



INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y ELÉCTRICA

AUDIO SURFACE

TESIS

**QUE PARA OBTENER EL TÍTULO DE INGENIERO EN COMUNICACIONES Y
ELECTRÓNICA POR LA OPCIÓN DE TITULACIÓN**

TESIS COLECTIVA Y EXAMEN ORAL INDIVIDUAL DEBERÁN DESARROLLAR

**C. CARLOS FEDERICO CRUZ LAZCANO
C. MIGUEL ÁNGEL MONTOYA JIMÉNEZ**

**ASESORES:
ING. XUNAXI GUADALUPE DE LA CRUZ CARTAS
ING. JOSÉ JAVIER MUEDANO MENESES**

MÉXICO, D.F.

2010



INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y ELECTRICA
UNIDAD PROFESIONAL “ADOLFO LÓPEZ MATEOS”

TEMA DE TESIS

**QUE PARA OBTENER EL TÍTULO DE INGENIERO EN COMUNICACIONES Y ELECTRÓNICA
POR LA OPCIÓN DE TITULACIÓN TESIS COLECTIVA Y EXAMEN ORAL INDIVIDUAL
DEBERA(N) DESARROLLAR**

**C. CARLOS FEDERICO CRUZ LAZCANO
C. MIGUEL ÁNGEL MONTOYA JIMÉNEZ**

“AUDIO SURFACE”

**DESARROLLAR UN INSTRUMENTO MUSICAL ELECTRO-ACÚSTICO TANGIBLE PARA LA
MEZCLA DE MÚSICA ELECTRÓNICA, QUE LE PERMITA AL USUARIO GOZAR DE UN MANEJO
INTERACTIVO DONDE SE INVOLUCREN LOS SENTIDOS DE LA VISTA, OÍDO Y EL TACTO.**

- ❖ **INTRODUCCIÓN.**
- ❖ **ANTECEDENTES.**
- ❖ **MARCO TEORICO.**
- ❖ **DESARROLLO.**
- ❖ **ANÁLISIS DE COSTOS.**
- ❖ **CONCLUSIÓN.**
- ❖ **ANEXOS.**
- ❖ **BIBLIOGRAFÍA.**

MÉXICO D. F., A 08 DE NOVIEMBRE DE 2010.

ASESORES


ING. XUNAXI GUADALUPE DE LA CRUZ CARTAS


ING. JOSÉ JAVIER MUEDANO MENESES


**M. EN C. SALVADOR RICARDO MENESES GONZÁLEZ
JEFE DEL DEPARTAMENTO ACADÉMICO DE
INGENIERÍA EN COMUNICACIONES Y ELECTRONICA**

**SECRETARÍA
DE I.C.E.**

Índice

Objetivo	5
Justificación	6
Introducción	7
Capítulo I: Antecedentes	9
1.1 Evolución de la Música Electrónica	10
1.2 Historia de el Disc Jockey's (DJ's)	16
1.3 Evolución del Sintetizador	20
1.3.1 Funcionamiento del Sintetizador	22
1.3.2 La Síntesis del Sonido y las Técnicas de Procesamiento	23
1.3.3 Música por Computadora y Música Electro-acústica	24
Capítulo II: Marco Teórico	27
2.1 Sonido	28
2.2 Música	28
2.3 Ritmo	28
2.4 Timbre	29
2.5 Tono	29
2.6 Ruido	30

2.7 Tipos de Ruido	30
2.7.1 Ruido Blanco	30
2.7.2 Ruido Rosa	31
2.8 Síntesis	32
2.8.1 Tipos de Síntesis	32
2.8.1.1 Síntesis Sustractiva	32
2.6.1.2 Síntesis Aditiva	33
2.6.1.3 Síntesis por Tabla de Ondas	34
2.6.1.4 Síntesis por Modulación de Frecuencias	35
2.9 Audio Digital	35
2.10 Caja de Ritmos	37
2.11 Secuenciador	38
2.12 Sample	39
2.13 reacTIVision	39
2.14 Pure Data	40
2.15 Protocolo TUIO	41
2.15.1 TUIO Trackers	42
2.15.2 TUIO Client	43
2.16 Protocolo OSC [Open Sound Control]	43
2.17 Trabajos Relacionados	44

Capítulo III: Desarrollo	47
3.1 Configuración de reactIVision	49
3.1.1 Proceso de Configuración	50
3.1.1.1 Elección de Patrón Fiducial y Puerto de Comunicaciones	50
3.1.1.2 Configuración en Tiempo Real	51
3.2 Configuración de Pure Data	52
3.2.1 Configuraciones de Audio	52
3.2.1.1 Tasa de Muestreo	53
3.2.1.2 Delay (Latencia)	53
3.2.1.3 Dispositivos de Entrada y Salida	54
3.2.1.4 Objetos Gráficos	54
3.3 Programación	55
3.3.1 Decodificación de reactIVision	55
3.3.2 Reproductor de Samples	61
3.3.3 Velocidad de Reproducción	62
3.3.4 Scratch	62
3.3.5 Delay	63
3.3.6 “patch” Final	64
3.4 Descripción de el manejo de los Knob's	66
3.4.1 Control de On/Off (ID = 0)	67
3.4.2 Control del Sampler (ID 1-3)	68
3.4.3 Control de la Velocidad de Reproducción (ID 4-6)	69

3.4.4 Control de los Efectos (ID 7-8)	70
3.5 Diseño de la mesa	72
3.5.1 Mesa y Superficie	72
3.5.2 Iluminación	72
3.5.3 Cámara	73
Capítulo IV: Análisis de Costos	74
Conclusión	77
Anexos	79
Anexo A: Glosario de Anglicismos	80
Anexo B: Símbolos Fiduciales	84
Anexo C: Datos Biográficos Breves	80
Bibliografía	95

Objetivo

Desarrollar un instrumento musical electro - acústico tangible para la mezcla de música electrónica, que le permita al usuario gozar de un manejo interactivo donde se involucren los sentidos de la vista, oído y tacto.

Justificación

El género electrónico musical le dio un nuevo dinamismo a la forma de hacer música, el uso de tornamesas, mezcladoras, sintetizadores le brindaron al compositor una libertad de crear cuanto sonido le viniera en mente. Estos “nuevos” instrumentos le daban una practicidad y libertad, que en un espectáculo en vivo, aprovechaba para bailar e interactuar de una forma diferente con su público.

Por muchos años no hubo alguna innovación en la creación de nuevos instrumentos que siguieran proporcionando esa versatilidad al Dj, ya que sólo se incluyeron CD players y computadoras. Es por tal motivo que como ingenieros en Comunicaciones y Electrónica con especialidad en Acústica, nos dimos a la tarea de investigar y buscar una nueva plataforma interactiva para la mezcla de música electrónica.

Es por esto que se llegó a la conclusión de que se necesitaba diseñar una interfaz tangible que le permitiera el Dj explorar la música de nuevas formas y que le brindara una gran movilidad, versatilidad y le permitiera involucrar más de un sentido en su creación musical.

Lo que se pretende con **Audio Surface** es crear un instrumento electro - acústico tangible e interactivo, con el cual el Dj sea capaz de manipular una serie de objetos para mezclar sonidos, manipular parámetros, añadir efectos y que le permita llevar su creatividad musical a otro nivel.

Introducción

La música ha formado una parte importante en el desarrollo de las culturas alrededor del mundo, ha servido para la expresión de sentimientos e ideas. Esta ha evolucionado en conjunto con el ser humano y muchos han sido los instrumentos utilizados para su creación, desde rudimentarias piedras y maderas hasta los instrumentos más modernos como sintetizadores, cajas de ritmos, computadoras, etc.

La música electrónica tiene un puesto importante en la historia musical, ya que desde principios del siglo XX una serie de compositores experimentales, en busca de la creación de nuevos ritmos y formas de expresión, buscaron instrumentos no acústicos con los que pudieran crear nuevos sonidos. Para ello, se hizo uso de los avances tecnológicos del momento; desde el *Theremin* del ruso **Leon Theremin** y pasando por las *Ondas-Martenot*, compuestas por **Maurice Martenot**.

Posteriormente el inventor estadounidense **Robert Moog**, en su búsqueda incesante de nuevas tecnologías, fundó la base para la creación de música electrónica; junto con otros colaboradores inventó el sintetizador, el cual abrió significativamente las puertas a una inmensa gama de sonidos.

Existen personas que afirman que la electrónica resta creatividad a la música, en épocas anteriores un compositor debía esperar para escuchar su propia sinfonía, ya que esta dependía de una orquesta que se decidiera a tocarla. Hoy en día gracias a los sintetizadores, secuenciadores y computadoras, el autor puede escucharla antes de hacerla pública.

La creatividad musical del compositor ha dado un salto impresionante. Hoy el autor es capaz de componer sin la tediosa mediación de partituras, conocimientos de solfeo y demás. Puede interpretar todos los instrumentos que aparecen en su trabajo y diseñarlos a su antojo, con lo cual no se limita a crear notas, sino también sonidos. Incluso puede generar e interpretar en el teclado voces, coros, lluvias, ventiscas, cantos de aves, y cuantos sonidos captamos en nuestro universo, gracias a los samplers. Tiene en sus manos el poder expresar todo lo que surge de su creatividad.

Audio Surface será un instrumento musical electro - acústico tangible e interactivo, basado en una superficie sensible (mesa), la cual es tocada mediante la manipulación de objetos (**knob's**), que son colocados sobre ella. Cada objeto cuenta con una función específica para generar, modificar y controlar el sonido. Estos objetos están marcados en su parte inferior por un código fiducial. Una cámara de video en conjunto con reactIVision son los encargados de rastrear su colocación y orientación, así como de mandar esta

información por medio de un protocolo TUIO a otros dispositivos, los cuales se encargan de decodificar dicha información.

Esta nueva generación de instrumentos electro - acústicos han abierto la posibilidad de un manejo físico de acciones y parámetros, dando como resultado un control más atractivo, en el que pueda conjuntar el mayor número de sentidos.

En el Capítulo I se encontrará un compendio de antecedentes acerca de la historia de la música electrónica y su evolución en el tiempo. Se habla también del cambio que han tenido los instrumentos electro - acústicos usados para su creación, como sintetizadores y su funcionamiento. Así mismo se tocará el surgimiento y desarrollo de los Dj's, los cuales se convirtieron en la parte más importante del movimiento electrónico.

En el siguiente capítulo se describirá el marco teórico de este proyecto; es decir se tratará de explicar de la mejor forma posible el funcionamiento de cada una de las partes que comprenden a **Audio Surface**, el software a utilizar, protocolos, etc.

El Capítulo III contendrá los pasos a seguir y problemas enfrentados al momento de realizar este proyecto, se podrá ver el funcionamiento más detallado de cada uno de los programas, el método de programación usado, así como el proceso de decodificación de lo que detecta la cámara de video, como se realiza el link ó enlace con el sonido y cómo este puede manipularse con el movimiento de un objeto.

El análisis económico es una parte muy importante en el desarrollo de un proyecto, ya que es aquí donde se determina que tan viable es. Esto debido a que si el proyecto es costoso, no será factible ya que el precio de comercialización sería alto, por lo que no podrá ser distribuido con facilidad. Esto se detallará en el Capítulo IV.

Capítulo I

Antecedentes

1.1 Evolución de la Música Electrónica

Hablar hoy en día de música electrónica equivale casi a hablar de música en general, pocos son los trabajos, no sólo en las corrientes pop sino dentro de toda la música compuesta actualmente, en las que no aparezca algún sonido generado electrónicamente. Esto aunado al hecho de que **Audio Surface** es un instrumento electro - acústico, que brindará una nueva forma para mezclar sonidos electrónicos de una manera más interactiva, hace interesante conocer el origen y evolución de la música electrónica.

Se podría pensar que la historia de la música electrónica es corta y se enfoca a unos pocos años, pero esta idea es errónea ya que sus inicios se remontan a principios del siglo XX. En 1913 el pintor futurista italiano **Luigi Russolo** [Fig. 1], concibe la idea de utilizar los ruidos cotidianos para elaborar una nueva era de expresión musical, esto lo lleva a publicar el manifiesto llamado *Ruidismo* (el Arte del Ruido) el cual iba acompañado de una máquina denominada *Intonarumori* (Entona Ruidos), por lo que se le conoce como precursor de la música concreta.



Fig. 1 Luigi Russolo
[Precursor de la Música Concreta]

Años más tarde, en 1919, el físico ruso **Leon Theremin** [Fig. 2a] crea un dispositivo al cual le da por nombre *Theremin*. Su diseño consiste en una caja con dos antenas y su funcionamiento se basa en la distancia de las manos con respecto a cada una de ellas, la antena horizontal servía para controlar el volumen, mientras que la antena vertical controlaba la frecuencia.

En 1928 el compositor, ingeniero y chelista francés, **Maurice Martenot** [Fig. 2b], crea un instrumento electrónico llamado, *Ondas-Martenot*; este instrumento está formado por un teclado, altavoz y un generador de baja frecuencia, este era un instrumento monofónico; es decir, no producía ondas simultáneas. A finales de la década de los 40's el músico francés, **Pierre Schaeffer** [Fig. 2c], da a conocer en Paris los frutos de sus experiencias sonoras y con esto nace formalmente la música concreta, la cual se define como una serie de fenómenos sonoros

existentes, manipulados y transformados por medio de instrumentos electro - acústicos.



Fig. 2a Leon Theremin
[Inventor del Theremin]



Fig. 2b Maurice Martenot
[Componer Ondas-Martenot]



Fig. 2c Pierre Schaeffer
[Padre de la Música Concreta]

En 1950 se inician de la mano del alemán **Herbert Eimert** [Fig. 3a], algunas de las experiencias más renovadoras en la música de este siglo, ya que la fuente sonora primordial eran los sonidos producidos por generadores eléctricos, ondas sinusoidales, ruido blanco, etc. Todo esto se vió beneficiado con la aparición de los circuitos integrados, ya que se crearon una nueva generación de instrumentos electrónicos de fácil uso, fiables y populares.

A finales de 1963 y a raíz de un encuentro con el profesor de música y compositor experimental **Herb Deutsch** [Fig. 3b], el investigador **Robert Moog** [Fig. 3c] buscó la forma de darle un uso musical a los osciladores electrónicos; su diseño consistía en el empleo de un control de tensión para variar el tono del oscilador o el volumen del amplificador. Después de un año, **Robert Moog** le dio vida al primer sintetizador llamado *Modular Moog* [Fig. 3d].



Fig. 3a Herbert Eimert
[Compositor Experimental]



Fig. 3b Herb Deutsch
[Compositor Experimental]



Fig. 3c Robert Moog
[Inventor del Sintetizador]

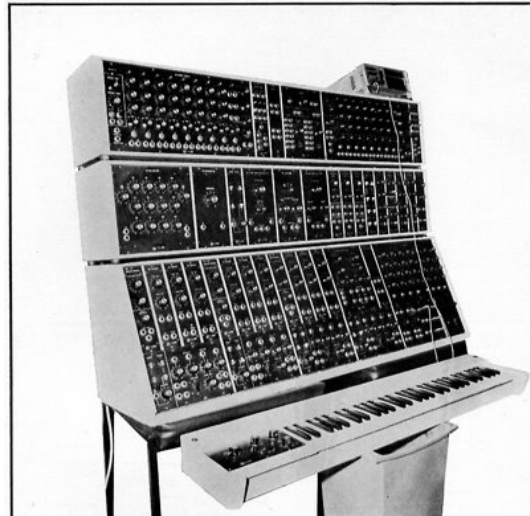


Fig. 3d Modular Moog

Gran parte de estos sintetizadores fueron creados para universidades o investigadores reconocidos como **John Cage** [Fig. 4a] y **Vladimir Ussachevsky** [Fig. 4b], lo que hacía que el sintetizador estuviera muy alejado de las masas. Sin embargo **Wendy Carlos** [Fig. 4c], tenía otras ideas y comenzó a grabar con sintetizadores, interpretaciones de la música clásica de Bach, lo que la llevó a crear *Switched On Bach*, el cual se convirtió en un éxito musical y técnico, ya que se aprovechó de un modo espectacular la suma de la potencia y gama expresiva del *Modular Moog*.

Pero lo más importante es que supuso un enorme éxito mundial que vendió más de un millón de copias y acabó por desplazar al sintetizador de las sombras académicas para ganarse un papel protagonista en la música popular.

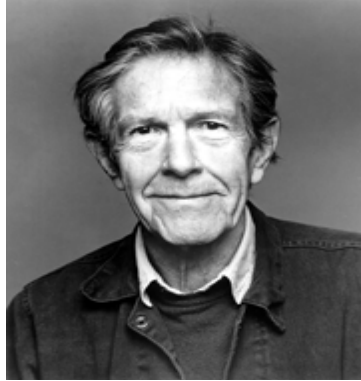


Fig. 4a John Cage



Fig. 4b Vladimir Ussachevsky

[Investigadores]



Fig. 4c Wendy Carlos
[Compuso Switched On Bach]

El final de la década de los 60's marca el inicio de cierta experimentación electrónica de algunas bandas como **Pink Floyd** [Fig. 5a], **Tangerine Dream** [Fig. 5b], etc. Pero donde se estaba expandiendo el concepto musical electrónico era en el tercer mundo, particularmente en los ghettos de Kingston, Jamaica. Empujados por el poder y visión del rastafarismo, se crea el Dub, de la mano de **King Tubby** [Fig. 5c], quien inicia esta revolución sónica y se convierte en el creador del remix.



Fig. 5a Pink Floyd



Fig. 5b Tangerine Dream



Fig. 5c King Tubby
[Creo el Dub y Remix]

Así mismo en Alemania se da el surgimiento de un grupo el cual sería considerado como el primer grupo auténticamente electrónico, **Kraftwerk** [Fig. 6], quien es definitivamente la raíz de todo lo que hoy suena a música electrónica. **Kraftwerk** tiene sonidos mecánicos, melodías sencillas, un poco repetitivas y su atenta mirada a la complicada relación del ser humano con la tecnología (máquinas, robots y computadoras).



Fig. 6 Kraftwerk
[1er Banda de Música Electrónica]

Con el auge de los nuevos conceptos electrónicos, resultaron intérpretes como el griego **Vangelis** [Fig. 7a] o el francés **Jean Michael Jarré** [Fig. 7b], quienes explotaron en todos los sentidos esas nuevas técnicas, en un subgénero de fácil proceso que dio pie a lo que hoy se llama Ambient.

A principio de los años 70's, comienza a aparecer un nuevo tipo de música urbana promovida por un DJ llamado **Kool Herc** [Fig. 7c], quien empezó a cantar versos sobre las secciones de percusión de los éxitos pop del momento, para extender la duración de estas sesiones utilizó un mezclador de audio y dos tornamesas con el mismo disco y en el proceso inventó el arte de la mezcla y de la música Hip-Hop o Rap.



Fig. 7a Vangelis



Fig. 7b Jean Michael Jarré

[Pioneros del Ambient]



Fig. 7c Dj Kool Herc

[Crea la mezcla y el Hip-Hop]

A partir de este momento en muchos lugares donde había jóvenes segregados por su condición social, empezaron a brotar corrientes musicales y sonidos nunca antes escuchados. El house nace de la mano del **DJ Frankie Knuckles** [Fig. 8a], residente de una discoteca llamada Warehouse de Chicago. Poco tiempo después surge en Detroit, el Techno, un género producido por **Kevin Saunderson** [Fig. 8b] y **Derrick May** [Fig. 8c] influenciados por la música de **Kraftwerk**, que no se sentían representados por ninguno de los géneros musicales del momento.



Fig. 8a Dj Frankie Knuckles

[Crea el género House]



Fig. 8b Dj Kevin Saunderson



Fig. 8c Dj. Derrick May

[Creadores del género Techno]

Estos movimientos tuvieron su momento cúlpe cuando a finales de los 80's en las fiestas de lo que se denominó Acid House, donde se combinaba música de diversos géneros electrónicos con el uso de drogas. Hoy en día la música electrónica sigue creciendo y la cantidad de subgéneros es muy grande (Progressive, Tech House, Electro, Psy-Trance, etc.). Así mismo la cantidad de DJ's es impresionante, como por ejemplo, Paul van Dyk, Sasha, Carl Cox, Gabriel & Dresden, Above & Beyond, ente muchos otros.

