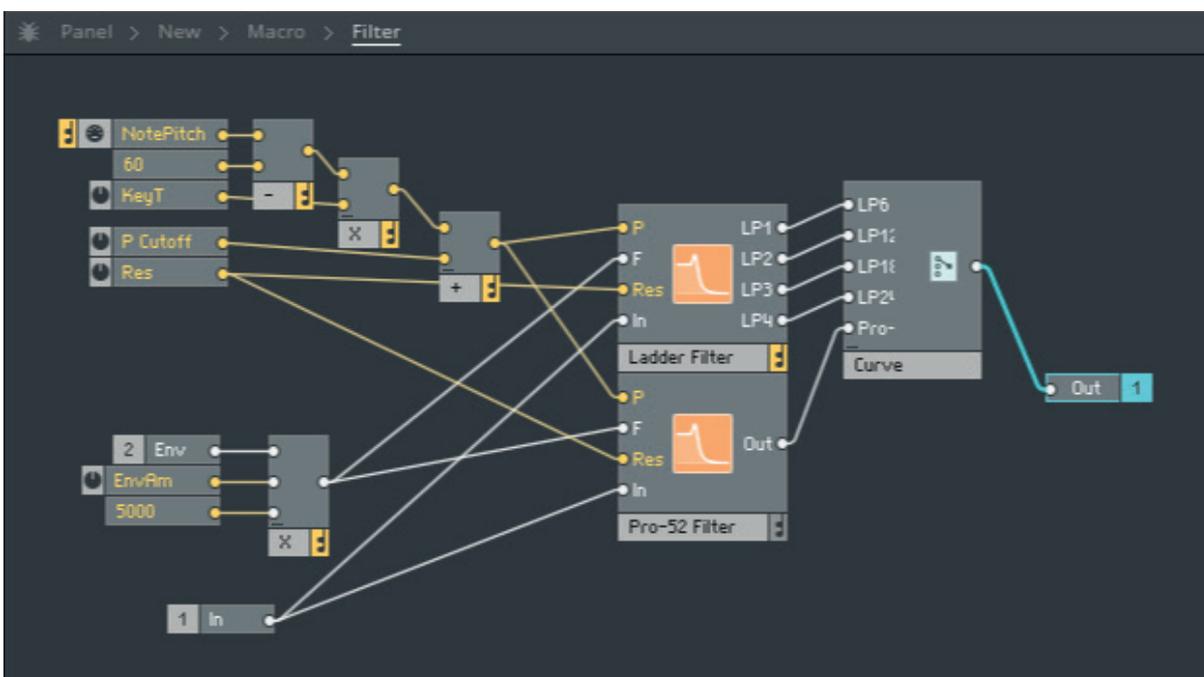
O macro do oscilador, a seguir:



O nosso macro possui 4 diferentes módulos de oscilador, um para cada onda que quisermos ter, no caso: Saw, sine, pulse e Tri. Cada oscilador possui uma série de controles de modulação e todos estão conectados a um módulo de switch, no nosso caso chamado de "Wave/On". Esse módulo permite que possamos escolher qual oscilador será utilizado pelo programa. Quando os outros não estão ativados, o consumo de CPU é reduzido.

O macro do Oscilador 2 é idêntico ao do Oscilador 1.

O macro do filtro (imagem a seguir) possui 2 módulos centrais, um Ladder Filter e um Pro-52 Filter. Observe que o módulo Ladder possui 4 saídas e o Pro-52 possui apenas uma. O ladder possui essas variadas saídas pois cada uma representa uma curva de corte por oitava diferente. Todas as saídas são conectadas a um módulo switch, o que economiza cpu ao desativar módulos não selecionados pelo usuário. Observe que neste macro existem vários módulos matemáticos para o processamento correto de sinal. Módulos de soma, multiplicação e subtração.



O desenvolvimento do instrumento na plataforma Reaktor foi um fator determinante na conclusão deste projeto. Se essa plataforma não existisse, é provável que o instrumento permanecesse no plano das ideias e o resultado deste projeto fosse apenas uma interface gráfica estática sem aplicação real.