La importancia de los DJ's se centra en la aparición de herramientas como el sampler, el cual permite crear canciones con pedazos de otras (ayudados con tornamesas y vinilos), o softwares para computadoras que permiten hacer toda clase de ensambles, recortes, efectos con mínimos esfuerzos.

1.2 Historia de el Disc Jockey (DJ)

El Dj [Fig. 9] es una parte muy importante en el desarrollo de la música electrónica ya que este es el encargado de poner la música, pero no sólo se encarga de mezclar entre una canción y otra, también funge en muchos casos como productor de música electrónica, es decir crea su propia música, compone y después la reproduce, como **Audio Surface** es una interfaz diseñada para la mezcla de música electrónica, no se puede dejar de lado la parte más importante de esta, el DJ.



Fig. 9 Disc Jockey – DJ

El comienzo de la historia del DJ va de la mano con el surgimiento del Dub, ya que con este se genera una nueva forma de entretenimiento musical, conocido como *Sound Systems* [Fig. 10]. Estos eran vehículos equipados con altavoces y equipo de audio, de donde se reproducía música como Blues, Ska, Reggae y Dub, con el objeto de que la gente gozara de buena música y conociera los nuevos ritmos generados por el Disc Jockey, este era el encargado de seleccionar y poner la música en estas *fiestas callejeras*.



Fig. 10 Sound Systems

Después de haberse dado a conocer y consolidarse como el creador de la mezcla y del Hip-Hop, el jamaquino **Clive Campbell** [Fig. 11a] viaja a Nueva York, donde posteriormente se daría a conocer como **Dj Kool Herc**. Hacia 1973 y a la corta edad de 15 años, tratando de generar en su ghetto algunas de las costumbres de su tierra natal, comienza a realizar *fiestas callejeras* al estilo de los *Sounds Systems*, que resultan ser un éxito.

Todo esto sería sólo una simple anécdota, si no fuera por el hecho de que en estas mismas fiestas a **Dj Kool Herc** se le ocurrió inventar una de las técnicas más usadas en Dj'ing hasta hoy en día como es el backspin y así dar comienzo

a toda una nueva forma de expresión musical al mando de las tornamesas [Fig. 11b].



Fig. 11a Clive Campbell



Fig. 11b Tornamesas

El backspin es una técnica que consiste en repetir una sección determinada de música a través de dos bandejas y un mezclador con dos discos iguales, así mientras está sonando la sección de breakbeat (parte instrumental de la canción) en uno de los discos, el otro se retrocede y se "prepara" para repetir el mismo breakbeat. La intención de **Kool Herc** era lograr extender las secciones de breakbeat y así darle a los B-Boys (break-boys, llamados así porque se dedicaban a bailar en los breakbeats) más tiempo para realizar sus pasos de baile, ya que éstos aprovechaban esta sección de la música para tirarse al suelo a demostrar sus pasos de baile. Por otro lado, también se propiciaba que el "animador" de la fiesta (ahora MC) pudiera animar al público sin "pisar" las partes vocales de la música que estaba sonando.

A partir de ese momento es cuando nace la leyenda del "Padre del Dj'ing" que no es ni más ni menos que **Dj Kool Herc**. Posteriormente vinieron otros grandes Dj's como **Grandmaster Flash** [Fig. 12a] y **Grand Wizard Theodore** [Fig. 12b], que le agregaron a esta actividad nuevas e innovadoras formas de mezclar música. Técnicas como el scratch que fue inventada por **Grand Wizard Theodore**, pasaron a formar parte de la cultura del Dj'ing. El tiempo fue pasando las técnicas se fueron refinando y mejorando, el Dj'ing y el Dj fueron tomando protagonismo dentro de la escena musical, aparecieron grupos de Rap como **Run DMC**, que dependían de un Dj, que iba generando desde los platos las bases musicales para que los MC's canten sobre éstas.

El Dj también participa activamente en grupos de música, como percusionista aportando efectos, samples y principalmente el scratch a los temas. En otros ámbitos la imagen del Dj también va teniendo protagonismo con la aparición de nuevas versiones de temas hechas por DJ's, remixes y discos de mezclas. Todo esto va generando un crecimiento y popularidad del Dj, no sólo como la persona

encargada de "poner" música sino también por su papel como productor de bandas, músico, remixer y turntablist.



Fig. 12a Grandmaster Flash



Fig. 12b Grand Wizard Theodore
[Creó el Scratch]

En 1987 se realiza el primer campeonato mundial de DJ's de la DMC (Disco Mix Club). Los mejores DJ's del mundo se dan cita para demostrar sus técnicas y habilidades en el manejo de los platos. El evento es patrocinado entonces y hasta la fecha por la empresa *Technics*. Participa en la inauguración el grupo **Public Enemy** [Fig. 13a] que en ese entonces tenía mucha popularidad. Este campeonato de DJ's se siguió realizando todos los años desde 1987 en adelante y se fueron dando a conocer DJ's que luego serían leyendas en el mundo del Dj'ing como, **Dj Noise** [Fig. 13b], **Crazy B** y muchos más.



Fig. 13a Public Enemy



Fig. 13b Dj Noise

Todos estos DJ's y otros mostraron nuevas técnicas de manipulación de discos, mejorando los sonidos del scratch, creando el beat-juggling mediante

manipulación de beats y todas estas nuevas formas de manipular los sonidos recibieron el nombre de turntablism.

1.3 Evolución del Sintetizador

Los primeros sintetizadores fueron construidos en la década de 1920, no fue hasta los años 60's cuando comenzaron a popularizarse. Su desarrollo tuvo lugar principalmente en los laboratorios de electrónica de las universidades de los Estados Unidos. Allí, algunos pioneros como **Robert Moog** construyeron prototipos de sintetizadores e hicieron demostraciones. Al principio, el sintetizador era visto como algo puramente experimental y elitista, quizá por el hecho de que sólo algunos artistas vanguardistas se atrevieron a componer música hecha para sintetizadores [Fig.14].



Fig. 14 Sintetizador

En el año 1968, un músico llamado **Wendy Carlos** en colaboración con **Robert Moog** grabó una serie de obras de Johann Sebastian Bach en un disco llamado *Switched-on Bach* (conocido en los países de habla hispana como *Bach Electrónico*), usando un sintetizador *Modular Moog* y una grabadora de 4 pistas. El álbum fue recibido con inusual atención, vendió cientos de miles de copias, fue el primer álbum de música clásica en obtener un premio Grammy, vender un millón de copias y probó al público que el sintetizador podía ser adaptado a la música tradicional.

Con el surgimiento de un nuevo mercado los fabricantes diseñaron modelos más pequeños como el *Minimoog*, [Fig. 15a] y además comenzaron a aparecer fábricas en Japón de la mano de marcas como Roland y Yamaha. Los nuevos estilos musicales de los 70's, como el rock progresivo, demandaban nuevos sonidos y el sintetizador fue adoptado con gusto. Algunos de los héroes de los sintetizadores de los años 70's fueron **Keith Emerson** de *Emerson, Lake & Palmer* [Fig. 15b] y **Rick Wakeman** de Yes [Fig. 15c].



Fig. 15a Minimoog

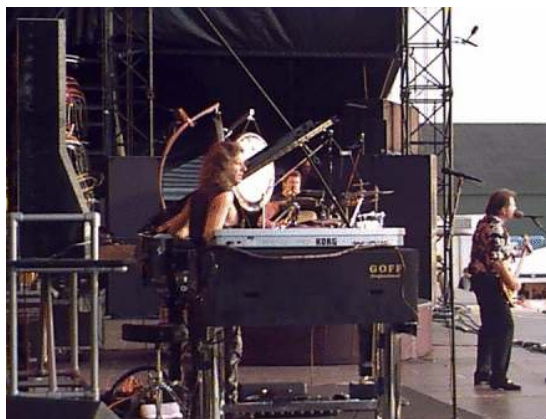


Fig. 15b Emerson, Lake & Palmer



Fig. 15c Rick Wakeman

Hasta este momento los sintetizadores eran usados para agregar sonidos novedosos a los instrumentos ya existentes. Pero con la llegada de la tecnología digital fue posible que éstos comenzaran a emular instrumentos ya existentes, como vibráfonos y pianos eléctricos.

En esta época, nació la que hoy es considerada la música electrónica de mano de pioneros como el grupo alemán **Kraftwerk**. La música electrónica derivó en sub-estilos como el techno pop o synth-pop, donde se destacaron bandas como Depeche Mode, Orchestral Manoeuvres in the Dark, Ultravox, Alphaville, The Human League, Erasure, Soft Cell, Gary Numan, Pet Shop Boys y más en la actualidad grupos como Destroyer, The Russian Futurist o los españoles Limousine.

En 1983 se adaptó la interfase MIDI [Fig.16], que permitía a diferentes marcas de sintetizadores comunicarse entre sí y grabar lo que se tocaba en un aparato llamado secuenciador. Esto revolucionó la forma de hacer música ya que cualquiera podía programar aunque no fuera un buen ejecutante. Los sintetizadores más representativos de esta era son el Yamaha DX7 y el Roland D-50.



Fig. 16 Interfase MIDI

A comienzos de los 80's se desarrolló también el sampler o muestreador, que permitía grabar sonidos reales y reproducirlos. Esta tecnología posibilitó lo que es común hoy en día, que se utilicen sintetizadores y samplers para emular casi todos los sonidos existentes.

1.3.1 Funcionamiento del Sintetizador

Un sintetizador es un instrumento musical electrónico diseñado para producir sonido generado artificialmente, usando técnicas como síntesis aditiva, substractiva, de modulación de frecuencia, de modelado físico o modulación de fase, para crear sonidos [Fig. 17].



Fig. 17 Sintetizador Actual

El sintetizador crea sonidos mediante manipulación directa de corrientes eléctricas (como los sintetizadores analógicos), mediante la manipulación de una onda FM digital (sintetizadores digitales), manipulación de valores discretos usando computadoras (sintetizadores basados en software), o combinando cualquier método.

En la fase final del sintetizador, las corrientes eléctricas se usan para producir vibraciones en altavoces, auriculares, etc. Este sonido sintético se distingue de la grabación de sonido natural, donde la energía mecánica de una onda de sonido se transforma en una señal que más tarde se convertirá de nuevo en energía mecánica durante su reproducción.

1.3.2 La Síntesis del Sonido y las Técnicas de Procesamiento

Las diferentes técnicas utilizadas para crear música electrónica pueden dividirse entre las que se usan para crear nuevos sonidos y las que se utilizan para modificar, o transformar, sonidos ya registrados. En épocas pasadas se usaban osciladores y filtros de control por voltaje, pero estas funciones las realiza hoy la computadora. La creación de sonidos completamente nuevos obtenidos por la superposición o la mezcla de sonidos puros es un proceso llamado síntesis aditiva.

Por otra parte, el modelado de sonidos o filtración de ruido se conoce como síntesis sustractiva. El compositor francés **Jean-Claude Risset** [Fig. 18a], ha diseñado muchos timbres por síntesis aditiva en obras como *Mutations I* (1979), mientras que el estadounidense **Charles Dodge** [Fig. 18b], usa la síntesis sustractiva con gran efectividad en su obra *Cascando* (1983).



Fig. 18a Jean Claude Risset



Fig. 18b Charles Dodge

La síntesis por frecuencia modulada (FM), documentada por vez primera por el compositor estadounidense **John Chowning** en la década de 1960 [Fig. 19], se emplea como vía para crear sonidos sintéticos complejos. Su pieza para cinta *Stria* (1976) está compuesta de varios timbres FM mezclados. Se pueden usar filtros de diferentes tipos para acentuar tonos o armonías específicas, y los sonidos pueden prolongarse en el tiempo, o elevarse y bajarse de tono, gracias a técnicas como el *vocoding de fase* o codificación predictiva lineal.

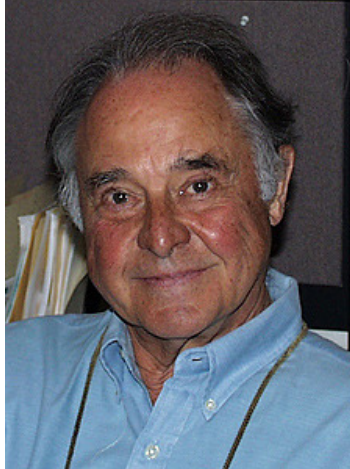


Fig. 19 John Chowning

Las *seis fantasías sobre un poema de Thomas Campion* de **Paul Lansky** (1979) [Fig. 20], son una obra pionera en esta área. Además, se usan técnicas de reverberación para añadir ecos o un cierto retardo a los sonidos grabados o sintetizados, mientras que se utilizan técnicas de simulación de espacios o salas para imitar el carácter acústico de las grandes salas de conciertos y otros espacios. Todas estas técnicas, aisladas o combinadas, son de uso frecuente entre los compositores de música electrónica.

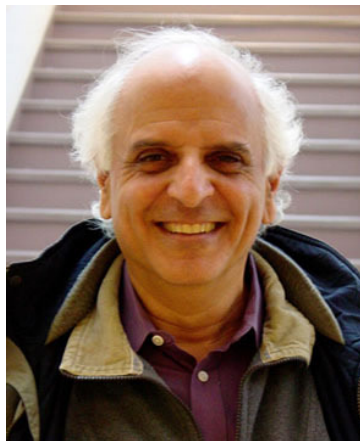


Fig. 20 Paul Lansky

1.3.3 Música por Computadora y Música electro - acústica

La música por computadora se caracteriza por el empleo de la tecnología digital. Si la música creada mediante el corte y pegado de cintas utilizaba grabadoras y dispositivos analógicos, las computadoras almacenan la información de forma digital, como números que pueden manipularse mediante procesos matemáticos usando programas informáticos. Desde la década de 1960, los términos música electro - acústica y por computadora se refieren, respectivamente, aunque no de forma constante, a la música analógica y a la digital. Hoy esos términos se usan

casi de forma equivalente, dado que casi toda la música electrónica utiliza ahora tecnología digital.

Son hoy muchos los compositores que escriben música electrónica, en una gran variedad de estilos. En alguno de estos casos la tecnología digital ofrece la oportunidad de crear una música para cinta magnetofónica que explore los sonidos del entorno natural.

El compositor francés **Luc Ferrari** [Fig. 21a] ha escrito una serie de piezas, *Presque Rien n° 1-3*, usando grabaciones directas de los sonidos del mar o del campo. Otros compositores, como **Denis Smalley** [Fig. 21b] en el Reino Unido o el argentino **Horacio Vaggione** [Fig. 21c], continúan la tradición de la música concreta en la que los sonidos grabados son sometidos a un procesamiento y una edición muy intensa.



Fig. 21a Luc Ferrari

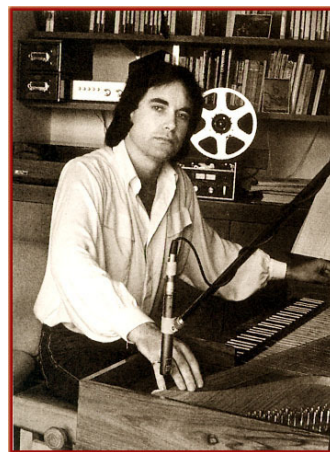


Fig. 21b Denis Smalley



Fig. 21c Horacio Vaggione

Este tipo de música, en la que los sonidos grabados son tratados como objetos sonoros abstractos. Para compositores como el canadiense **Robert Normandeau** [Fig. 22a], es importante cómo se presenta la música de cinta en

una actuación, su obra *Spleen* (1993) ha de interpretarse con varios altavoces. Otros autores, como **Luis de Pablo** en España [Fig. 22b] o **Juan García Esquivel** [Fig. 22c] en México, también se han servido de medios electrónicos para componer sus obras.



Fig. 22a Robert Normandeau



Fig. 22b Luis de Pablo



Fig. 22c Juan García Esquivel

Capítulo II

Marco Teórico

2.1 Sonido

Se define como cualquier fenómeno que involucre la propagación en forma de ondas elásticas audibles o casi inaudibles, generalmente a través de un fluido (u otro medio elástico) que este generando movimiento vibratorio de un cuerpo.

El sonido humanamente audible consiste en ondas sonoras consistentes en oscilaciones de la presión del aire, que son convertidas en ondas mecánicas en el oído humano y percibidas por el cerebro. La propagación del sonido es similar en los fluidos, donde el sonido toma la forma de fluctuaciones de presión. En los cuerpos sólidos la propagación del sonido involucra variaciones del estado tensional del medio

La propagación del sonido involucra transporte de energía sin transporte de materia, en forma de ondas mecánicas que se propagan a través de la materia sólida, líquida o gaseosa. Como las vibraciones se producen en la misma dirección en la que se propaga el sonido, se trata de una onda longitudinal.

2.2 Música

Mousikē es un concepto griego que significa “el arte de las musas”. De allí deriva la palabra música, que define al arte de organizar sensible y lógicamente una combinación coherente de sonidos y silencios. Para esto, se utilizan los principios fundamentales de la melodía, la armonía y el ritmo.

Como manifestación artística, la música es un producto cultural que busca suscitar una experiencia estética en el oyente. También constituye un estímulo en el campo perceptivo del hombre, que puede cumplir con distintas funciones, como el entretenimiento, la ambientación o la comunicación.

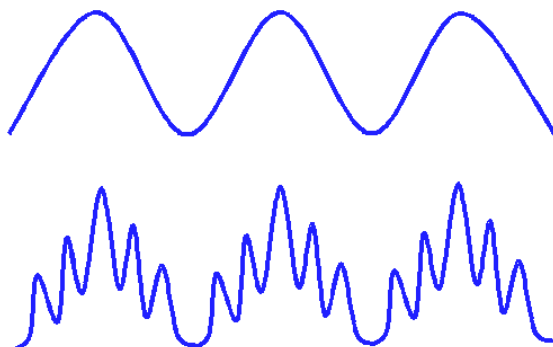
2.3 Ritmo

Se denomina ritmo, a la sucesión de los acontecimientos, cuando estos guardan un orden acompasado. Los elementos se repiten armoniosa y organizadamente, ya sean sonoros o visuales, conformando una estructura. Es un concepto dinámico, ya que necesita de un fluir. Lo estático carece de ritmo. Se puede aplicar no solo al arte, que es su ámbito más específico, sino a otros espacios vitales, por ejemplo, al ritmo de sucesión de ciertos fenómenos meteorológicos.

El ritmo musical es la proporción que guarda el tiempo de un movimiento, y el de otro distinto. Depende de las diferentes duraciones de sonidos y silencios. En la música instrumental, el ritmo es la repetición y periodicidad de distintas formas sobre compases de igual medida.

2.4 Timbre

El timbre [Fig. 23] es la cualidad del sonido que permite distinguir sonidos procedentes de diferentes instrumentos, aun cuando posean igual tono e intensidad. Debido a esta misma cualidad es posible reconocer a una persona por su voz, que resulta característica de cada individuo.



Forma de las ondas de dos sonidos con el mismo tono pero distinto timbre

Fig. 23 Timbre

El timbre está relacionado con la complejidad de las ondas sonoras que llegan al oído. Pocas veces las ondas sonoras corresponden a sonidos puros, sólo los diapasones generan este tipo de sonidos, que son debidos a una sola frecuencia y representados por una onda armónica. Los instrumentos musicales, por el contrario, dan lugar a un sonido más rico que resulta de vibraciones complejas.

Cada vibración compleja puede considerarse compuesta por una serie de vibraciones armónico simples de una frecuencia y de una amplitud determinadas, cada una de las cuales, si se considerara separadamente, daría lugar a un sonido puro. Esta mezcla de tonos parciales es característica de cada instrumento y define su timbre. Debido a la analogía existente entre el mundo de la luz y el del sonido, al timbre se le denomina también color del tono.

2.5 Tono

El tono es la cualidad del sonido mediante la cual el oído le asigna un lugar en la escala musical, permitiendo por tanto, distinguir entre los graves y los agudos [Fig. 24]. La magnitud física que está asociada al tono es la frecuencia. Los sonidos percibidos como graves corresponden a frecuencias bajas, mientras que los agudos son debidos a frecuencias altas. Así el sonido más grave de una guitarra corresponde a una frecuencia de 82.4 Hz y el más agudo a 698.5 Hz.

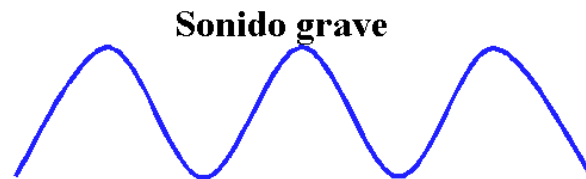


Fig. 24 Tono

Junto con la frecuencia, en la percepción sonora del tono intervienen otros factores de carácter psicológico. Así sucede por lo general que al elevar la intensidad se eleva el tono percibido para frecuencias altas y se baja para las frecuencias bajas. Entre frecuencias comprendidas entre 1 000 y 3 000 Hz el tono es relativamente independiente de la intensidad.

2.6 Ruido

Es aquel sonido molesto producido por la mezcla de ondas sonoras de distintas frecuencias y distintas amplitudes. La mezcla se produce a diferentes niveles ya que se conjugan tanto las frecuencias fundamentales como los armónicos que las acompañan.

2.7 Tipos de Ruido

2.7.1 Ruido Blanco

El ruido blanco es una señal aleatoria que se caracteriza porque sus valores de señal en dos instantes de tiempo diferentes no guardan correlación estadística. Como consecuencia de ello, su densidad espectral de potencia PSD, (Power Spectral Density) es una constante, su gráfica es plana [Fig. 25]. Esto significa que la señal contiene todas las frecuencias y todas ellas tienen la misma potencia. Igual fenómeno ocurre con la luz blanca, lo que motiva la denominación.

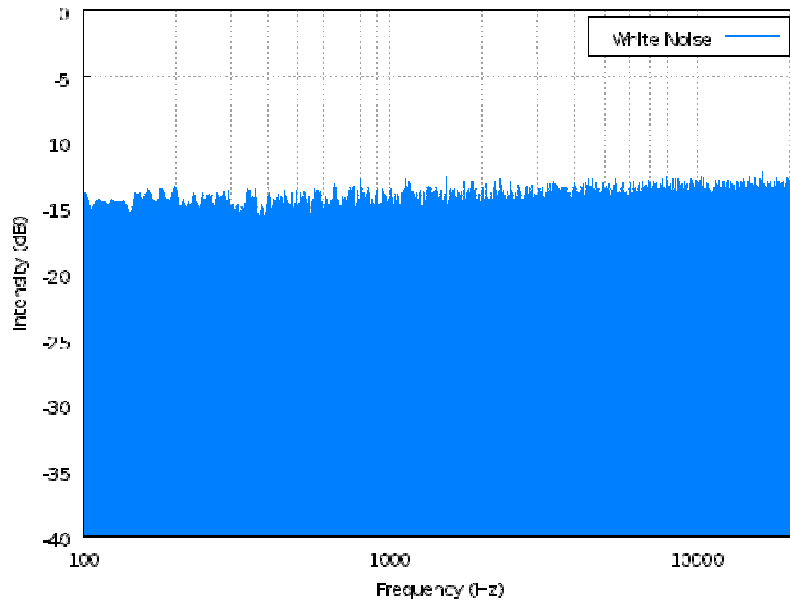


Fig. 25 Ruido Blanco

2.7.2 Ruido Rosa

Se denomina ruido rosa [Fig. 26] a una señal o un proceso con un espectro de frecuencias tal que su densidad espectral de potencia es proporcional al recíproco de su frecuencia. Su contenido de energía por frecuencia disminuye en 3 dB por octava. Esto hace que cada banda de frecuencias de igual anchura (en octavas) contenga la misma energía total. Un ejemplo de aplicación de este tipo de ruido es la obtención de respuesta en frecuencia de amplificadores de audio clase A, de manera que se reduzca el efecto de distorsión de segundo y tercer armónico que producen.

Por el contrario, el ruido blanco, que tiene la misma intensidad en todas las frecuencias, transporta más energía total por octava cuanto mayor es la frecuencia de ésta. Por ello, mientras el timbre del ruido blanco es silbante como un escape de vapor (como "Pssss..."), el ruido rosa es más apagado al oído (parecido a "Shhhh..."). Se usa mucho como señal de prueba en mediciones acústicas.

El nombre "ruido rosa" obedece a una analogía con la luz blanca (que es una mezcla de todos los colores) que, después de ser coloreada de forma que se atenúen las frecuencias más altas (los azules y violetas) resulta un predominio de las frecuencias bajas (los rojos). Así pues, el ruido rosa es ruido blanco coloreado o filtrado de manera que es más pobre en frecuencias altas (sonidos agudos).

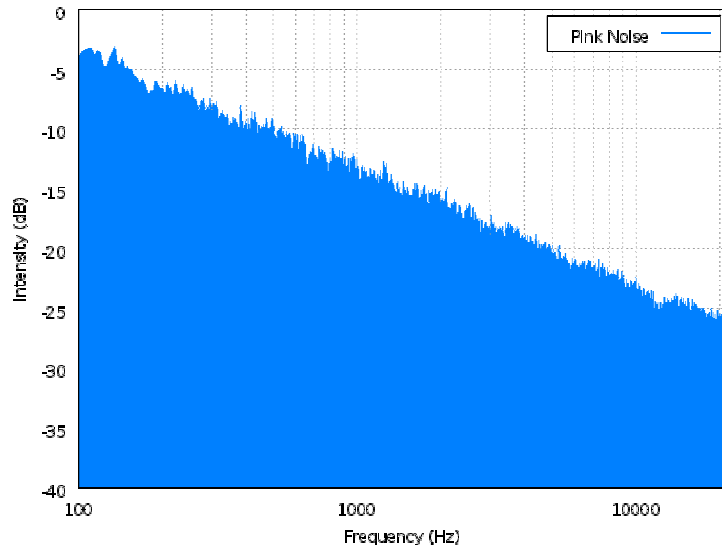


Fig. 26 Ruido Rosa

2.8 Síntesis

La síntesis, de sonido, consiste en la obtención de sonidos a partir de medios no acústicos, como variaciones de voltaje, en el caso de la síntesis analógica o por medio de programas de computadora, en el caso de la síntesis digital.

Hay quien opina que la síntesis no es el arte de crear sonidos, sino el arte de crear nuevos instrumentos, partiendo desde cero y dotándolos de sus características expresivas propias. Esto puede ser una cuestión de matiz, pero lo que sí es cierto es que cada método de síntesis posee unas características propias que lo diferencian de los demás y en consecuencia, también lo hace más o menos apropiado para generar sonido de un carácter u otro.

2.8.1 Tipos de Síntesis

2.8.1.1 Síntesis Sustractiva

Se trata del método clásico de síntesis utilizado en la inmensa mayoría de sintetizadores analógicos y en muchos sintetizadores basados en la reproducción de samples.

La síntesis sustractiva [Fig. 27] se basa en tomar un sonido (o forma de onda), preferentemente que posea un espectro muy rico (por ejemplo, formas de onda diente de sierra, cuadrada o una muestra de piano) y hacerla pasar por un filtro y

un amplificador modulables, de forma que se pueda alterar su timbre y dinámica. Es decir, normalmente se refiere a la configuración clásica “oscilador / filtro / amplificador”.

Síntesis sustractiva

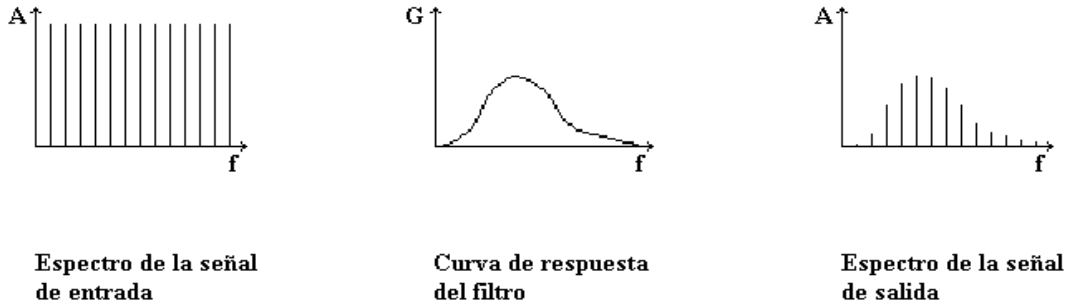


Fig. 27 Síntesis Sustractiva

2.8.1.2 Síntesis Aditiva

Cualquier sonido, por complejo que sea, puede ser descrito como la suma de un determinado número de formas de onda senoidales que lo componen, cada una de ellas con diferentes fases y amplitudes. Estos son los parciales del sonido, que también reciben el nombre de armónicos si sus frecuencias son múltiplos enteros de la frecuencia fundamental.

El método para generar el espectro de un sonido complejo por medio de la suma de un determinado número de senoidales simples se denomina síntesis de Fourier. Si las formas de onda sumadas para conseguir el sonido resultante no son senoidales, el proceso recibe el nombre genérico de síntesis aditiva [Fig. 28].

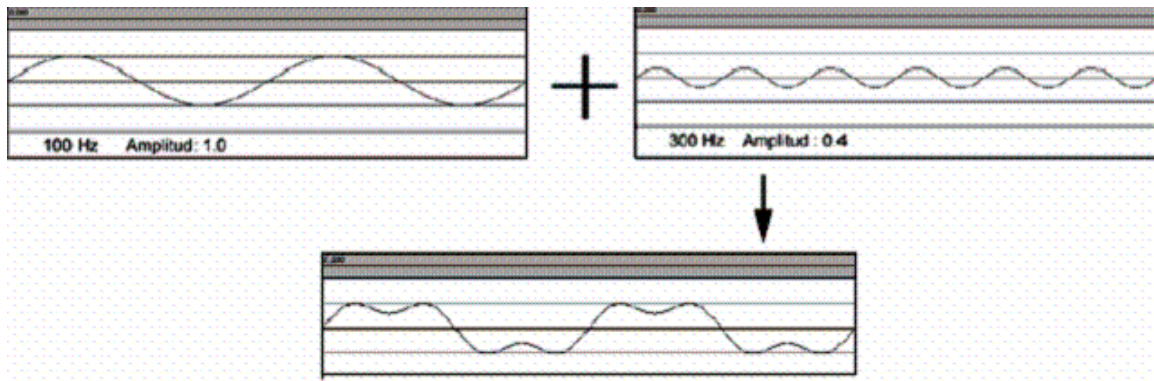


Fig. 28 Síntesis Aditiva

Idealmente, para realizar síntesis de Fourier, son necesarios muchos osciladores que entreguen senoidales; su número dependerá del rango necesario y de la riqueza armónica del sonido que se busca. Por ejemplo, una nota grave muy brillante, como la que corresponde a un bajo slap, pueden precisar más de un centenar de senoidales, mientras que un sonido de tono elevado y un contenido armónico discreto pueden conseguirse con tan sólo una docena.

Para que el sonido entregado por un sintetizador que trabaje con síntesis aditiva sea capaz de ofrecer sonidos dinámicos y ofrezca una buena capacidad de expresividad, es necesario contar con control sobre varios parámetros; idealmente, cada oscilador debe contar con su propia envolvente de amplitud, envolvente de tono, sensibilidad a la velocidad y posibilidades de modulación.

Aunque pueda parecer que las necesidades de hardware son el factor que puede limitar la popularización de sintetizadores basados en síntesis aditiva, lo cierto es que la complejidad de su programación resulta ser el factor más decisivo. Muchos de los parámetros que se manejan en un sintetizador de estas características poseen una influencia casi inapreciable, desde el punto de vista auditivo, sobre el sonido resultante, y resulta muy complicado hacerse una imagen previa de qué aspecto debe tener el espectro del sonido que estamos buscando. Por ello, la emulación de instrumentos acústicos (o reales) se hace extremadamente difícil si no se dispone del adecuado software y hardware de análisis.

2.8.1.3 Síntesis por Tabla de Ondas

Este término se utiliza para denominar dos conceptos totalmente diferentes; algunos fabricantes de tarjetas de audio para computadoras utilizan esta denominación para describir las posibilidades de reproducción de muestras almacenadas en memoria, ya que estas muestras se encuentran almacenadas en la memoria ROM en forma de una tabla.

Sin embargo, en los sintetizadores PPG Wave y Waldorf Microwave y Wave, este término se utilizó para describir la habilidad de producir un sonido realizando una secuencia a través de una tabla que contiene diferentes formas de onda; esta secuencia se realizaba durante el tiempo de duración de una nota, es decir, se utilizaba para producir un sonido, y no un patrón más o menos rítmico. Las formas de onda se almacenan en memorias ROM, aunque también es habitual disponer de una cierta cantidad de memoria RAM donde el usuario podrá cargar también sus propias formas de onda.

Una vez seleccionada la tabla de ondas, esta puede ser controlada por una envolvente o cualquier otra fuente de modulación en tiempo real. Normalmente, estos sintetizadores también son capaces de interpolar las diversas formas de

onda que componen la tabla, de forma que se suavicen las transiciones entre ellas, consiguiendo así sonidos complejos, pero con cambios tímbricos suaves.

Las formas de onda suelen ser simplemente de un ciclo, por lo que la emulación realista de sonidos acústicos suele estar fuera del alcance de este tipo de sintetizadores; siempre es mucho más fácil conseguirlo con un sintetizador que se base en la reproducción de muestras. Pero, a pesar de ello, consiguen sonidos electrónicos con un marcado carácter orgánico, ya que el sonido que entregan suele disfrutar de un gran dinamismo armónico.

2.8.1.4 Síntesis por Modulación de Frecuencias

Normalmente, la modulación en frecuencia se abrevia como FM [Fig. 29] o AFM (para Advanced Frequency Modulation). Este tipo de síntesis fue en gran medida el responsable de la gran expansión comercial de los sintetizadores digitales durante la segunda mitad de la década de los 80's. Básicamente, se trata de controlar la frecuencia de un oscilador de audio por medio de la frecuencia de otro oscilador de audio. Lo más interesante que ofrece este método es la posibilidad de generar una amplia paleta de sonidos con espectros amplios y unos transitorios de ataque muy poderoso y bien definido.

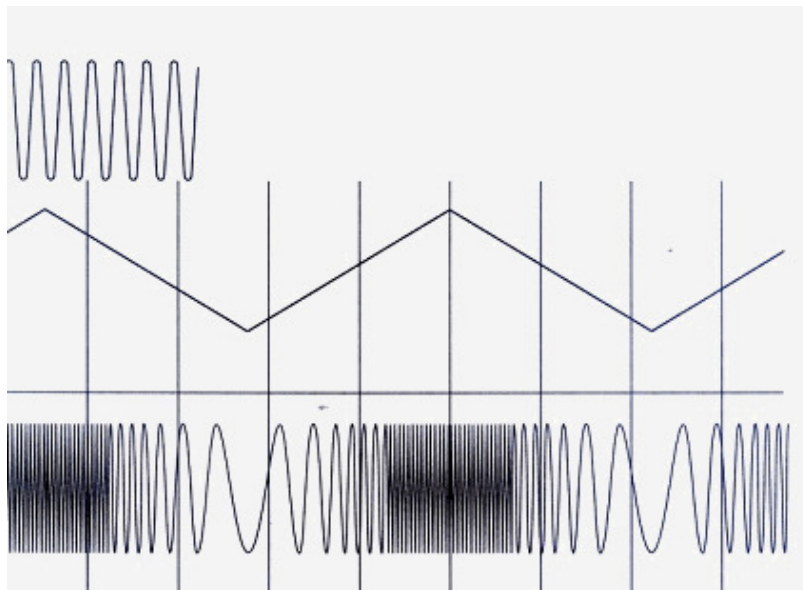


Fig. 29 Síntesis por Modulación de Frecuencias

2.9 Audio Digital

Debido a que se está utilizando Pure Data para la creación y manejo de sonidos los cuales para Pd representan solo un conjunto de números, es importante recordar de qué manera funciona el audio digital [Fig. 30].

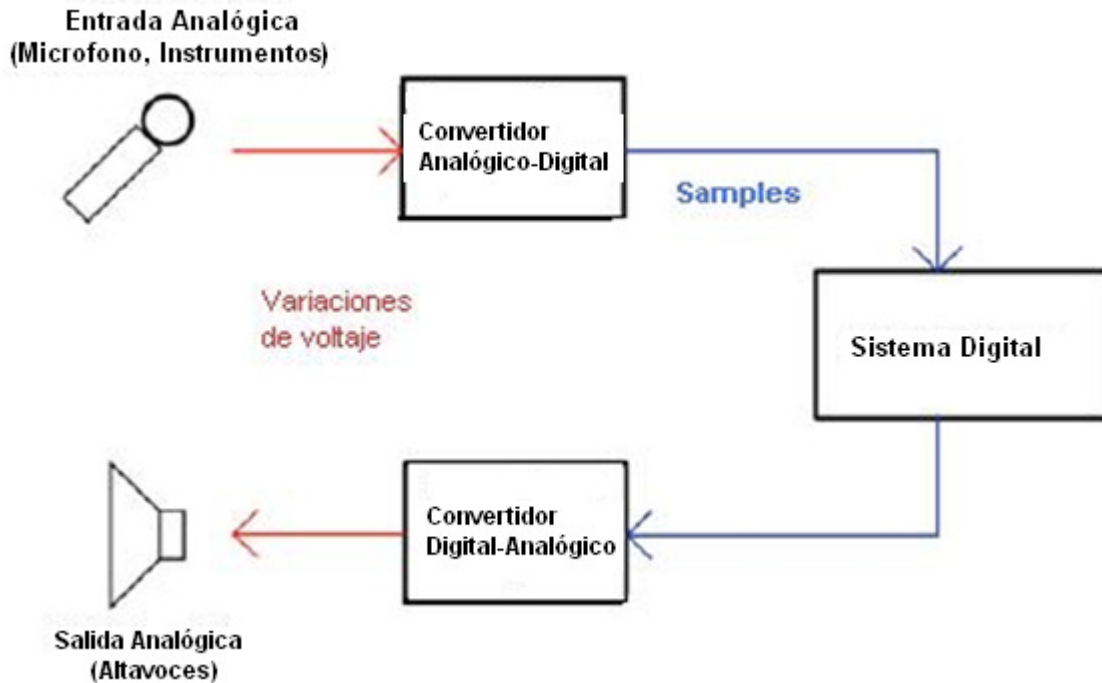


Fig. 30 Diagrama de Audio Digital

El audio digital consiste en señales de audio grabadas en forma digital, es decir, representada por ceros y unos. En cómputo, el audio digital es muy utilizado en aplicaciones multimedia mediante el uso de archivos wave audio, que son grabaciones hechas a partir de señales analógicas.

WAV es el formato estándar para almacenar archivos wave audio, puede almacenar con una resolución de 8 y 16 bits a frecuencias de muestreo de 11.025, 22.050, 44.1 y 48 kHz. WAV es compatible prácticamente con todos los programas y aplicaciones de multimedia, y con la mayoría de los editores y grabadores de audio. Los niveles de calidad están determinados por el tamaño y la frecuencia de las muestras, así como del software y hardware utilizado para la grabación.

Es importante recordar que los límites de la sensibilidad humana al sonido se sitúan entre los 20 Hz y los 20 kHz (1 kHz equivale a 1,000 vibraciones por segundo). La frecuencia de muestreo y la frecuencia de un sonido no están directamente relacionadas. Debido al Teorema de Nyquist, la frecuencia de muestreo siempre es el doble de la frecuencia de un sonido en particular.

Esta es la razón por la cual la frecuencia de muestreo utilizada para grabar un CD de audio estándar (44.1 KHz) supera ligeramente el límite superior de la capacidad humana de audición (20 KHz). En una tarjeta de sonido con límite de

frecuencia de muestreo de 12 KHz se traduce en una frecuencia de 6 KHz en sonidos reales.

En una grabación de 8 bits de resolución, el sonido es salvado en cualquier instante en particular como un valor situado en una escala que va de 0 a 255. En cambio, en una grabación de 16 bits, el sonido es registrado como un valor entre 0 y 65,536 niveles. Un oído medio puede distinguir la diferencia entre las grabaciones de audio a 8 bits y las de 16 bits. Las grabaciones a 8 bits albergan ruidos. Los realizados a 16 bits, en estéreo y a una frecuencia de 44.1 kHz presentan una grabación de mayor definición y claridad.

2.10 Caja de Ritmos

Dispositivo electrónico que permite componer, programar y reproducir patrones de ritmo mediante un secuenciador interno y un generador de sonidos de percusión [Fig. 31].



Fig. 31 Caja de Ritmos

Se basa en la programación de patrones, que son grupos limitados de compases que se reproducen de forma cíclica. Esto significa que una vez puesta en marcha, la caja de ritmos reproducirá el mismo loop hasta que no se dé la orden de pasar a otro. La programación puede realizarse en tiempo real (pulsando los botones) o bien por pasos, introduciendo las notas una a una sobre una gráfica de patrón, dividida en compases y subdividida según una cuantización prefijada por el usuario.