Desenvolver nesta plataforma envolve lidar com uma série de restrições para o usuário, no entanto. Para rodar o instrumento é necessário que o usuário possua o programa devidamente instalado em seu computador. Sem o Reaktor, o instrumento não é utilizável. Durante esse semestre, enviei algumas versões demo do programa para que colegas e produtores desconhecidos pudessem testá-lo. Contactei 30 pessoas e todas já possuíam o software previamente. Junto das versões demo do sintetizador que enviei foi um questionário com uma série de coisas a respeito do sintetizador. Das 30 pessoas que enviei o questionário em Abril, nenhuma retornou o formulário preenchido, mesmo após duas sucessivas cobranças.

4.3.2 A parte gráfica

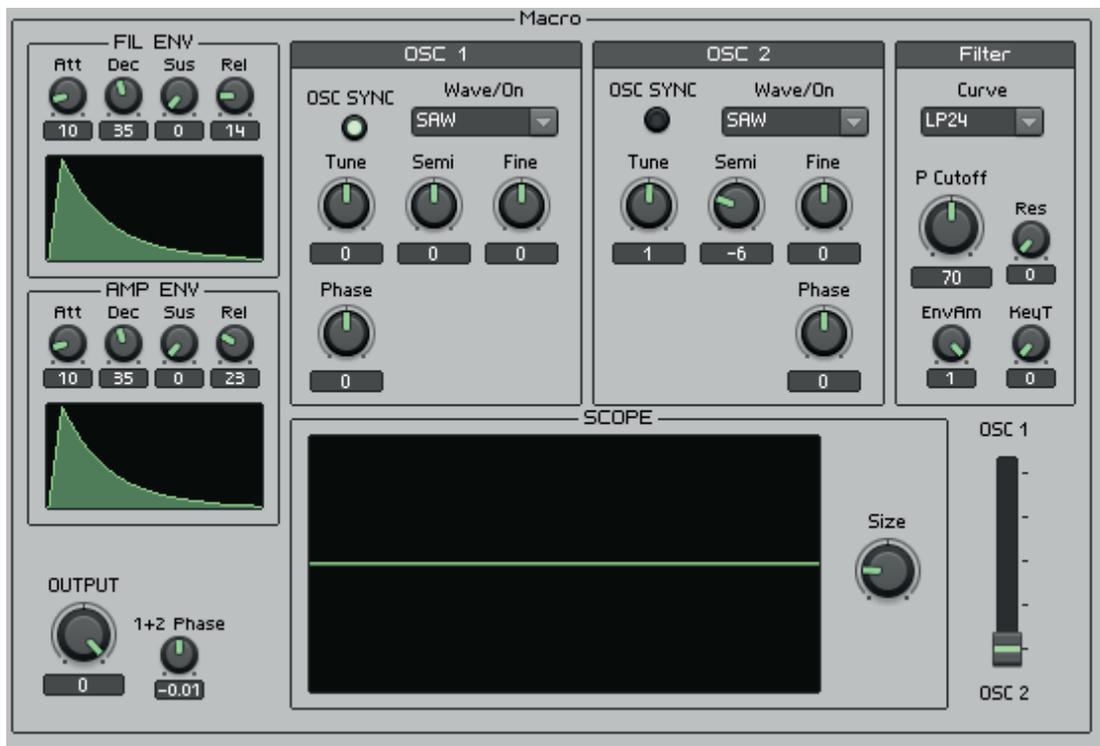
Para a parte do desenvolvimento da GUI do sintetizador, podemos separar o processo em duas etapas: O desenvolvimento do painel e o desenvolvimento dos elementos.

Como referência visual, utilizei-me de vários plugins que utilizei e ainda utilizo ao longo de todos esses anos. Observei os plugins mais populares e alguns com funções mais esdrúxulas. Montei um painel de referências que pode ser observado na página seguinte.



A disposição espacial dos elementos na interface é de suma importância num sintetizador. Os parâmetros similares ficam agrupados para que não haja confusão por parte do usuário. A denominação das funções está em inglês. Debatido à respeito de nomear em português com o meu orientador e concluímos que as traduções para o português são menos compreensíveis do que manter o padrão comercial internacional. Todos que mexem com sintetizadores estão acostumados com a nomenclatura em inglês, então foi decidido manter essa língua.