Los sonidos están almacenados en forma de librería y se agrupan en forma de subgrupos según su tipología (baterías acústicas, percusiones latinas, percusiones étnicas, electrónica, etc.). La capacidad de edición de éstos varía según las prestaciones del aparato, existiendo modelos muy potentes que se pueden considerar como verdaderos sintetizadores de percusión.

2.11 Secuenciador

El secuenciador es una de las herramientas básicas de la Informática musical, se trata de programas especialmente diseñados para la creación de eventos musicales. Permiten la creación de varias pistas melódicas, armónicas o rítmicas, que pueden ser tratadas, editadas y reproducidas de forma individual o simultánea.

Aparte de la afinación, duración y posición de las notas, hay muchos otros parámetros a los que los secuenciadores nos permiten acceder, tales como volumen, efectos, sonido, etc. todo ello siempre con la posibilidad de tratar no sólo la pista en conjunto sino cada una de sus notas de forma individual.

Es importante recalcar que los secuenciadores no tienen sonidos propios, sino que utilizan los de la Tarjeta de Sonido. Lo que el secuenciador hace es informar a la tarjeta de cuando debe reproducir una nota, con que volumen sonará, cuánto durará, etc. Por lo tanto la calidad del sonido no depende del Software (secuenciador) sino del Hardware (Tarjeta de Audio).

Existen infinidad de secuenciadores como Cubase [Fig. 32], Nuendo, Ableton Live e incluso algunos "especiales" para determinados tipos de música (techno, dance, etc.). Todos ellos tienen en cuenta la posible falta de conocimientos musicales del usuario, y por eso muestran la música con diferentes lenguajes gráficos que no son el estrictamente musical (Teclas de un piano, o diferentes tipos de gráficos).



Fig.32 Ventana de Cubase

2.12 Sample

Es la unidad básica con la que trabaja un sintetizador, la materia prima con la que se fabrican las notas que sonaran provenientes de un controlador. Técnicamente un sample no es más que un archivo de audio que ha sido grabado a partir de un instrumento real, y tiene asociado un número que indica la nota que es (ej. 60 es un DO).

2.13 reactIVision

Este software de visión por computadora constituye una parte muy importante en el desarrollo de **Audio Surface**, ya que este con ayuda de una cámara de video son los encargados de rastrear los objetos colocados en la superficie de la mesa.

reactIVision es una herramienta de visión por computadora diseñado por Martin Kaltenbrunner [Fig. 33a], Sergi Jordà [Fig. 33b], Günter Geiger [Fig. 33c] y Marcos Alonso [Fig.33d] en el Grupo Tecnológico Musical de la Universidad Pompeu Fabra de Barcelona, España, que permite el rápido y amplio rastreo de marcadores fiduciales adheridos a objetos, así como el rastreo de huellas. Fue diseñado como una herramienta para el desarrollo de TUI (Tangible User Interface), basadas en mesas e interfaces tangibles.



Fig. 33a Martin Kaltenbru



Fig. 33b Sergi Jordà



Fig. 33c Günther Geiger



Fig. 33d Marcos Alonso

Es una aplicación que se encarga de enviar mensajes OSC (Open Sound Control), a cualquier servidor conectado a él. Este ejecuta un protocolo llamado TUIO diseñado especialmente para transmitir la colocación y orientación de los objetos colocados en la superficie de la interfaz. Su última versión 1.4 implementa un mejor rastreo de huellas digitales, identificándolas como gotas.

reactIVision se encarga de rastrear objetos con los códigos fiduciales en tiempo real. Dicha imagen es transferida a blanco y negro mediante un algoritmo de formación de umbrales. Posteriormente esta imagen es segmentada en un árbol con regiones alternantes blancas y negras. Finalmente estas secuencias son comparadas con una biblioteca en la que cada símbolo tiene un único número de identificación. El diseño fiducial permite un cálculo eficiente de su punto central y su orientación.

2.14 Pure Data

Pd (Pure Data) es un ambiente gráfico de programación en tiempo real para el proceso de audio, en especial orientado para la creación de música electrónica. El uso de este software tiene una serie de ventajas sobre otros programas destinados al procesamiento de audio, el primero de ellos es que debido a que es un programa open source, es gratuito además de que permite que su código sea mejorada por diferentes personas y no solo por su diseñador.

Tradicionalmente, los programadores han trabajado con lenguajes de programación basados en texto. Para lidiar con este problema, Pd usa objetos gráficos que el usuario coloca y modifica en la pantalla. Estos objetos pretenden ser una analogía de los estudios donde se producía la música electrónica antes de la aparición de la computadora. Distintos dispositivos (representados por pequeñas cajas) son conectados usando cables, simbolizados físicamente por medio de las conexiones entre las cajas.

La mayor ventaja de Pd es el aspecto de “tiempo real”, en contraste con la mayoría de los ambientes de programación donde el texto era primero ingresado y después tenía que ser procesado por la computadora de forma separada para

obtener el resultado, Pd lo realiza durante la programación. Esto es similar a un instrumento clásico, el usuario escucha el resultado inmediatamente y por tanto lo puede cambiar inmediatamente.

2.15 Protocolo TUIO

Para que dos programas independientes sean capaces de trabajar de manera sincronizada y compartiendo información fue necesario utilizar un protocolo capaz de realizar dicha tarea, TUIO – este protocolo fue diseñado especialmente para trabajar en conjunto con reactIVision- es el encargado de crear el vínculo entre los dos programas utilizados para el desarrollo de **Audio Surface** (reactIVision y Pure Data).

Este protocolo permite la transmisión de una descripción abstracta de superficies interactivas, codifica información de control de una aplicación de rastreo (TUIO Tracker) y manda esta a cualquier aplicación cliente que sea capaz de decodificar dicho protocolo TUIO Client [Fig. 34]. Existe un gran número de aplicaciones de rastreo y bibliotecas TUIO para diferentes ambientes de programación, así como aplicaciones capaces de soportar dicho protocolo.

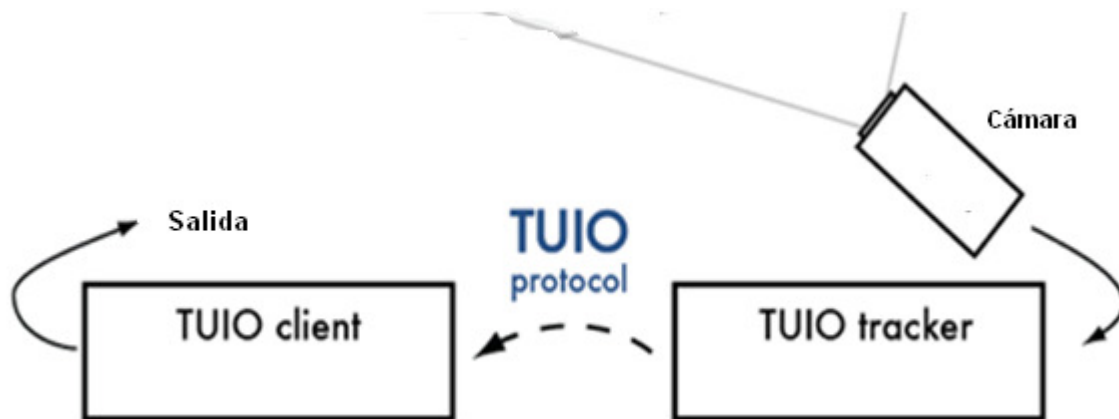


Fig.34 Diagrama TUIO

TUIO maneja dos tipos de mensajes, *set message* que está destinado para comunicar información sobre los objetos como su posición, orientación y otros estados que estos presenten y *alive message* los cuales están destinados a indicar que objeto se encuentra en la superficie usando una lista de ID's. Además los *fseq messages* están destinados para marcar de forma única cada actualización con una secuencia de identificación de marco única.

Como en todo sistema de transferencia de datos existe la posibilidad de tener pérdidas en la transmisión de la información, para evitar posibles problemas de este tipo en cuanto a la pérdida de paquetes, no se incluyen mensajes explícitos o directos de agregar y remover en el protocolo. Para evitar baja latencia en la

transmisión se transporta la información vía UDP, para evitar la pérdida de información TUIO incluye información redundante para corregir la posible pérdida de paquetes, mientras mantiene un eficiente uso del canal.

En resumen se puede decir que, los parámetros de los objetos son enviados después de que el estado de estos cambia por medio de *set message*, cuando los objetos son removidos se envía una *alive message*. El TUIO client (ver apartado 2.2.2) deduce cuando un objeto es agregado o removido por medio de los 2 tipos de mensajes y por último, los *fseq messages* asocian un ID único con un conjunto de *set* y *alive message*.

La combinación de mejoras de rastreadores TUIO, protocolos y aplicaciones cliente han permitido el rápido desarrollo de interfaces basadas en mesas táctiles. TUIO fue diseñado como una abstracción para superficies interactivas, pero también ha sido usado en muchas otras áreas relacionadas. Técnicamente TUIO está basado en OSC – estándar emergente para ambientes interactivos no solo limitado al control de instrumentos musicales electrónicos – y puede ser fácilmente implementado en cualquier plataforma que soporte OSC.

2.15.1 TUIO Trackers

La herramienta de visión por computadora es una parte fundamental para el desarrollo de **Audio Surface** ya que esta nos permitirá rastrear de manera adecuada y precisa la posición de los objetos colocados en una mesa, es por esto que fue necesario realizar una amplia investigación para así poder escoger de manera correcta esta herramienta.

Estos programas son diseñados para interactuar con superficies interactivas, como su nombre lo dice su principal función es rastrear los objetos colocados en la superficie de la mesa y transportar esta información de su colocación, orientación y ángulo por medio del protocolo TUIO a los TUIO client.

En la actualidad existen una gran cantidad de aplicaciones TUIO client, también existe una gran variedad de software de TUIO Trackers, cada uno con diferente función, como:

- reacTIVision: software de visión por computadora para rastreo de objetos.
- Ortholumen: rastreador de pluma con luz.
- LightDraw: rastreador de puntero láser.
- Touche: software gratuito para rastreo óptico de superficies de contacto táctil basado en MacOS X Leopard.

- Community Core Vision: software rastreador de contacto táctil de NUI.

2.15.2 TUIO Client

Estas aplicaciones son otra parte importante en el desarrollo de **Audio Surface**, ya que son las encargadas de decodificar los mensajes TUIO enviados por los TUIO Trackers, así mismo en estas se podrá programar las funciones que realizara cada uno de los **Knob's**.

Existe una gran variedad de lenguajes de programación para diferentes sistemas operativos, su principal función es la de decodificar los mensajes TUIO acerca de la orientación, posición y ángulo de los objetos. Una vez decodificada esta información se encargan de regresar una respuesta de acuerdo a él ID de dicha pieza y la función designada en la programación realizada. Algunos de estas aplicaciones son:

- C++ (todas las plataformas)
- Java (plataforma independiente)
- Processing (plataforma independiente)
- Pure Data (todas las plataformas)
- Max/MSP (todas las plataformas)
- Quartz Composer (Mac OSX 10.5)
- Flash (plataforma independiente)

2.16 Protocolo OSC [Open Sound Control]

Es un protocolo de comunicación entre computadoras y sintetizadores musicales y otros dispositivos multimedia. Brinda beneficios tecnológicos modernos de trabajo de red para instrumentos musicales electrónicos, entre sus ventajas incluye precisión, flexibilidad, amplia organización y documentación. Provee todo lo necesario para un control en tiempo real de sonido y demás procesos multimedia.

2.17 Trabajos Relacionados

Reactable [Fig. 35] es un instrumento musical electrónico diseñado para crear y realizar música electrónica, es una combinación de tecnología y un diseño simple y abstracto; lo que permite a los músicos experimentar con el sonido, cambiar su estructura, controlar sus parámetros. Es una interfaz tangible en la cual el músico es capaz de controlar el sistema por medio de la manipulación de objetos sobre una mesa.



Fig. 35 Reactable

Audio Pad [Fig. 36] es una interfaz que funciona en base a un secuenciador táctil basado en loops, usando samples y sonidos prefijados. Es operado mediante la manipulación de cilindros pequeños llamados *pucks*, cada uno de estos representa un loop. De igual forma cada uno de estos objetos representa diferentes funciones y son capaces de interactuar entre si, por ejemplo al mover el *puck* de modificación cerca de otro *puck*, permite la visualización de un menú. El usuario usa el *puck* de modificación para navegar en los distintos menús y aplicar efectos y cambios en los loops.



Fig. 36 Audio Pad

Audio D-Touch [Fig. 37] interfaz tangible en la cual el usuario manipula objetos, lo que le permite crear patrones y beats. Utiliza una web-cam colocada en la parte superior la cual se encarga de identificar los objetos, estos tiene impresos un código que permite que el software los reconozca. Puede funcionar de tres formas: compositor, caja de ritmos y secuenciador.



Fig. 37 Audio D-Touch

Dudtable [Fig. 38] es una mesa de mezcla musical interactiva que permite al usuario mover, golpear y mezclar sonidos para crear nueva música; está diseñado para su fácil uso donde cualquiera puede participar y mezclar. Está basada en sonidos reggae y provee una nueva forma de crear ritmos.



Fig. 38 Dubtable

Tangible Sequencer [Fig. 39], está compuesto por ocho cubos de colores diferentes, cada uno con un botón triangular que tiene la función de reproducción, este tiene que ser presionado para poder escuchar el sonido de cada uno de estos cubos. Para poder crear una melodía más larga y compleja es necesario colocar estos cubos uno delante de otro siguiendo el sentido de las flechas y estos se irán reproduciendo uno después de otro.



Fig. 39 Tangible Sequencer

Capítulo III

Desarrollo

El presente capítulo contiene los retos, metas, etapas, pruebas, problemas y soluciones que presentó el proceso de desarrollo de **Audio Surface**. Es por esto que el desarrollo se convierte en la parte más importante de un proyecto ya que nos muestra detalladamente el proceso seguido para obtener el resultado deseado.

La primera etapa para el desarrollo de **Audio Surface** comprende la configuración de la herramienta de visión por computadora (reactIVision) y nuestro TUJO Client (Pure Data). Estos dos programas son la base del funcionamiento de **Audio Surface**, ya que el primero se encarga de rastrear, idéntica y enviar la información de cada uno de los **Knob's** para que esta sea procesada y utilizada por Pure Data.

Una vez configurado y teniendo funcionando de manera correcta ambos programas, se dio paso a la segunda etapa de el desarrollo, la transmisión de información entre ellos. Debido a que Pure Data (Pd) utilizará la información enviada por reactIVision para el control de ciertos parámetros era sumamente importante que esta comunicación fuera estable.

Después de haber establecido una comunicación estable entre estos programas se dio paso a la tercera etapa, la programación y asignación de las funciones controladas por **Audio Surface**, así como la realización de pruebas de cada una de dichas tareas hasta que su funcionamiento no presentara errores.

La cuarta y última etapa consistió en el diseño y categorización de los **Knob's** de acuerdo a su tipo y/o función, así como la asignación de los parámetros que estos modificarían dependiendo de su ubicación.

3.1 Configuración de reactIVision

Para poder comenzar con el desarrollo y construcción de **Audio Surface** es importante tener un orden para llevar a cabo dicho procedimiento y completar con éxito cada una de las etapas expuestas. Como primer paso se decidió trabajar con la configuración de reactIVision (herramienta de visión por computadora) [Fig. 40], el motivo de esto es que este programa será nuestro puerto de entrada para poder controlar **Audio Surface**.



Fig. 40 Ventana de reactIVision

Como se mencionó anteriormente este programa es una herramienta de visión por computadora, esto quiere decir que su funcionamiento depende de la información captada por una cámara de video. Sus funciones principales serán la identificación, rastreo de los símbolos fiduciales y envío de dicha información a Pd, por lo que una correcta configuración es esencial para su óptimo desempeño.

reactIVision tiene la gran ventaja de que es un software open source y así mismo es compatible con los sistemas operativos más conocidos en el mercado como lo son: Windows, Mac OS X, Ubuntu, Fedora y Linux, esto lo hace ser un programa muy versátil y no limitado en su desempeño ya que sólo necesita una computadora y una cámara de video.

Para poder obtener un óptimo desempeño de esta herramienta, se realizó un proceso de configuración de parámetros, con el fin de mejorar la calidad de la imagen, evitar la distorsión y que objetos externos como sombras generen ruido visual lo que impediría tener un ágil reconocimiento de los símbolos fiduciales (la plantilla utilizada se encuentra en el Anexo B).

3.1.1 Proceso de Configuración

El proceso de configuración de reactIVision se realizó en dos partes, la primera de ellas relazada desde el código fuente del mismo donde se selecciono el patrón de símbolos fiduciales y el puerto de comunicación por donde se relazaría la transmisión de datos entre reactIVision y Pure Data.

La segunda parte de esta configuración es realizada en tiempo real, es decir no hay necesidad de realizar una compilación para ver los cambios realizados, los cuales pueden ser modificados y observados con el programa funcionando.

3.1.1.1 Elección de Patrón Fiducial y Puerto de Comunicación

reactIVision cuenta con una serie de plantillas de símbolos fiduciales las cuales fueron diseñadas especialmente para trabajar con él, cada una de estas plantillas tiene una amplia cantidad de símbolos, por lo que la correcta elección de la plantilla era primordial. El hecho de contar con una amplia gama de símbolos nos permite poder programar una cantidad de funciones directamente proporcional a dicho número.

Después de realizar pruebas con cada una de las plantillas se concluyó que la plantilla *amoeba* en su versión *small trees* presentaba un mejor funcionamiento ya que esta era reconocida con mayor velocidad y precisión por reactIVision. Esta modificación se realizó en el código fuente del programa [Fig. 41].

La elección del canal de comunicación era sumamente importante ya que estos programas estarán en constante transmisión de datos por lo que un puerto exclusivo para esta tarea presentará poca latencia. El criterio para la elección de este fue en base a recomendaciones de ambos programas.

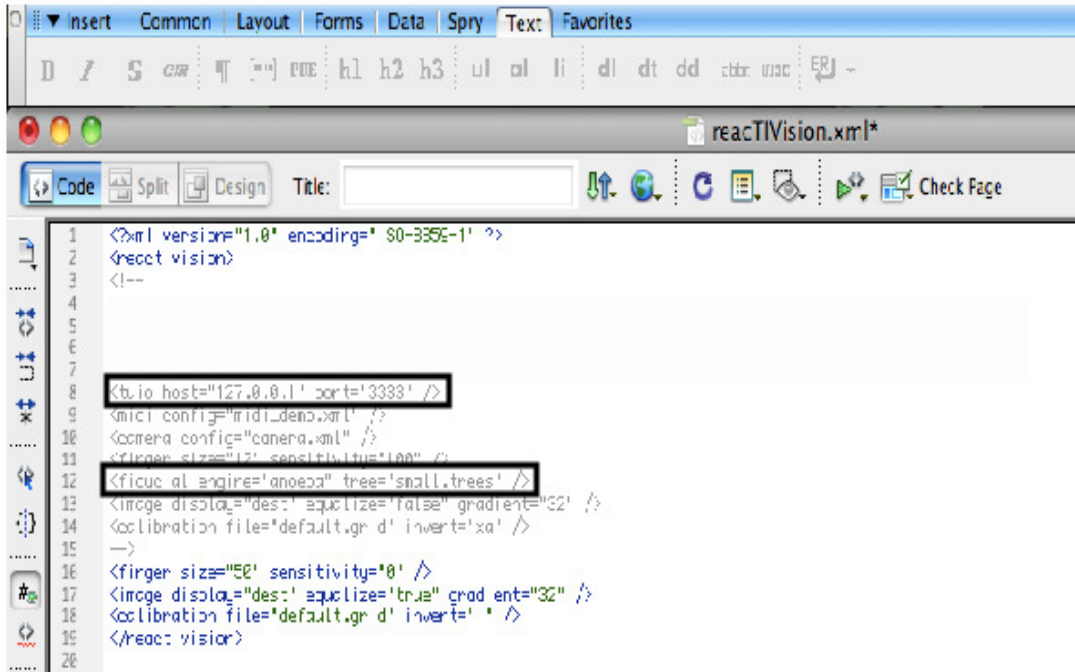


Fig. 41 Ventana de Dreamweaver

3.1.1.2 Configuración en Tiempo Real

Este tipo de configuración es mucho más práctica que la anterior ya que permite ver los cambios al momento de estarlos realizando, así mismo es posible configurar una serie de parámetros, aquí solamente se deshabilitó la sensibilidad de detección de huella digital ya que no iba a ser usada en el proyecto [Fig. 42].