Utilizei como base para elaborar meus gráficos a interface gerada com os elementos padrão do Reaktor. Essa foi a primeira versão, utilizando os elementos default:



A elaboração desta versão utilizando os gráficos padrão foi essencial pois através dela que pude pensar a disposição espacial dos elementos de maneira prática. Essa foi a versão para testar o circuito do sintetizador. Enviei para 10 amigos próximos e pude perceber pequenos problemas corrigidos em seguida. Alguns problemas foram de módulos que estavam ligados de maneira errada e outros em que eu havia esquecido de conectar o módulo. Felizmente a versão final aparenta ter todas as suas funções intactas.

Quando estava satisfeito com o circuito do sintetizador, e decidi não acrescentar ou remover nenhuma função, parti para desenvolver o painel. Gerei várias opções de knobs e painéis. Eventualmente, cheguei no que seria a primeira versão da GUI. Por parecer plástica e artificial demais, acabei abandonando essa opção. O contraste nem as cores me agradaram. Mais tarde percebi que nem os knobs estavam com uma estética satisfatória.

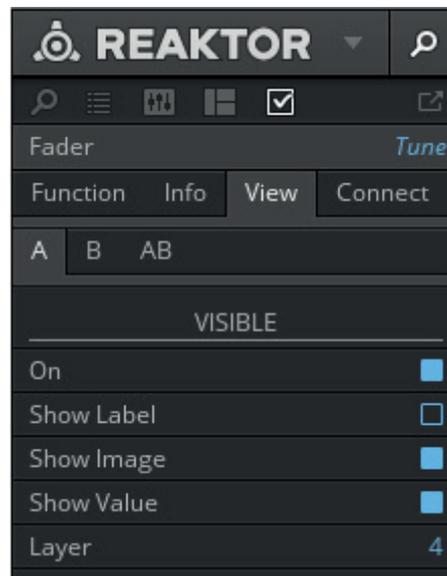


Após detalhada análise de vários instrumentos virtuais já existentes, pixel a pixel, eu pude começar a pensar em algo mais avançado. A versão final será a utilizada para descrever os processos de desenvolvimento à seguir.



4.3.2.1 O desenvolvimento dos elementos gráficos

Antes de apresentar o desenvolvimento dos elementos gráficos, é necessário esclarecer algumas dificuldades e limitações do Reaktor. Primeiramente, o software não possui suporte para aplicação de fontes personalizadas. A fonte nativa do programa é a única utilizável. Para contornar este problema, podemos ocultar o “Label” dos elementos na aba View de cada módulo, o que faz com que o nome do módulo desapareça do painel. O painel de fundo do software deve ser elaborado com as fontes personalizadas renderizadas na imagem.



Outro ponto de dificuldade é a respeito do dimensionamento do painel. O Reaktor não permite a definição de um tamanho fixo. O tamanho do painel é determinado pela quantidade de elementos e seu posicionamento no painel. Cada elemento possui uma margem invisível que, quando aproximada da borda do painel, o estende naquela direção. Não há como alterar ou remover essa margem de maneira satisfatória.