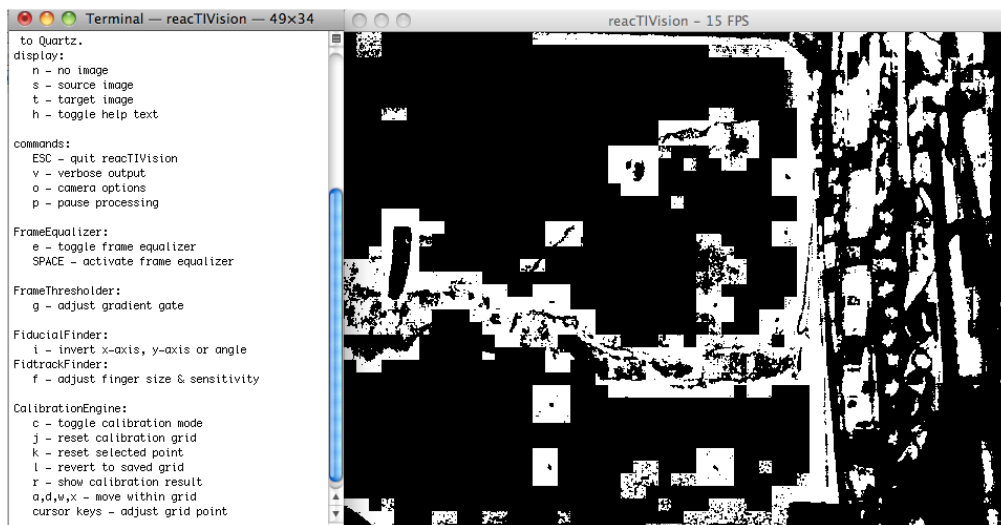


Fig. 42 Configuración Tiempo Real

3.2 Configuración de Pure Data (Pd)

Pure Data es un software libre el cual permite la creación y síntesis de sonidos, en un ambiente gráfico de programación. Lo que significa que las líneas de programación utilizadas para definir las funciones de un programa, han sido sustituidas por objetos gráficos que pueden ser manipulados en la pantalla.

El ambiente gráfico de programación que presenta Pd resulto ser más cómodo, ya que evitó la tediosa tarea de lidiar únicamente con texto, pero ésto no implica que lo haga más sencillo debido a que se deben de conocer los comandos que se necesitan, cómo funciona cada objeto y lo más importante lograr que este sea capaz de decodificar la información proveniente de reacTIVision y entregar una respuesta a esta.

3.2.1 Configuraciones de Audio

Debido a que Pure Data es un programa destinado a la composición y síntesis de música electrónica, es muy importante configurar de manera adecuada los parámetros involucrados en el manejo y control de audio digital en la computadora. El hecho de que se esté manejando un sistema de audio digital es muy importante controlar el tiempo de latencia para el procesamiento de audio y la tasa de muestreo para su calidad.

La tarjeta de audio es el elemento más importante ya que es la encargada de procesar las señales de audio tanto las de entrada como las de salida, para el desarrollo de **Audio Surface** no se buscó una tarjeta de audio externa de mejor desempeño y se usó la tarjeta de la computadora. Estos parámetros pueden ser modificados en la ventana de configuración de audio en Pd y los cambios realizados serán explicados detalladamente a continuación [Fig. 43].

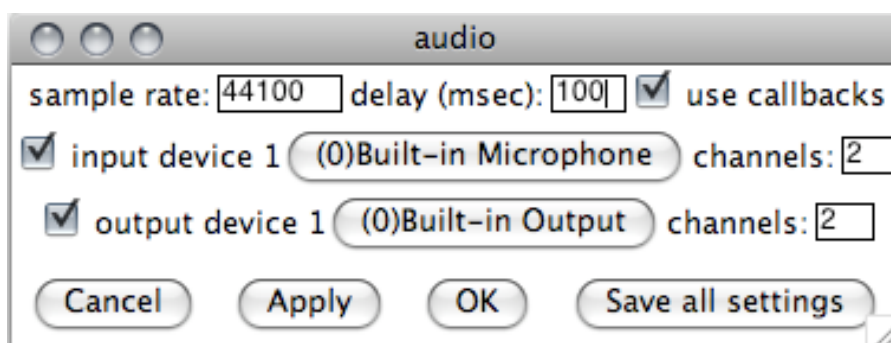


Fig. 43 Configuración de Audio

3.2.1.1 Tasa de Muestreo

La tasa de muestreo y la calidad del sonido son dos conceptos que van de la mano, ya que para obtener una buena calidad en cualquier sonido se debe de elegir de manera adecuada su tasa de muestreo. Para elegir el valor de la tasa de muestreo a utilizar nos basamos en dos puntos importantes, el *Teorema de Nyquist* y la tasa de muestreo de la tarjeta de audio de la computadora.

El *Teorema de Nyquist* nos dice que el valor de la tasa de muestreo debe de ser por lo menos el doble de la máxima frecuencia audible por el oído humano y sugiere una tasa de muestreo de 44. 100 Hz que actualmente es conocida como calidad de CD. La mayoría de las tarjetas de audio de las computadoras manejan una tasa de 44. 100 Hz y en algunos casos de 48. 000 Hz. Así mismo existen tarjetas de audio profesionales que son capaces de manejar frecuencias de muestro de hasta 96. 000 Hz.

Con esto se busca que al momento de grabar o reproducir un sonido, este contenga un amplio espectro de frecuencias y por lo tanto el espectador lo perciba como un sonido claro y agradable. Por lo que fue de gran importancia revisar la tasa de muestreo de la tarjeta de audio de la computadora y configurar este parámetro en Pd con el mismo valor.

3.2.1.2 Delay (Latencia)

Toda computadora necesita una cierta cantidad de tiempo para poder procesar la información, este tiempo es conocido como latencia y se define como la cantidad de tiempo entre el momento que se ejecuta un comando hasta que el usuario es capaz de escuchar u observar el resultado de dicha acción.

La correcta elección de este tiempo era de gran importancia para lograr que Pd tuviera una buena respuesta al ejecutar una función; una corta latencia significa que el usuario escuchará los resultados de una forma más rápida, dando la impresión de un sistema más ágil, sin embargo esto también puede ocasionar que se generen interrupciones en el audio, esto se debe a que la computadora no tiene el suficiente tiempo para “pensar” en el proceso antes de enviarlo a la tarjeta de audio.

Así mismo un tiempo de latencia muy alto daría como resultado un desfaseamiento en el audio, es decir el usuario percibiría el resultado del comando ejecutado con un retraso de tiempo, por lo que no tendría la certeza de estar escuchando su trabajo en tiempo real. Pd permite modificar este parámetro de acuerdo a las necesidades del usuario pero es altamente recomendable hacer uso del tiempo predeterminado de Pd.

3.2.1.3 Dispositivo de Entrada y Salida

Estos dispositivos son los encargados de recibir la señal enviada a Pd por medio de un micrófono y de proporcionarnos la señal de salida, por lo tanto el hecho de no configurar correctamente estos dispositivos ocasionaría que no seamos capaces de escuchar sonido alguno. Al no contar con diferentes tarjetas de audio en la computadora, automáticamente Pd busca en el sistema operativo y nos muestra la tarjeta de audio disponible; por lo que no se necesitó realizar ningún ajuste a este parámetro.

3.2.1.4 Objetos Gráficos

Los objetos gráficos son elementos muy útiles para el manejo de Pd, ya que nos permiten tener un control más interactivo de algunos parámetros dentro del *patch*. Dentro de Pd estos objetos son conocidos como GUI (Graphical User Interface) a continuación mencionaremos aquellos objetos que fueron requeridos para la programación de **Audio Surface** junto con una breve explicación de su funcionamiento [Fig. 44].

- Bang: este objeto tiene como función el enviar un mensaje llamado “bang” cada vez que se presiona, este mensaje especial es interpretado por otros objetos como “realiza esta acción ya”.
- VU: muestra el nivel de volumen promedio de cualquier señal de audio conectada a ella en decibeles, es posible modificar la escala en las propiedades del vúmetro.
- Vslider y Hslider: estos objetos gráficos son muy prácticos ya que permiten un control de un rango de valores (0-127 predeterminado) el cual puede ser configurado según las necesidades del usuario. Su manejo es muy sencillo ya que funcionan con el deslizamiento de una barra.

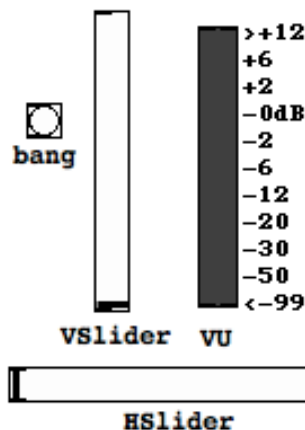


Fig. 44 Objetos Gráficos

3.3 Programación

La programación de **Audio Surface** se dividió en dos partes, la primera de ellas es la más importante ya que es aquí donde se establecerá la conexión entre reactIVision y Pd, con esto se busca que Pd sea capaz de procesar la información recibida por reactIVision al detectar los símbolos fiduciales para su decodificación y su posterior uso. La segunda parte sin ser menos importante a la primera se basa en programar las funciones que realizará **Audio Surface** y asignar cada una de ellas a los símbolos fiduciales elegidos.

3.3.1 Decodificación de reactIVision

La decodificación de reactIVision es la primer y más importante parte de la programación de **Audio Surface**, ya que con esto se busca obtener una sincronización entre el software utilizado para su desarrollo (reactIVision y Pd). Este programa o mejor dicho *patch* (que es el nombre técnico utilizado en Pd) es la base del funcionamiento de **Audio Surface**, ya que tiene la tarea de recibir al información enviada por reactIVision acerca del ID y posición de cada uno de los **Knob's**, para ser utilizada posteriormente.

Junto con reactIVision se descargó una carpeta, la cual contenía una serie de *patches* de Pd, aquí se encontró un *patch* llamado *TuioClient-help.pd* [Fig. 45]. Como su nombre lo dice, este un *patch* de ayuda o guía que permite a Pd trabajar como un TUIO Client, por lo tanto Pd es capaz de recibir y procesar la información enviada por reactIVision sobre el ID y la posición de cada **Knob**.

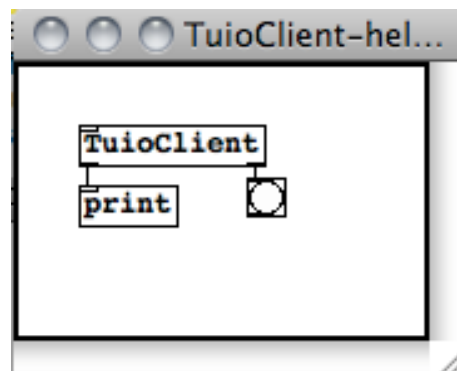


Fig. 45 patch Guía

A pesar de la simplicidad que demuestra este *patch*, su funcionamiento fue muy importante para el desarrollo de **Audio Surface** porque este cumple con el primer objetivo. El objeto TuioClient es el encargado de establecer una comunicación entre reactIVision y Pure Data; permitiendo a Pd recibir los mensajes TUIO enviados por reactIVision acerca de la posición e ID de los **Knob's** a través del puerto asignado.

Como se observa en la figura anterior el objeto Tuio Client tiene dos objetos conectados a él, uno en cada una de sus *outlets* (es el termino técnico utilizado en Pd para referirse a las salidas de un objeto). El primero de ellos (outlet derecho) es un objeto gráfico que tiene por nombre *bang* y tiene la función de enviar un pulso (o *bang* de donde recibe su nombre) con un período de 1 segundo con la finalidad de indicar que la conexión entre ambos programas esta activa.

El segundo objeto colocado a la izquierda contiene la instrucción *print*, se encarga de imprimir en la ventana principal de Pd los mensajes TUIO que reciba con la información sobre la posición, ID y el momento en que el objeto fue agregado o retirado (addObject: ID, (x,y), ángulo; updateObject: ID, (x,y), ángulo, este actualiza la información constantemente de los objetos ya detectados; removeObject: ID del objeto retirado).

Después de haber comprendido el funcionamiento del *patch* se realizó una prueba [Fig. 46] con el motivo de corroborar que su funcionamiento fuera el adecuado, es decir que los mensajes TUIO enviados por reactIVision fueran recibidos por Pd. Al realizar dicha prueba se vió que efectivamente Pd estaba recibiendo los mensajes TUIO de reactIVision sin ningún problema, pero esta solo era desplegada en su ventana principal y no podía ser utilizada de forma diferente; en otras palabras este *patch* solo era informativo y no funcional.

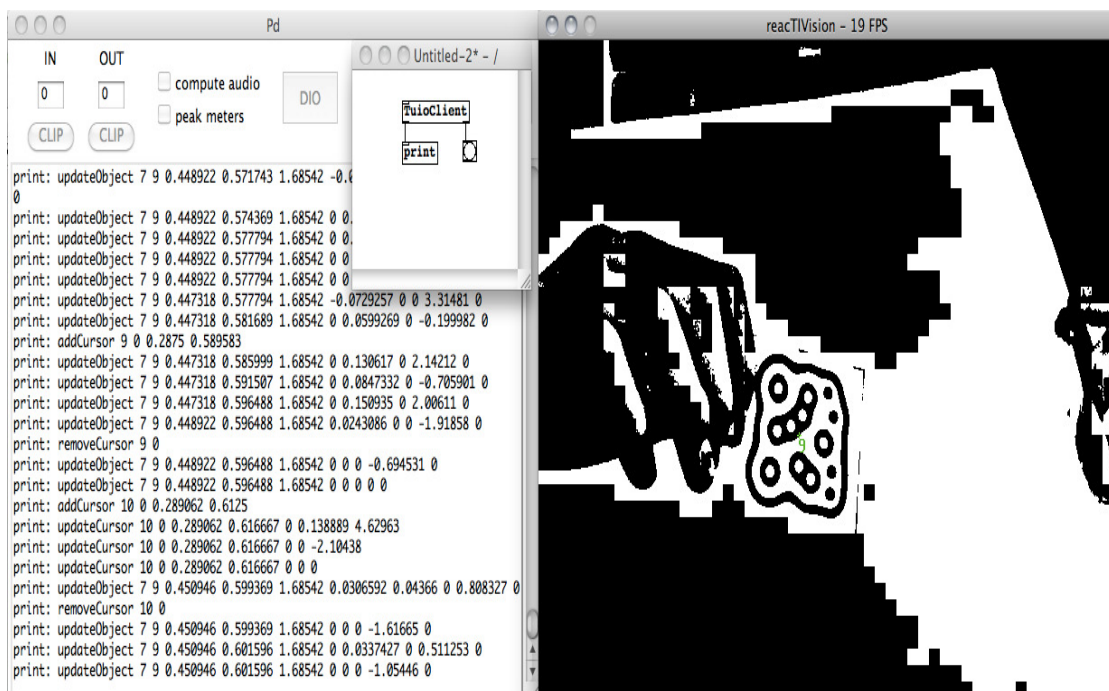


Fig. 46 Primer Prueba

Por lo que fué necesario modificar este *patch* para filtrar los mensajes TUIO, es decir que el despliegue de estos no sólo fuera informativo y excesivo, si no que dicha información fuera separada para poder ser utilizada para modificar parámetros en otros *patches*.

En el segundo *patch* [Fig. 47] se puede observar que es más complejo que el anterior, ya que cuenta con objetos nuevos y por lo tanto realiza funciones que el otro no. A diferencia del primer *patch* aquí el objeto *print* no se encuentra conectado directamente a Tuio Client en su lugar tiene un objeto con la instrucción *route* y tres variables.

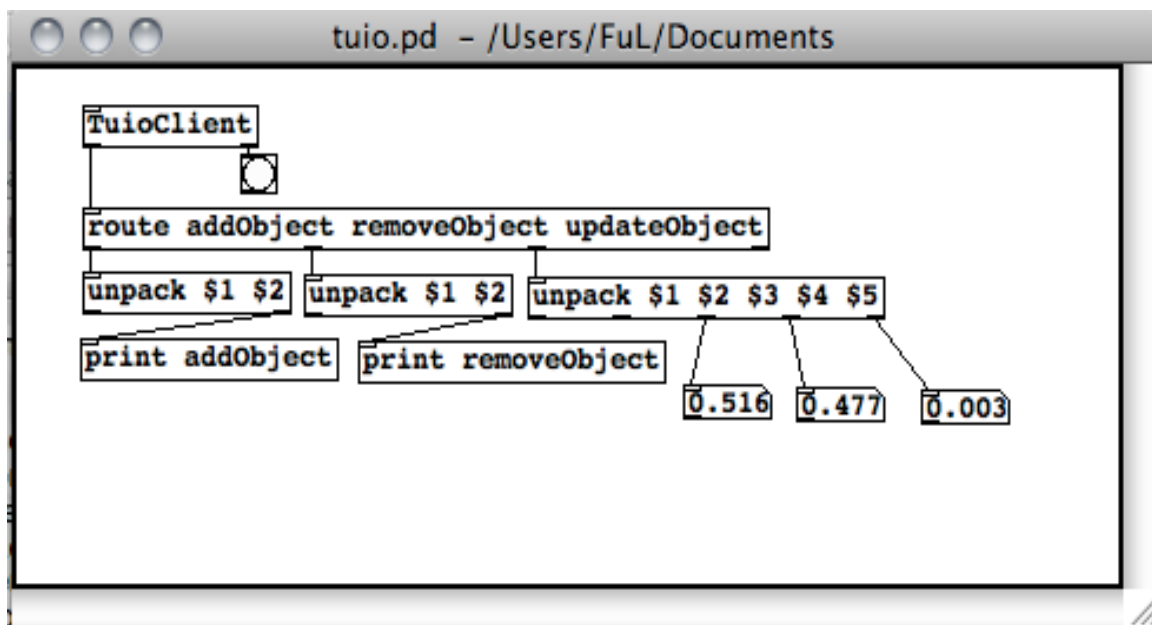


Fig. 47 Segundo “patch”

Estas tres variables (*addObject*: indica cuando el objeto fue agregado; *removeObject*: cuando fue retirado y *updateObject*: despliega constantemente la información sobre la posición del spot) están contenidas en el mensaje TUIO que es enviado por *reactIVision* y cada una tiene información específica sobre los **Knob's**; el objeto *route* permite separar estas variables internas para utilizar su información de manera independiente.

Los objetos conectados a las salidas de *route* (*unpack*) tiene la función de desglosar la información de cada una de las variables, ya que a su vez estas variables tienen sus propias variables internas con información sobre los **Knob's**. El primero de ellos ubicado en la extrema izquierda es el encargado de desplegar en la ventana principal de Pd le mensaje *addobject* acompañado del ID del objeto agregado.

El objeto de la parte de enmedio tiene una función muy similar ya que también despliega en la ventana principal de Pd el mensaje *remove Object* seguido del

ID del objeto removido. A diferencia del primer *patch* que desplegaba toda la información de los mensajes TUIO en la ventana de Pd de una forma rápida que no permitía distinguirla, en este segundo *patch* se muestra menos información en dicha ventana lo que da como resultado un mejor manejo de la información.

Por último el objeto que se encuentra en la extrema derecha es el más importante de los 3 ya que es el que se encarga de mostrar en una caja numérica la posición del **Knob** en las coordenadas (x,y) y su ángulo, en tiempo real, es decir esta información irá cambiando conforme se mueva el **Knob**

Esto presenta una gran ventaja y un considerable avance con respecto al primer *patch*, ya que en primer lugar, en la ventana principal de Pd sólo se despliega la información sobre el ID del objeto agregado y/o removido; por lo que no se tiene un exceso de información. Por otro lado se avanzó considerablemente ya que el despliegue de la información se presentó en 3 cajas numéricas y este es en tiempo real. Por lo que esta puede ser utilizada para controlar ciertos parámetros.

Sin embargo, al realizar las pruebas correspondientes sobre el funcionamiento del *patch* [fig. 48] se notaron dos cosas; en primer lugar, se comprobó que el despliegue de la información sobre la posición de los **Knob's** era más ordenado, pero así mismo surgió un problema, ya que al momento de colocar dos **Knob's** al mismo tiempo la información de estos se desplegaba en la misma caja numérica. Por lo que para el funcionamiento de **Audio Surface** esto presentaba un gran problema.