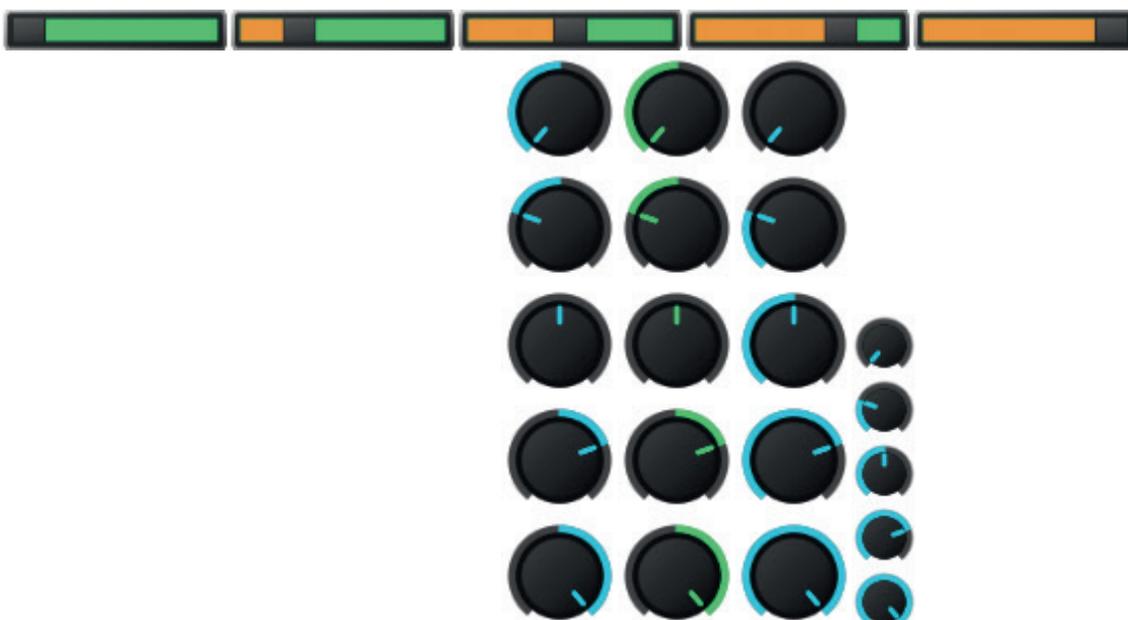
A maneira usada para contornar tal limitação foi introduzir o módulo chamado Text, que se encontra na aba de módulos de painel. Criamos esse módulo e não o preenchemos e desativamos todas as opções na aba view. Na prática isto nos dá um objeto que não pode ser visto no painel, mas que também possui a “margem invisível” que nos permite estabelecer limites ao painel de maneira independente dos outros módulos.

O posicionamento dos módulos no painel apresenta outra restrição, que felizmente pode ser superada em partes. A movimentação dos módulos está presa a um grid de 4x4px, o que limita o posicionamento dos módulos. É possível remover essa limitação clicando em Settings, na parte superior esquerda do programa e em seguida desmarcar a opção “Snap to grid”. Isso permitirá a movimentação pixel a pixel dos objetos. No entanto, o Reaktor não fornece nen-

hum tipo de ferramenta de alinhamento para os objetos. Não há, também, algo semelhante à ferramenta de régua e guia no Illustrator. O único método de alinhamento é utilizando as linhas pontilhadas que aparecem no painel. Essas linhas, por sua vez, estão presas no grid previamente citado, de 4x4 px. Outra maneira de alinhar seria de elaborar o painel integralmente em um grid de 4x4px, o que foi sugerido em momento posterior ao desenvolvimento deste projeto. Isto permitiria um posicionamento mais fácil dos objetos na etapa final de arranjo dos elementos na interface.



Os knobs e faders foram desenvolvidos com o software Knobman. É uma ferramenta muito prática. De maneira similar ao Reaktor, o Knobman possui uma livreria enorme de conteúdo criado por usuários. O software renderiza strips de bitmaps que, lidos por um script agem como animações. O programa funciona de maneira similar ao Photoshop, trabalhando com camadas. É possível para o usuário escolher quantos frames quer renderizar em cada bitmap. Para o nosso sintetizador, sempre iremos renderizar exatamente 128 frames por elemento. Esse número é o número que o software reaktor requer para funcionamento ideal. Existe uma justificativa técnica baseada no número máximo de vozes por canal MIDI, mas a explicação está fora do escopo deste relatório. Exemplos de bitmap strips gerados pelo Knobman:



Dimensões:

Knob grande: 58x58px

Knob pequeno: 32x32 px

Fader: 113x20 px

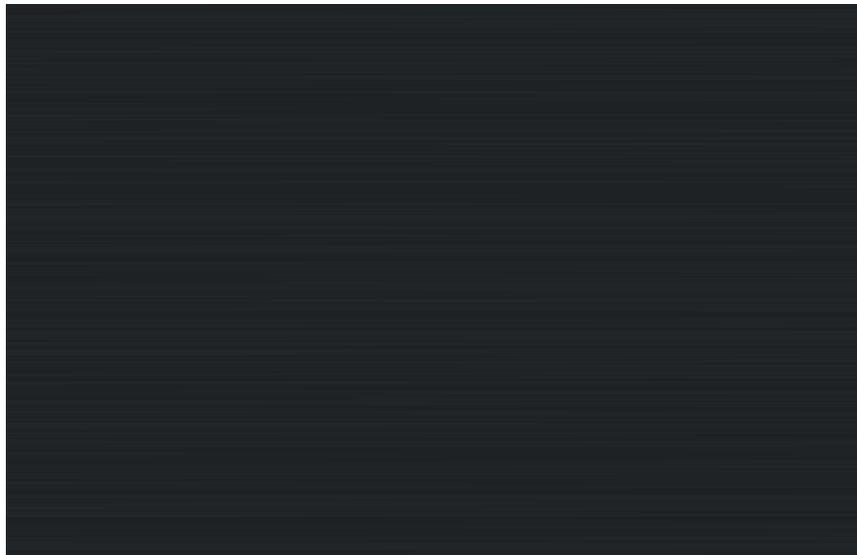
4.3.2.2 O desenvolvimento do painel

Primeiramente devemos falar sobre as dimensões escolhidas para a interface. A dimensão é de 710x460px. Um plugin de dimensões muito grandes ocupa muito espaço na tela e por isso dificulta o uso da interface simultaneamente à outras coisas, por isso optei por uma dimensão relativamente pequena em relação a outros plugins populares semelhantes (ex: Serum ~802x650px, Razor ~922x544px).

Como citado anteriormente, há o problema da definição de dimensão na plataforma. Sendo assim, o painel com os elementos default, já concluído, foi utilizado como “wireframe” para a construção dos gráficos personalizados. O Reaktor não possui uma ferramenta que permita ao usuário exportar uma imagem do painel em desenvolvimento. A maneira encontrada para extrair a imagem do painel foi a realização de print screen. Feito o print, a imagem era colada num artboard do Illustrator e as medidas eram tiradas utilizando-se as ferramentas do programa.

Inicialmente foi determinado um valor arbitrário para os elementos que preenchem o painel, isto é, os knobs, faders, telas. Esse valor “arbitrário” foi escolhido pois, por serem os objetos que o usuário irá interagir primariamente no produto, seriam, de fato, os elementos principais daquela interface, ou seja, a interface deve ser pensada levando em conta a prioridade dada aos elementos que a preenchem. Os knobs, nesse caso foram o objeto que teve sua dimensão pensada num primeiro momento. Foi determinado o valor de 58x58px para os knobs genéricos e o valor de 32x32px para os knobs pequenos. Esses valores foram determinados depois de várias etapas de tentativa e erro criando knobs de diferentes dimensões no Knobman e testando-os no Reaktor até encontrar uma dimensão que fosse agradável.

O painel de fundo do sintetizador é de extrema importância para se obter uma GUI de qualidade. A direção estética que tomei foi a de criar uma interface que transitasse entre uma possível interface real física e uma digital. O painel foi desenvolvido pensando em atingir-se uma textura de aço escovado escuro, remanescente de hardware de áudio.