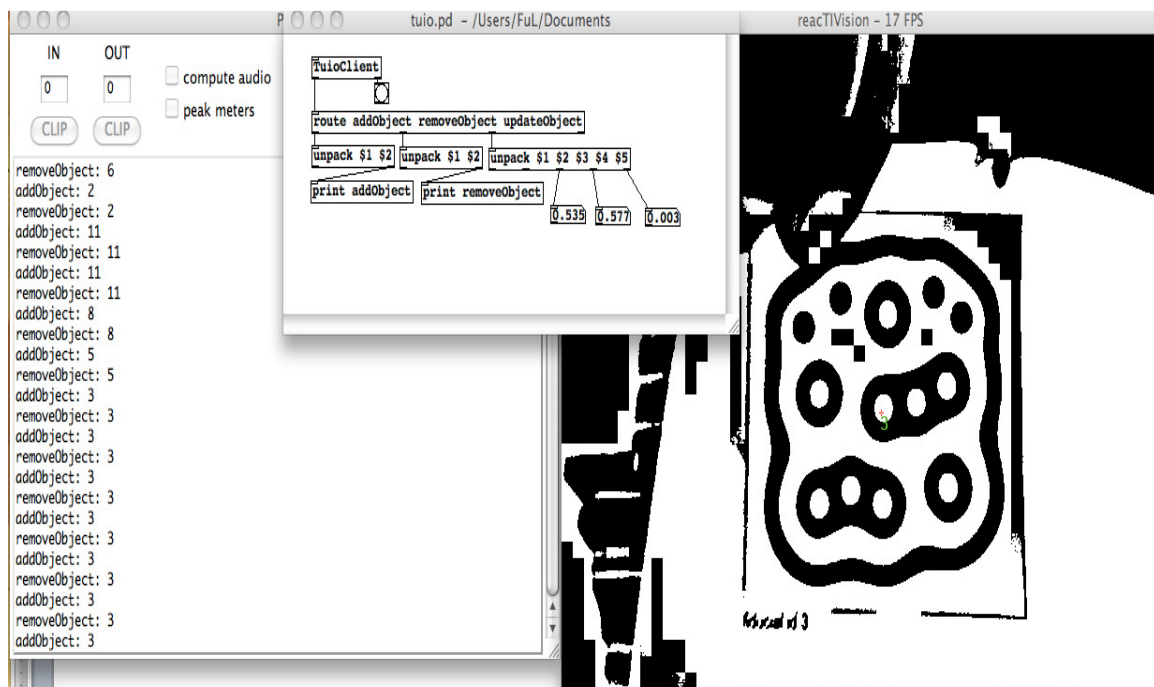


Fig. 48 Segunda Prueba

Por lo que en último *patch* se buscó filtrar esta información aun más, es decir, buscar el despliegue de la información de la posición de cada **Knob** independientemente para que de esta forma cada **Knob** pudiera ser asignado a controlar una función en específico, como modificación de volumen, cambio de pitch, los parámetros de un efecto, etc.

Para cumplir con este objetivo se utilizaron nuevos comandos, en conjunto con los alive message contenidos en los mensajes TUIO ya explicados (removeObject y addObject) que son enviados por reactIVision [Fig. 49], los cuales serán más selectivos al momento de reconocer un símbolo fiducial en la cámara de video y disparan la función asignada a este. Esto se consiguió con ayuda de dos nuevos comandos: el primero de ellos \$2-tuioObjectAlive que tiene como función decodificar el ID del símbolo fiducial y enviar un “1” cuando es reconocido por reactIVision y un “0” cuando es removido.

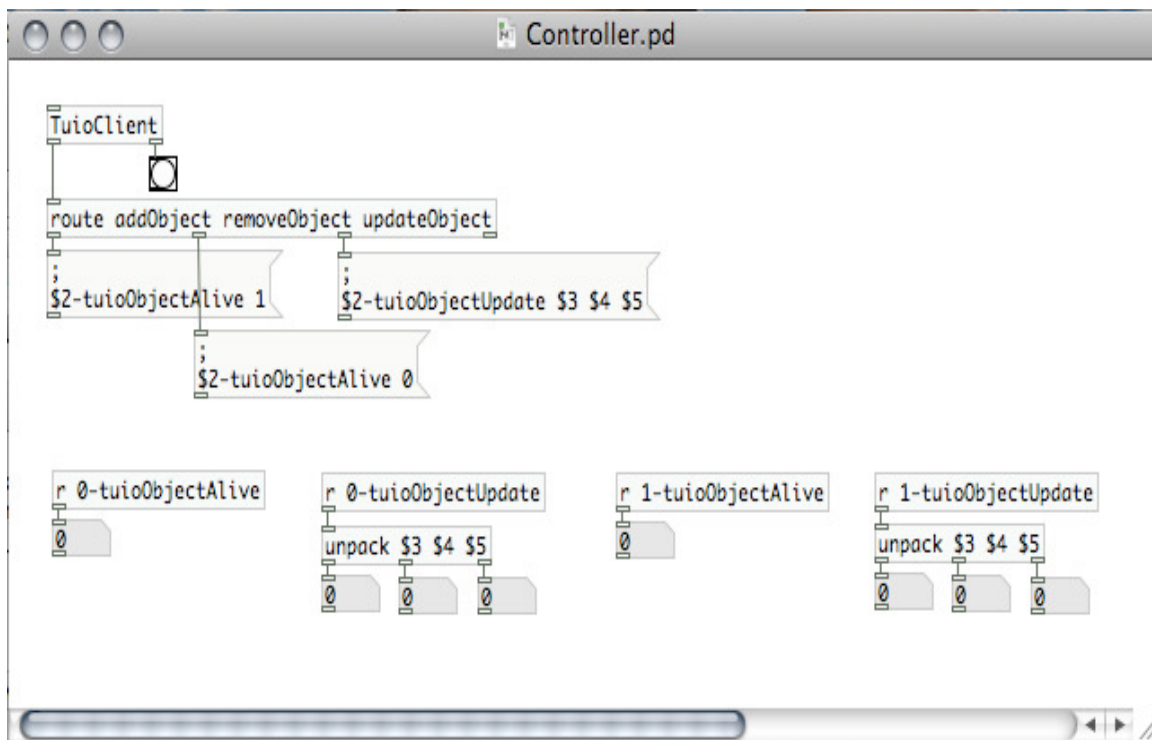


Fig. 49 Tercer *patch*

Esta información es enviada a su contraparte `r x-tuioObjectAlive` que tiene la función de recibir la información enviada por `$2-tuioObjectAlive` y desplegarla en la caja numérica conectada a él; con esto se buscó dejar de utilizar la ventana principal de Pd para el despliegue de información.

El comando `$2-tuioObjectUpdate` sigue realizando la misma función de decodificar la posición y el ángulo de los **Knob's**, sin embargo tiene algunas diferencias a el utilizado el *patch* anterior, la primera es que a diferencia del

anterior este se encuentra en una caja de mensaje, con el propósito de hacer el envío de los datos de manera “inalámbrica”. La segunda diferencia es que se tiene el sufijo \$2, el cual es utilizado como una variable la cual tomará el valor del ID detectado por reactIVision y enviará la información sobre su posición.

Esta información es enviada al objeto `r x-tuioObjectUpdate`, donde se decodificará la información de la posición y se desplegará en tres diferentes cajas numéricas de la misma forma que el *patch* anterior. Como se puede ver en el *patch* se tienen que crear tantos objetos como símbolos fiduciales vayan a ser utilizados, para **Audio Surface** solo se trabajará en un principio con los primeros 9 símbolos fiduciales.

Al realizar la evaluación de este *patch* [fig. 50], se comprobó que el reconocimiento de los símbolos fiduciales así como su posición eran completamente independientes. Por lo que esta información podía ser utilizada para modificar los parámetros que les serán asignados para el desarrollo de **Audio Surface**.



Fig. 50 Tercer Prueba

Una vez completando el objetivo de la sincronización y transferencia de datos entre reactIVision y Pure Data de manera exitosa, se emprendió un nuevo camino para cumplir de la misma forma un nuevo objetivo. Este era el de

programar las funciones de **Audio Surface**, como reproductor de samples, efectos, etc.

Se tomó la decisión de comenzar la programación con el reproductor de samples, ya que la parte principal para la mezcla musical es contar con reproductor que nos permita escuchar los archivos de audio con los que se cuenta.

3.3.2 Reproductor de Samples

Este *patch* [Fig. 51] tiene como objetivo el de reproducir un archivo de audio repetidamente, es decir hacer un loop infinito de este. Una vez logrado realizar esto se asignaron ciertos parámetros a los **Knob's** designados a cumplir esta función con el objetivo de que todo sea controlado por medio de la manipulación de los **Knob's**. Los parámetros asignados fueron: control de volumen y play/stop.

Como se mencionó anteriormente la programación de este *patch* se dividió en dos partes, la primera parte consiste en abrir y reproducir un archivo de audio en Pure Data. Cuando el archivo de audio es ubicado dentro del disco duro de la computadora este es cargado, su forma de onda es desplegada dentro del arreglo y se empieza a reproducir en forma de loop infinito.

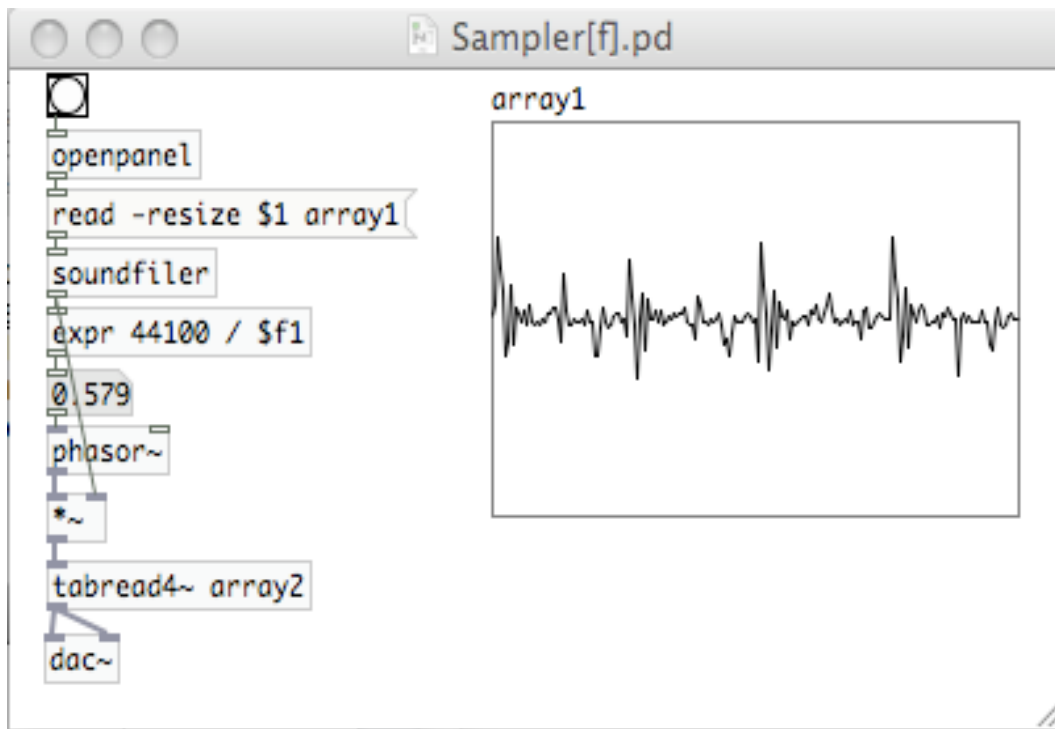


Fig. 51 Reproductor de Samples

3.3.3 Velocidad de Reproducción

Después de haber completado exitosamente el reproductor de samples se le hizo una variación que permitiera al usuario tener más control sobre el sample en reproducción. Por lo que se agregó otro parámetro a modificar el cual es de un uso fundamental en el ambiente del dj, el cambio de pitch en un track o de la velocidad de reproducción [fig. 52]. Con esto se busca que el usuario cuente con funciones similares a las que le ofrece un CD player o tornamesa.

Al igual que el *patch* anterior este nos permite manipular por medio del **Knob** el nivel de volumen del archivo de audio y play/stop. Y se añadió uno nuevo como se mencionó anteriormente, que es el control de pitch.

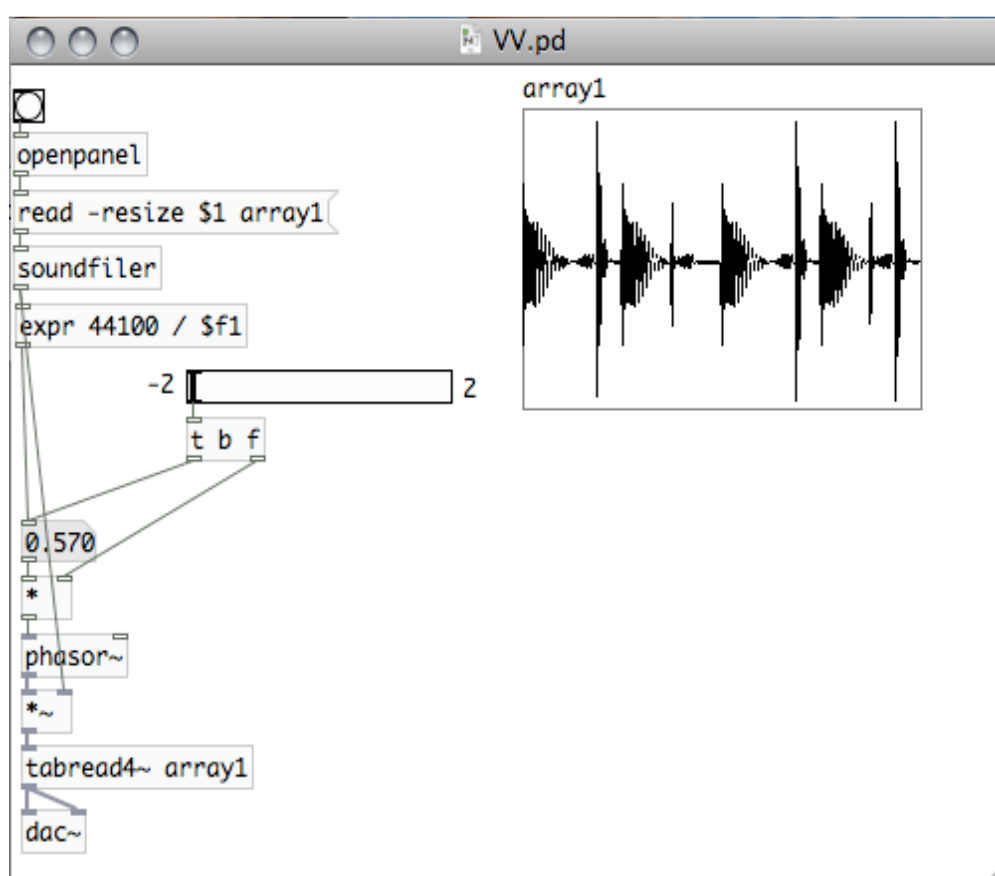


Fig. 52 Cambio de Velocidad de Reproducción de un Sample

3.3.4 Scratch

El uso de las tornamesa le permitió al Dj crear variaciones al sonido, el scratch [Fig. 53] fue una de las primeras técnicas utilizadas por los Dj para lograr esto. Es por esto que para que **Audio Surface** fuera capaz de brindar al DJ una

herramienta completa con funciones similares a las utilizadas ya sea en CD players o tornamesas, pero que a la vez le brindaran un mayor dinamismo comparado con el que le brindaban dichos equipos.

La función de este *patch* es producir un sonido similar al scratch, el cual se produce al mover un disco en la tornamesa hacia atrás y hacia delante provocando un efecto parecido al de rayar un disco. Así mismo este trabajará en conjunto con el reproductor de samples, es decir será un efecto que se le añadirá a un sample en reproducción si el usuario así lo requiere.

Este *patch* no manejará un archivo de audio de forma independiente como los *patch* anteriores, para el control de este *patch* por medio de los **Knob's** se asignaron tres parámetros a ser controlados: encendido/ apagado, el volumen del efecto dentro de la mezcla, el control del efecto de forma similar a como un dj lo haría en un CD player o tornamesa.

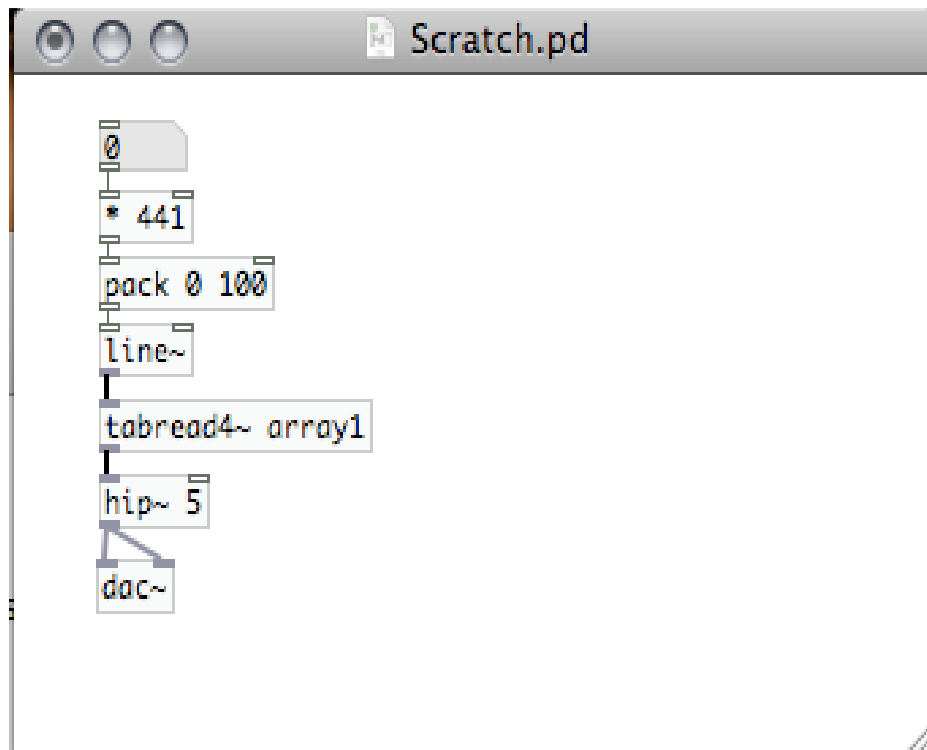


Fig. 53 Scratch

3.3.5 Delay

En la de producción de música electrónica los efectos juegan un papel muy importante, ya que ayudan a darle la “textura” electrónica a la pieza musical, por lo que se decidió programar otro efecto para poder brindar a los usuarios las herramientas necesarias para realizar su composición. El hecho de que para el

desarrollo de este proyecto sólo se cuenten con dos efectos no significa que estos serán los únicos con los que **Audio Surface** contará.

Al igual que en el *patch* del *scratch* este *patch* tampoco manejará un archivo de audio de forma independiente y será agregado a la señal de audio que manejan los *patches* anteriores. Esto se hace para que el efecto poder ser utilizado con más de un solo *sample* al mismo tiempo y no tenga que ser exclusivamente asignado a uno solo.

Cuando el usuario active este efecto la señal original se “duplicará” ya que se escuchará una suma de dos señales (señal original + señal procesada por el efecto). El proceso del efecto sobre la señal nos permite retrasar esta señal cierta cantidad de milisegundos para lograr un efecto de eco. Los parámetros que serán controlados por medio del **Knob** asignado a este efecto son: tiempo del *Delay* y el volumen del efecto [Fig. 54].

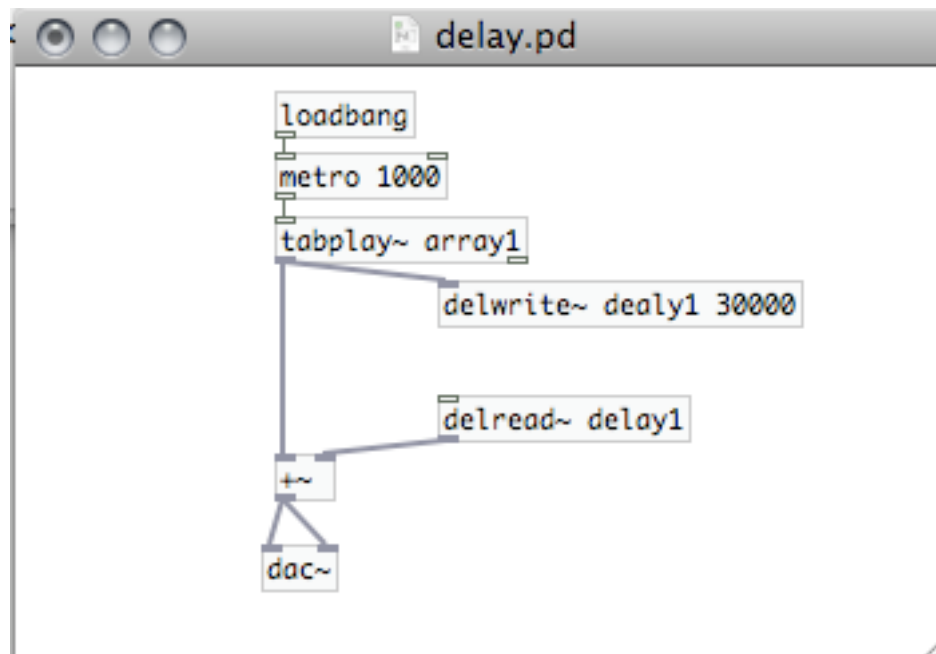


Fig. 54 Delay

3.3.6 Descripción del *patch* Final

Una vez que se programaron los *patch* que se necesitaban para el desarrollo de **Audio Surface**, estos tuvieron que ser agrupados en un nuevo *patch* con el motivo de tener un solo programa que tenga la facultad de realizar todas las funciones programadas. Para completar este objetivo se hizo uso de los *subpatch*, los cuales son subprogramas que crean nuevas ventanas para tener

un mejor manejo de la información y ayudan al CPU a dividir los procesos y por lo tanto a tener un mejor desempeño.

La ventana principal [Fig. 55] del *patch* final presentara muy poca información, debido a que todos los programas y la forma en que estos serán manipulados y activados por medio de la información enviada por reactIVision estará contenida en los *subpatch* y cada uno de ellos tendrá una función en específico.

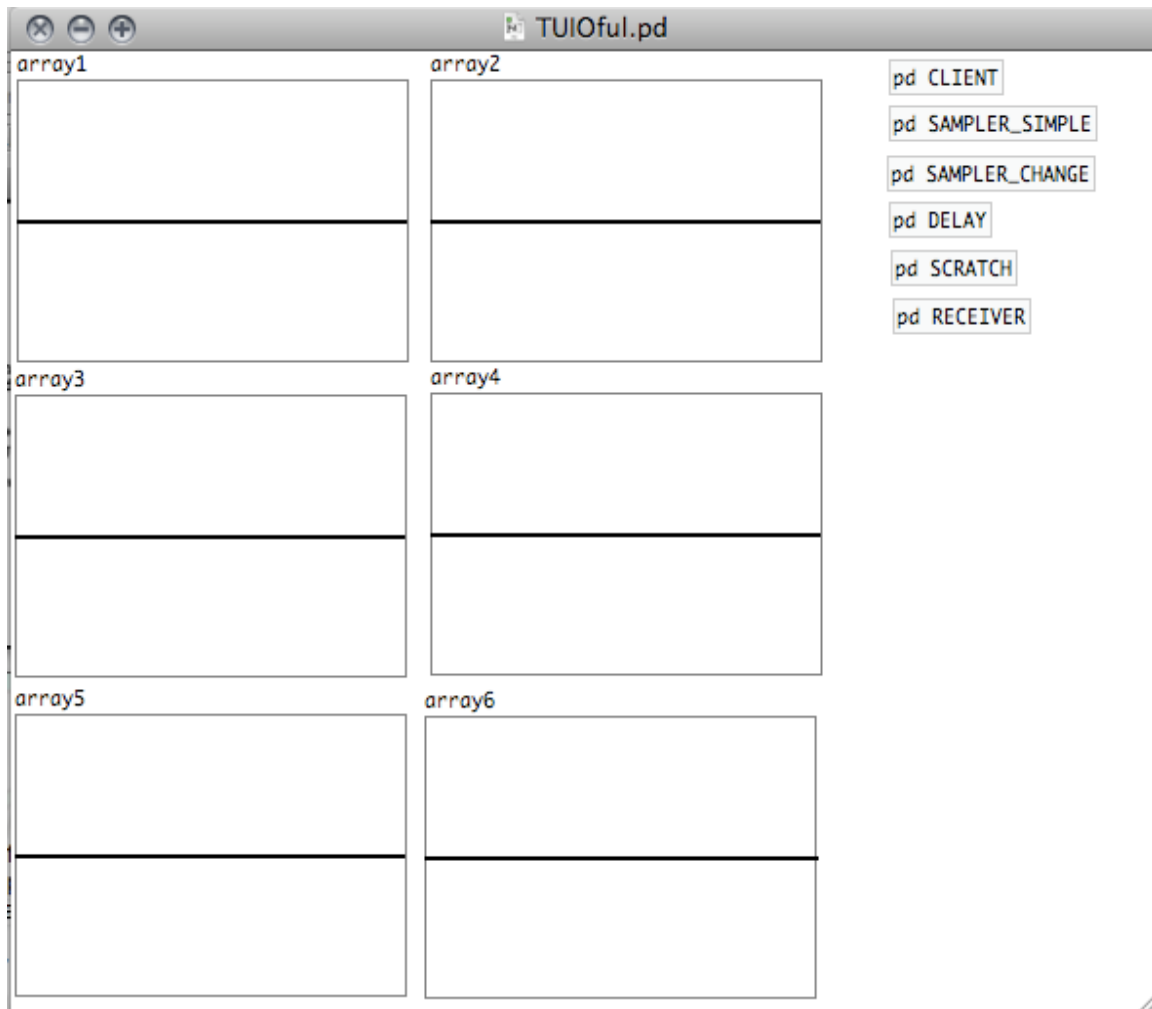


Fig. 55 patch Final

A la derecha de esta ventana se observan seis rectángulos con distintos nombres, estos son los *subpatch* del programa, cada uno de ellos tendrá una función en específico para el funcionamiento de **Audio Surface**, y para ver el contenido de cada uno de ellos es necesario dar un click sobre este y se desplegará una ventana con el programa de cada *subpatch*.

Las funciones de estos *subpatch* son las siguientes:

- pd CLIENT: se encarga de decodificar la información enviada por reactIVision (apartado 3.3.1).

- pd SAMPLER_SIMPLE: su función es reproducir tres samples que serán representados en los arreglos 1-3 (apartado 3.3.2).
- pd SAMPLER_CHANGE: reproduce tres samples que estarán en los arreglos 4-6, con la diferencia de que este permitirá poder modificar la velocidad de reproducción de dicho sample (apartado 3.3.3).
- pd DELAY: este efecto podrá ser agregado a los tres primeros samples (apartado 3.3.5).
- pd SCRATCH: al igual que el “Delay” este es otro efecto y de igual manera solo se podrá utilizar con los tres primeros samples (apartado 3.3.4).
- pd RECEIVER: este “subpatch” tiene una función informática ya que se encargará de mostrar cuando un objeto sea agregado o removido.

3.4 Knob's

Como se mencionó anteriormente **Audio Surface** será una interfaz tangible, esto quiere decir que el usuario será capaz de modificar o controlar funciones y parámetros por medio de una serie de objetos a los que llamaremos **Knob's**. Estos estarán divididos de acuerdo a la función que se les asignó previamente, así mismo algunos de ellos tendrán la capacidad de trabajar en conjunto o modificar a otro **Knob**.

Estos **Knob's**[Fig. 56] estarán fabricados de acrílico transparente con una dimensión de 6 cm x cm x 1.2 cm y cada uno de ellos contará con dos calcomanías, de un lado con el símbolo fiducial y de el lado contrario un símbolo que defina su función. Los símbolos ocupados para definir la función de cada uno de los **Knob's** fueron diseñados cuidadosamente con el motivo de conseguir que el usuario al momento de ver dicho símbolo lo relacione inmediatamente con la función asignada.



Fig. 56 Knob's

La selección de los símbolos fiduciales para **Audio Surface** fue realizada de acuerdo a la posición de estos en la lista de símbolos, por lo que se seleccionaron los primeros nuevos símbolos de dicha lista con los ID's 0-8; así mismo estos fueron dividido de acuerdo a su función asignada.

3.4.1 Control D On/Off (ID 0)

El primer **Knob** (ID = 0) [Fig. 57] es el más importante para el funcionamiento de **Audio Surface**, ya que su función es el encendido/apagado del sistema, es decir al momento de ser detectado por reactIVision, este mandara un mensaje a Pd con el valor de "1". Este valor será desplegado en una caja numérica la cual está conectada a la salida principal de audio, por lo que al recibir este valor el usuario será capaz de escuchar los sonidos producido.

Caso contrario sucede al retirar dicho **Knob** del campo de visión de la cámara de video, ya que ahora reactIVision mandará un mensaje con el valor de "0" por lo que ningún sonido será escuchado por el usuario (todos el **Knob's** utilizados en **Audio Surface** contarán con una función de on/off independiente que funciona de la misma forma que el on/off principal).

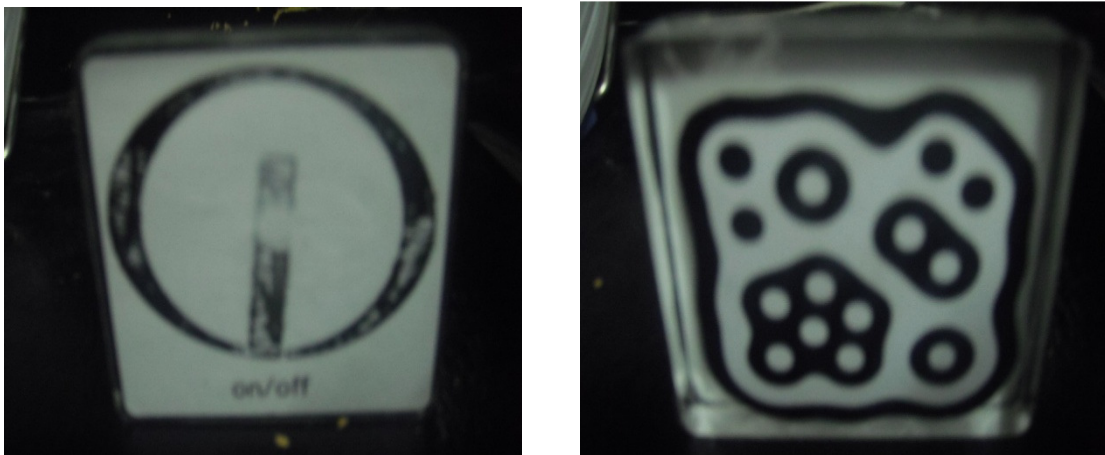


Fig. 57 "on/off" ID 0

Por lo tanto este **Knob** tiene que estar colocado de manera permanente bajo el campo de visión de la cámara de video y tiene que ser reconocido de manera adecuada por reactIVision para que Pd sea capaz de procesar dicha información y reproducir los sonidos asignados a los demás **Knob's**.

3.4.2 Control del Sampler (ID 1-3)

Este grupo de **Knob's** [Fig. 58] cumple una función básica pero fundamental para el funcionamiento de **Audio Surface**, ya que son los encargados de reproducir un sample. Su funcionamiento está ligado a el *subpatch* "SAMPLER_SIMPLE".

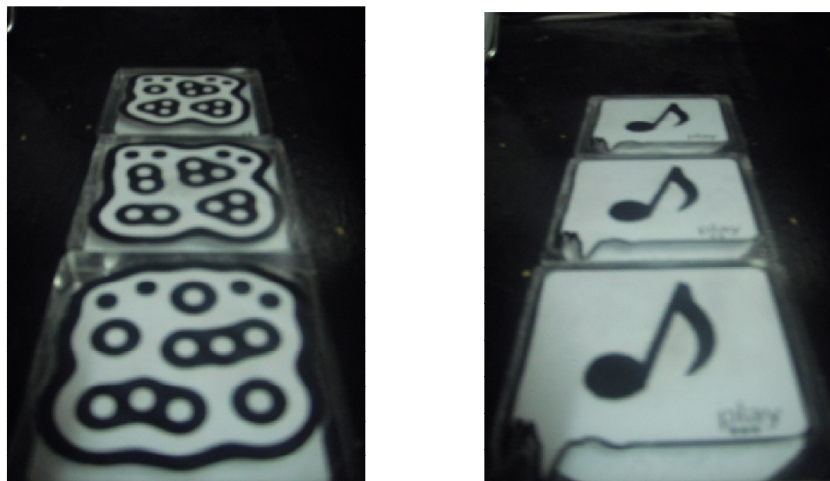


Fig. 58 Sampler Knob's

Como se mencionó anteriormente a este grupo de **Knob's** se les asignaron dos funciones, control de volumen y play/stop del archivo de audio, una vez que este sea reconocido exitosamente por reactIVision inmediatamente el usuario podrá escuchar dicho archivo. Esta reproducción creará un loop, es decir el archivo se repetirá una y otra vez hasta que el usuario lo retire de la superficie de **Audio Surface**.

El control de volumen se realizará con la información de el ángulo del **Knob** en la superficie de **Audio Surface** y se modificará al girar el **Knob** hacia la derecha (mayor volumen) o la izquierda (menor volumen), esto con el motivo de brindarle al usuario un control parecido al que tendría en una perilla [fig. 59].

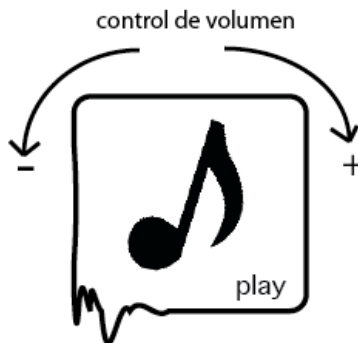


Fig. 59 Manejo del Knob

3.4.3 Control de Velocidad de Reproducción (ID 4-6)

SAMPLER_CHANGE [Fig. 60] está compuesta por tres **Knob's** que tienen un funcionamiento muy parecido al grupo anterior de **Knob's**. Cada uno de los **Knob's** está vinculado a un sample y su reproducción depende de que el ID del **Knob** sea reconocido por reactIVision ya que de lo contrario, este no será reproducido.

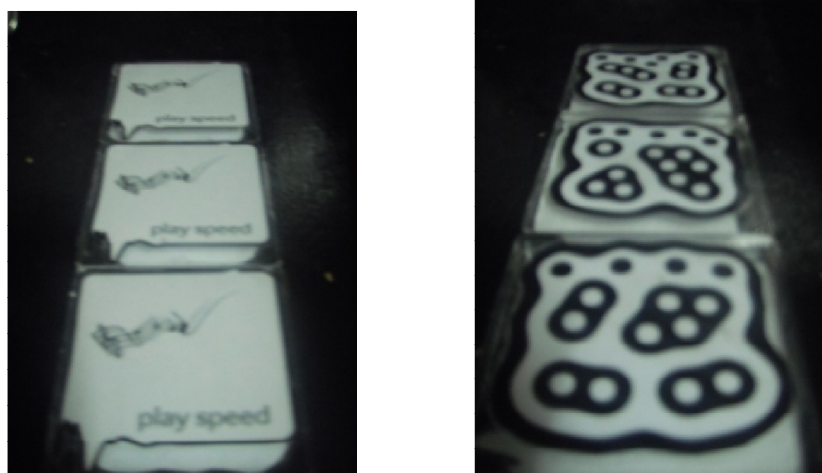


Fig. 60 Segundo Grupo de Knob's

A diferencia del sampler que sólo era capaz de controlar dos funciones, con este grupo de **Knob's** se pueden controlar tres parámetros, play/stop, control de volumen y la velocidad de reproducción del archivo de audio. Los dos primeros parámetros se controlan de la misma forma que en el sampler, mientras que la velocidad de reproducción varía de acuerdo al cambio de posición del **Knob** en el eje vertical, al moverlo hacia arriba la velocidad aumenta y en el sentido contrario la velocidad disminuye [Fig. 61].

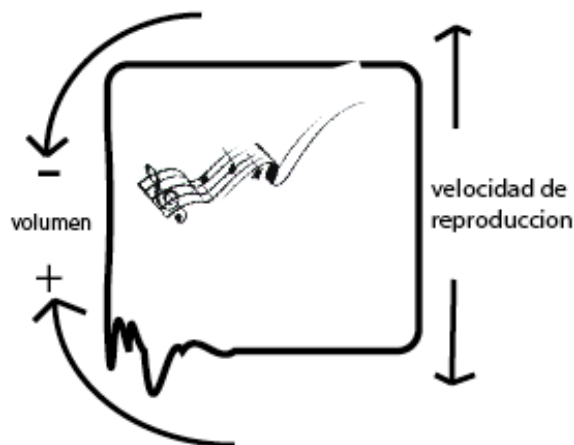


Fig. 61 Manejo de los Knob's

3.4.4 Control de Efectos (ID 7-8)

Por último los dos **Knob's** restantes fueron asignados a el control de los efectos (delay y scratch), estos únicamente pueden ser utilizados o agregados a nuestro primer grupo de **Knob's** (sampler), debido a que su única función es la de reproducir un archivo de audio a diferencia del SAMPLER_CHANGE donde dicho archivo ya es modificado con el mismo **Knob**. Cada uno de los **Knob's** tiene la capacidad de controlar dos parámetros de acuerdo al efecto asignado

El primero de ellos de acuerdo a su ID es el delay [Fig. 62], como se mencionó anteriormente la función de este efecto es la de sumar una señal desfasada en el tiempo con la señal original. Por lo que el control del tiempo de dicho desfasamiento es de suma importancia, este parámetro es modificado con la posición del **Knob** en el eje horizontal y este varía en un intervalo de 0s – 3s.

El otro parámetro es el control de volumen y este varía de acuerdo al cambio en el ángulo del **Knob** en la superficie de **Audio Surface**, con esto se tiene un control de qué porcentaje del efecto de desea escuchar dentro de la mezcla final.

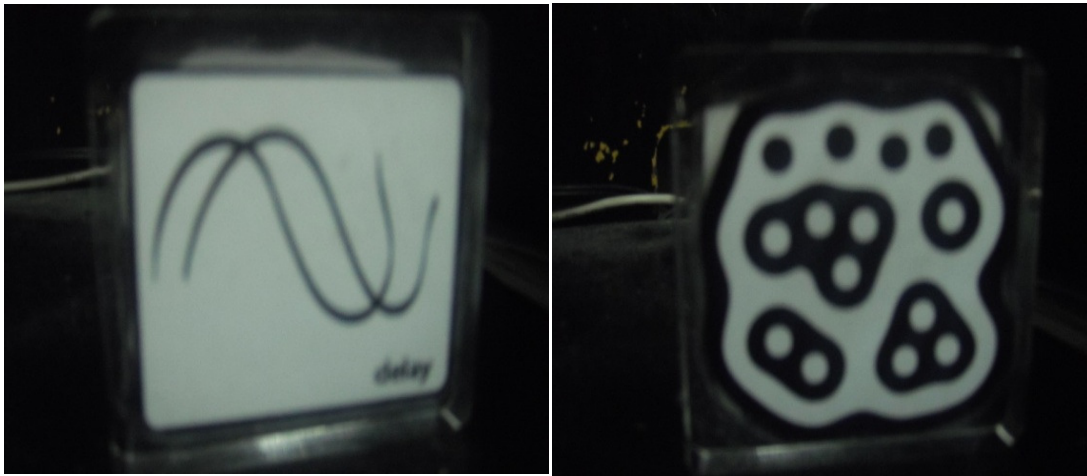


Fig. 62 Delay Knob

El último **Knob** fue asignado al efecto scratch [Fig. 63], el cual va de la mano con el uso de las tornamesas, ya que su sonido proviene del movimiento del disco hacia atrás y adelante provocando la sensación de estar rayando el disco, debido a esto el control de este parámetro se asignó a la variación del ángulo del **Knob** en la superficie de **Audio Surface**, con esto se buscó darle al usuario un control similar al de una tornamesa.



Fig. 63 Scratch Knob

El control de el delay [Fig. 64] es muy similar al de los demás **Knob** ya que aquí también se controla el volumen del efecto, que en otras palabras es la cantidad de presencia que el usuario quiere del efecto en la mezcla, este se controla al girar el **Knob** a la derecha e izquierda, el otro parámetro que nos permite modificar es el tiempo del delay el cual varía de acuerdo a su movimiento horizontal.