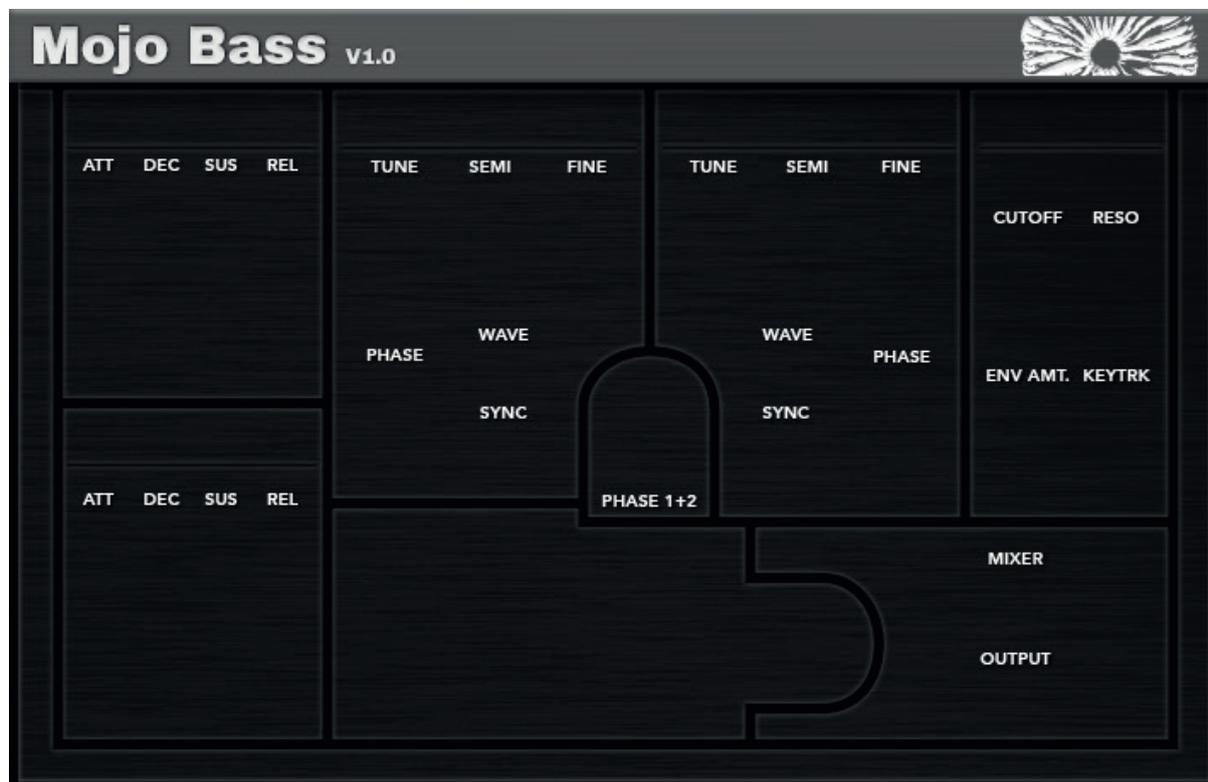
Exemplo de hardware na vida real, meu Roland System-1m. Observe a textura de aço escovado diante à luz. Tentei, no desenvolvimento da GUI, emular essa aparência, mas não com a intenção de fazer um render realista. A intenção era de “emular” a estética.



Após a aplicação da textura, é necessário fazer a separação das diferentes seções do sintetizador. Neste caso foi utilizado um objeto em vetor para as linhas. Esse objeto foi criado no Illustrator, inserido como um Smart Object no Photoshop e em seguida aplica-se uma camada de outer shadow para auxiliar na noção de profundidade.



Abaixo, a tipografia aplicada. Para o alinhamento dos textos com os elementos, utilizou-se o método de espelhamento. O software Reaktor não suporta a utilização de fontes personalizadas em suas ferramentas, sendo necessário renderizar a tipografia personalizada junto do painel de fundo. O Reaktor também não possui uma ferramenta que gere uma imagem do painel do instrumento, o que faz com que seja necessário trabalhar utilizando a ferramenta print screen do computador. Feito o print do painel com seus elementos em seu devido local, a imagem era inserida no Photoshop, diminuiu-se a opacidade para facilitar o alinhamento e então posicionam-se os textos.



Quanto à parte tipográfica, optei por manter uma ótica sóbria e direta. Para os elementos do painel utilizei a fonte Avenir LT Pro 55 Roman Bold. Escolhi essa fonte por sua extrema qualidade em todos os sentidos. A opção por uma geométrica é derivada, talvez da exposição ao longo de vários anos que tive utilizando diferentes plugins de áudio, que em sua maioria utilizam fontes geométricas. Abaixo, Avenir à esquerda e Archivo Black à direita.

Far far away, behind the
word mountains, far
from the countries
Vokalia and
Consonantia, there live
the blind texts.
Separated they live in
Bookmarksgrove right
at the coast of the
Semantics, a large
language ocean.

**Far far away, behind
the word mountains,
far from the
countries Vokalia
and Consonantia,
there live the blind
texts. Separated
they live in
Bookmarksgrove
right at the coast of
the Semantics,**

A fonte escolhida para o nome do instrumento, Mojo Bass foi a fonte Archivo Black. Queria desde o princípio uma fonte simples para o nome, nada muito espalhafatoso. Por isso a opção.

Após feedback da banca examinadora, foi discutido que a marca utilizando a Archivo era

destoante esteticamente do resto da interface gráfica. Optou-se então, por utilizar a Avenir 95 Black numa versão posterior.

O elemento gráfico utilizado no canto superior direito é derivado da marca de meu selo, minha gravadora de áudio. Esse selo foi fundado esse ano e nosso primeiro lançamento será no segundo semestre de 2018. A marca foge do escopo deste projeto mas colocarei aqui por motivos de demonstração.

A marca é relevante de ser introduzida graficamente no projeto pois a intenção do selo é a de contribuir para o psytrance de todas as formas possíveis, não só no lançamento de músicas, mas no caso, no desenvolvimento de ferramentas que contribuam para com o cenário. Como a marca é minha e as intenções dos escopos dos projetos se interseccionam, resolvi por relacionar os dois.



O resultado final do sintetizador, com todos os elementos aplicados:



Algumas considerações: Os mostradores de valor e as caixas de seleção ainda são padrões do Reaktor pois para implementar versões customizadas é necessário elaborar um macro extremamente complicado que envolve indexação de imagens devido à limitações do programa. Isso requer um conhecimento de script que eu não possuo. No entanto, não achei que esse problema atrapalhou muito o resultado final de um ponto de vista estético.

Após a revisão da banca examinadora, fiz leves mudanças gráficas na versão final. Essa versão, com as revisões, será lançada publicamente de maneira gratuita. Obtenha o software através deste link: <http://bit.do/mojobass>

O resultado que será lançado, após feedback da banca examinadora:



Considerações finais

O desenvolvimento deste projeto me trouxe vários bons frutos. Ao apresentar o projeto para amigos, colegas e mesmo desconhecidos, o feedback que recebi foi muito positivo. Um comentário constante que recebi foi de que o projeto era algo pioneiro aqui no Brasil, que não possui nenhuma grande desenvolvedora estabelecida de plugins de áudio virtual. A intenção que fica é a de continuar com o desenvolvimento de novos instrumentos e projetos que possam contribuir para a comunidade Brasileira de produção musical. Atualmente já estou desenvolvendo um novo sintetizador, desta vez um que funciona através de Síntese de Frequência Modulada, ou FM Synthesis. O objetivo deste produto será de desenvolver arpeggios e leads rítmicos com a flexibilidade timbral que a FM proporciona ao instrumento.

O projeto me auxiliou na mudança de percepção que tinha a respeito de se desenvolver plugins de áudio no sentido de desmistificar o processo. Pude compreender em partes o desenvolvimento básico de um sintetizador como um instrumento e compreender o funcionamento em um nível mais lógico.

Agradeço a todos os que me auxiliaram neste projeto, proporcionando auxílio indispensável na conceituação e elaboração. Obrigado.

Referências bibliográficas:

Mencionei previamente a dificuldade de encontrar bibliografia a respeito do assunto. Grande parte do conhecimento que adquiri nesse projeto foi através de pesquisa em fóruns online, conversas com pessoas do ramo e experiências próprias. Para aprender sobre a plataforma Reaktor, fiz um curso online promovido pela ADSR school.

Sites:

Reaktor

<https://www.native-instruments.com/en/products/komplete/synths/reaktor-6/>

Manual de referência do usuário pode ser baixado em

<https://www.native-instruments.com/en/products/komplete/synths/reaktor-6/downloads/>

Reaktor Forums

<https://www.native-instruments.com/forum/forums/reaktor.30/>

KNOBMAN

<https://www.g200kg.com/jp/software/knobman.html>

Tutoriais por ADSR em, curso on-line, pago

<https://www.reakortutorials.com/home>

Tutoriais gratuitos da ADSR

<https://www.youtube.com/user/ADSRtuts>

Martin Finkle's blog

<http://www.martin-finke.de/blog/articles/audio-plugins-007-gui/>

Recording Connection, Glossário

<https://www.recordingconnection.com/glossary/>

Livros

Sintetizadores à Brasileira - Usos e aplicações

Francisco Edson de Souza Pereira. Publicado por H.Sheldon serviços de marketing.

Kefauver, Alan P.; Patschke, David (2007). Fundamentals of Digital Audio, New Edition. A-R Editions. ISBN 9780895796110.