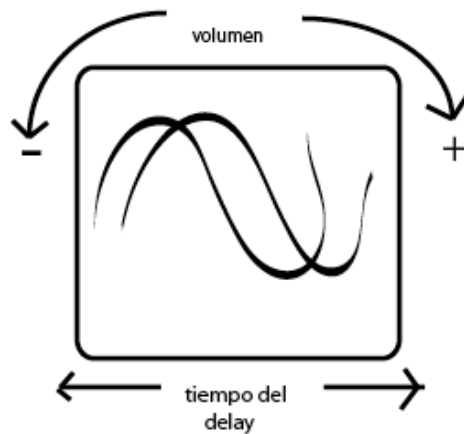


Fig. 64 Control del Delay

El scratch [Fig. 65] presenta una variación en cuanto al control del volumen ya que a diferencia de los otros **Knob's** aquí este depende del movimiento horizontal del **Knob** es decir aumentará al moverlo a la derecha y disminuirá a la izquierda. Mientras que en el control propio del efecto se realizará la girar este objeto, esto con el propósito de darle un control similar al de una tornamesa.

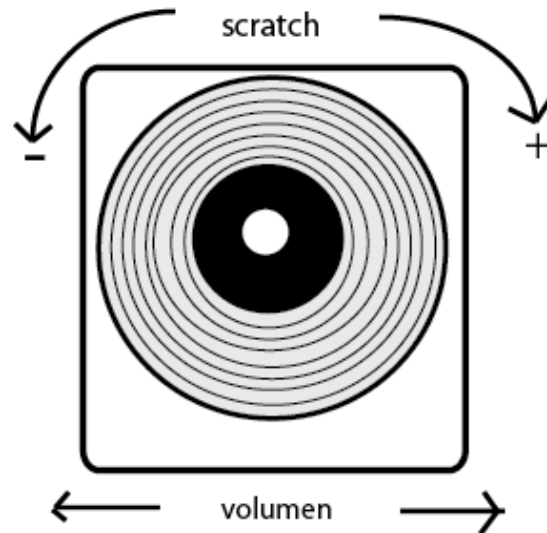


Fig. 65 Control del Scratch

3.5 Diseño de la Mesa [Fig. 66]

3.5.1 Mesa y Superficie

La mesa es una parte importante en la construcción de este proyecto, esta consta de una cámara de video y un proyector colocados en la parte inferior de la mesa, para que estos puedan cubrir toda la superficie de esta. Para la superficie de la mesa es recomendable usar plexiglás cubierto con una mica plástica transparente para su protección. Ya que estos materiales son transparentes para la superficie.

3.5.2 Iluminación

Para su rastreo dicha superficie tiene que estar bien iluminada para que la cámara y la visión por computadora los pueda reconocer correctamente. Así mismo para la proyección en la mesa esta tiene que estar lo suficientemente oscura para que las imágenes se puedan ver adecuadamente.

Debido a que la proyección debe de ser visible para el usuario, la cámara debe de trabajar con espectro invisible. Es por esto que se decide usar luz infrarroja, ya que no afecta la proyección en la superficie y muchas cámaras con CCD son sensibles a este tipo de luz.

3.5.3 Cámara

Es preferible usar cámaras de video que cuenten con sensores CCD ya que tendrán un mejor desempeño que aquellas que cuentan con chip CMOS. Y también que cuente con un buen lente, lo que excluye cámaras baratas.

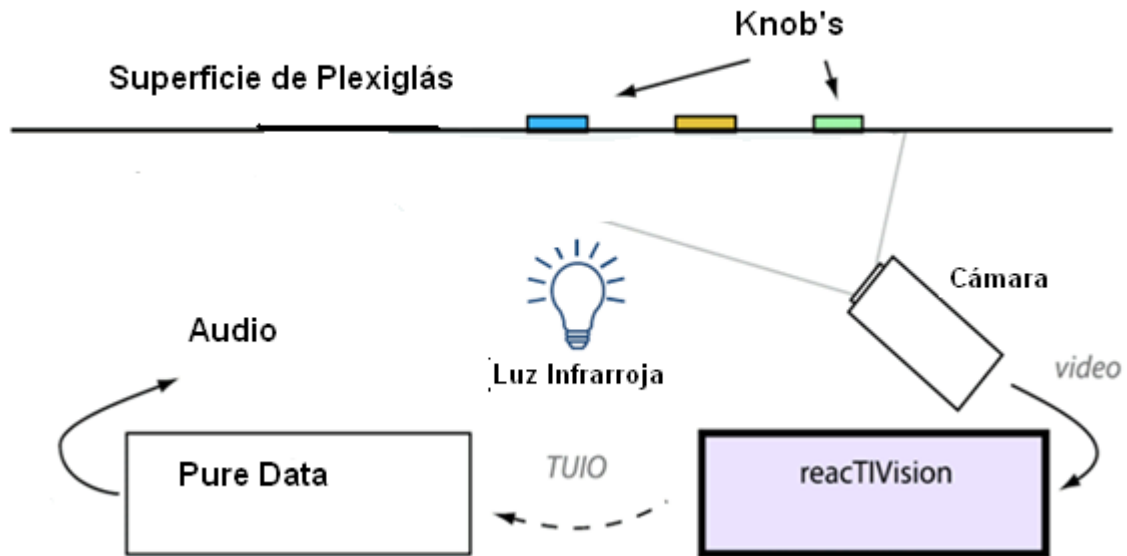


Fig. 66 Diagrama de la Mesa

Capítulo IV

Análisis de Costos

En este capítulo se explicarán los costos para la realización de **Audio Surface**, el análisis de costos es una parte fundamental para el desarrollo de cualquier proyecto, ya que es aquí donde se observa la viabilidad de este.

La parte más importante para poder comenzar a desarrollar **Audio Surface**, es lo que hoy en día se ha convertido un equipo de primera necesidad, la computadora, ya sea en su versión de escritorio o portátil. Para el desarrollo de este prototipo se necesitó una computadora lo puede hacer ver a este proyecto como costoso, pero para la versión comerciable de **Audio Surface** se tiene pensado sustituir la computadora por un microprocesador, lo cual ayudará a reducir costos considerablemente.

Como se menciona en capítulos anteriores, **Audio Surface** requiere de una cámara de video para poder rastrear los objetos que controlaran la síntesis musical. Por lo tanto la elección de la cámara de video es de gran importancia, para obtener una mejor calidad de imagen y mayor sensibilidad a la luz infrarroja, se recomienda el uso de una cámara de video con sensor CCD pero debido a que el costo de este equipo incrementa y el hecho de que este proyecto sea un prototipo; se utilizó una cámara con sensor de CMOS con costo de \$280.00 M.N.

Los **Knob's** son la base importante para el manejo de este instrumento electro-acústico, debido a que con ellos el usuario podrá manipularlo; es por esto que su tamaño requería ser el suficientemente grande para que el usuario los manipulara con mayor facilidad. Obedeciendo a esto se eligieron cubos de acrílico [Fig. 67] de 12 milímetros de espesor y 6 centímetros de lado, con costo de \$20.00 por cubo; tomando en cuenta que para este prototipo se usaran diez cubos el costo total de estos es de \$200.00

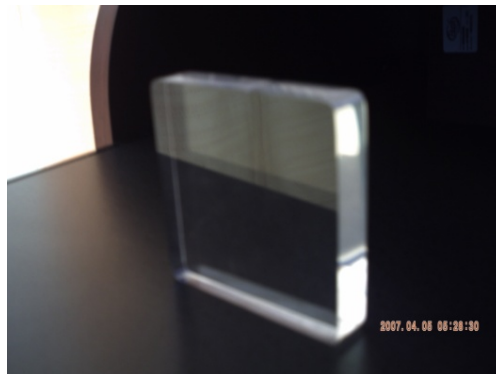


Fig. 67 Acrílico

El costo de las calcomanías [Fig. 68] es de \$300.00 esto se debe a que el proceso para su realización es más complejo debido al cuidado que se debe tener para la reproducción del código fiducial, ya que si este presenta alguna diferencia con el original no se será reconocido por reacTIVision.



Fig. 68 Calcomanía

Por lo tanto el costo de realización de este prototipo es de \$780.00 M.N., pero cabe aclarar que este costo no será el mismo de la versión comercial de **Audio Surface** debido a que en ella se incluirán una mayor cantidad de cubos de acrílico y por lo tanto de las calcomanías de los códigos fiduciales. Así mismo se añadirá el costo de una cámara de video con sensor CCD así como del microprocesador que se utilizara para sustituir la computadora y el costo del diseño y construcción de la mesa.

Otro punto importante dentro de los gastos de realización es la mano de obra del ingeniero; sólo se mencionó la parte de materias primas que se necesitan para elaborarlo, pero eso no es suficiente; también tomamos en cuenta el tiempo en que se tarda en realizar un programa, que no es una tarea sencilla, porque se necesita la destreza y la creatividad para poder hacer ritmos en la programación; tanto como hacer que varíe la frecuencia, el volumen, la amplitud, etc., dependiendo lo que el usuario necesite.

Este punto tiene un costo aproximado a los \$5,000.00 pesos; aquí es donde se ve reflejada la satisfacción de realizar un proyecto, que tenga como fin el placer de servir a la comunidad y a los amantes de la música electrónica.

Conclusión

Se concluye con este prototipo creado, que se cumplieron satisfactoriamente nuestras expectativas; las cuales fueron la creación de música electrónica en base a los objetos ya preestablecidos y con la parte esencial que éste tenía desde el principio, poder interactuar más con los sentidos dentro de la creación de sonidos, en base a estos poder innovar y desarrollar nuevos ritmos que lleven a los Dj's a estar más en contacto con la música.

Nuestro prototipo fue algo complicado pero al final resultó ser factible realizar, nos encontramos con problemas de programación que gracias a la ayuda de archivos relacionados nos pudimos dar guía de estos para salir avantes de la situación; también en relación con toda la información obtenida se nos facilitaron las cosas y nos abrieron campo al conocimiento de cosas que no teníamos en cuenta, porque sabíamos cosas esenciales pero conscientes de que no todas; sin embargo creemos que hicimos un buen trabajo y que va a ayudar a muchas personas y más a los amantes de la música de este género, porque no sólo es ponerse en frente y estar creando sonidos, sino también es darle sentido y forma a la creación de estos.

Para finalizar daremos gracias a todos los que nos apoyaron en este proyecto, con su ayuda fue posible realizarlo y creemos que esto va a revolucionar la música en todos los ámbitos posibles y más que eso; esto nos va a hacer que día a día trabajemos duro para poder superarnos y ser mejores.

Anexos

Anexo A

Glosario de Anglicismos

- Acid House: evolución del house hacia los sonidos crujientes, repetitivos y lisérgicos creados con el sintetizador TB-303 de Roland. Su símbolo fue el famoso "Smiley", y sus inventores el grupo Phuture (DJ Pierre y Spanky) con el tema "Acid Trax" (editado en 1987). En 1988 el acid house dio origen en Gran Bretaña al denominado "Verano del Amor".
- Ambient: es un género musical en el cual el sonido es más importante que las notas. Generalmente se identifica por ser ampliamente atmosférica y ambiental por naturaleza.
- CCD *charge-coupled device*: (dispositivo de cargas [eléctricas] interconectadas) es un circuito integrado que contiene un número determinado de condensadores enlazados o acoplados. Bajo el control de un circuito interno, cada condensador puede transferir su carga eléctrica a uno o a varios de los condensadores que estén a su lado en el circuito impreso.
- CMOS (*complementary metal oxide semiconductor*) utilizados en algunas cámaras digitales y en numerosas Webcam.
- Dance: es la música compuesta específicamente para acompañar al baile, y se caracteriza por tener sonidos sintéticos, ritmos bien marcados y bastante repetitiva para ser fácilmente bailables, letras pegadizas e instrumentación marcada en compases de 4/4.
- Dreamweaver: es una aplicación en forma de estudio enfocada a la construcción y edición de sitios y aplicaciones Web basadas en estándares. La gran ventaja de este editor sobre otros es su gran poder de ampliación y personalización del mismo, puesto que en este programa, sus rutinas (como la de insertar un hipervínculo, una imagen o añadir un comportamiento) están hechas en Javascript-C, lo que le ofrece una gran flexibilidad en estas materias. Esto hace que los archivos del programa no sean instrucciones de C++ sino, rutinas de Javascript que hace que sea un programa muy fluido, que todo ello hace, que programadores y editores web hagan extensiones para su programa y lo ponga a su gusto.
- Dub: género musical electrónico desarrollado en Jamaica a partir de los años 70, incluye habitualmente efectos de sonido de eco y reverberación añadidos a una canción ya existente, acompañados muchas veces de fragmentos de las letras de las canciones originales. Generalmente se retira gran parte de los vocales y se hace más énfasis en bajo y batería. Muchas veces también se incluyen otros efectos sonoros como tiros, sonidos de animales, sirenas de policía, alarmas, etc.
- Fiducial: Término dado por Ross Bencina para nombrar a los símbolos usados en conjunto con reactIVision.
- Ghetto: es un área separada para la vivienda de un determinado origen étnico, cultural o religioso, voluntaria o involuntariamente, en mayor o menor reclusión.

El término se empleó, originalmente, para indicar los barrios judíos; el uso se ha extendido hoy a cualquier área en la que la concentración de un determinado grupo social es excluyente.

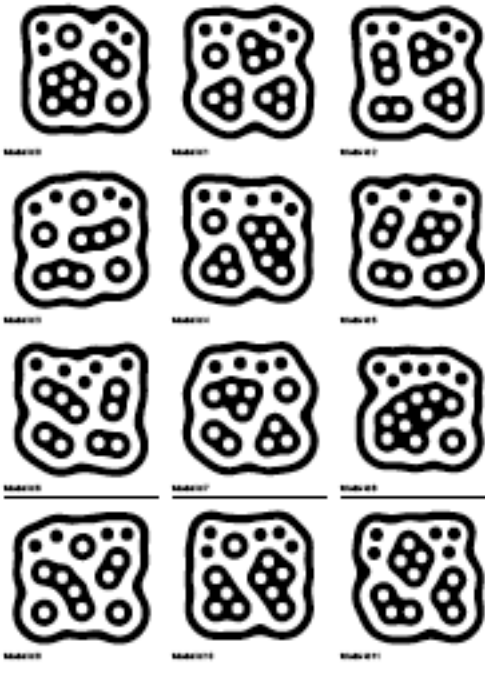
- **Hardware:** corresponde a todas las partes físicas y tangibles de una computadora: sus componentes eléctricos, electrónicos, electromecánicos y mecánicos; sus cables, gabinetes o cajas, periféricos de todo tipo y cualquier otro elemento físico involucrado.
- **Hip-Hop:** movimiento artístico que surgió en Estados Unidos a finales de los años 1960 en las comunidades afroamericanas y latinoamericanas de barrios populares neoyorquinos como Bronx, Queens y Brooklyn.
- **House:** estilo de música electrónica, que se originó en Chicago, Estados Unidos, a fines de los años 1970 y principio de los años 1980. tiene fuertes influencias del soul y el funk y la música disco. Generalmente, imita a la percusión de la música disco, con graves sintetizados muy prominentes, baterías electrónicas, efectos electrónicos, samples.
- **Loop:** anglicismo que en música electrónica consiste en uno o varios samples sincronizados que ocupan generalmente uno o varios compases musicales exactos y son grabados o reproducidos enlazados en secuencia una vez tras otra dando sensación de continuidad. El término se puede traducir como "bucle".
- **MIDI:** son las siglas del (Interfaz Digital de Instrumentos Musicales). Se trata de un protocolo industrial estándar que permite a las computadoras, sintetizadores, secuenciadores, controladores y otros dispositivos musicales electrónicos comunicarse y compartir información para la generación de sonidos.
- **Open Source:** (Código abierto) es el término con el que se conoce al software distribuido y desarrollado libremente.
- **Rap:** es un tipo de recitación rítmica de rimas, juegos de palabras y poesía surgido a mediados del siglo XX entre la comunidad negra de los Estados Unidos. Es uno de los cuatro pilares fundamentales de la cultura hip hop.
- **Remix:** en la música es una mezcla alternativa de una canción en un estudio de sonido, para darle un nuevo aspecto sonoro o mejorar la calidad de sonido, a veces con la incorporación de nuevos ritmos y efectos. Pero especialmente los remix son usualmente intencionados para transformar en bailables canciones que no lo son. La acción de crear un remix o remezcla es generalmente conocida como remixear o remezclar.
- **Sample:** es una voz inglesa que se traduce como muestra y se utiliza para denominar al sonido grabado en cualquier tipo de soporte. Comúnmente, en la música pop y otros estilos musicales actuales, el término se utiliza para designar a los fragmentos de sonido que se extraen de grabaciones anteriores y se encajan en la grabación de una nueva canción.
- **Sampler:** aparato que permite muestrear digitalmente secuencias sonoras o samples para ser reproducidas posteriormente, o transformadas mediante

efectos. También permite recuperar y almacenar éstas en un soporte de almacenamiento secundario, como discos duros, unidades ZIP, disquetes, etc.

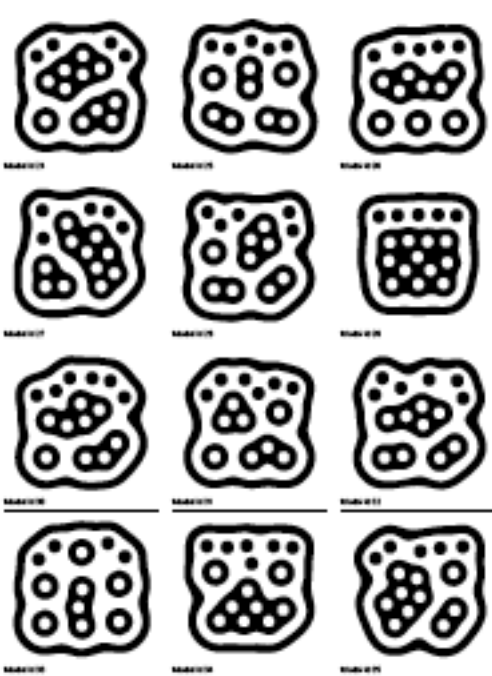
- **Scratch:** anglicismo que significa literalmente rayar o arañar, En música, se llama así una técnica utilizada por los DJs de hip-hop y música electrónica, que consiste en mover un disco de vinilo hacia adelante y hacia atrás sobre el plato del tocadiscos para crear un efecto parecido al de rayar el disco y que, bien utilizado, ayuda a construir ritmos y frases melódicas.
- **Slap:** técnica para tocar el contrabajo, aunque también se usa en el bajo eléctrico, produce un sonido muy percusivo ya que las cuerdas del instrumento se golpean contra el mástil.
- **Software:** palabra proveniente del inglés (literalmente: partes blandas o suaves), que en español no posee una traducción adecuada al contexto, por lo cual se la utiliza asiduamente sin traducir y así fue admitida por la Real Academia Española. Aunque no es estrictamente lo mismo, suele sustituirse por expresiones tales como programas (de computadora) o aplicaciones (de computadora).
- **Synthpop:** también conocido como **tecno pop**, es un género musical derivado de la new wave y desarrollado a finales de los años 70 y principios de los años 80, principalmente en el Reino Unido y Alemania, aunque también en Estados Unidos. El synth-pop tiene sus propias tendencias estilísticas que lo diferencian de otra música producida con los mismos medios. Estos incluyen la explotación de la artificialidad (los sintetizadores no son usados para imitar instrumentos acústicos), el empleo de ritmos mecánicos, arreglos vocales como contrapunto a la artificialidad de los instrumentos
- **Techno:** género musical de música electrónica que se caracteriza por el compás de 4/4, cuyos tiempos (negras) son marcados por un bombo (habitualmente llamado kick), y la consecuente utilización de instrumentos electrónicos, como sintetizadores y samplers. A diferencia de otros géneros como el electro pop, el techno no utiliza la estructura de una canción, sino que es estructurado de manera repetitiva, para maximizar el efecto bailable de la música.
- **Turntablism:** se podría traducir como «el arte de hacer girar "las mesas" (de discos)».
- **Turntablist:** es el arte de arreglar o crear música mediante efectos de sonido y manipulación de las rutinas de rotación y lectura de los discos de vinilo sobre un plato giradiscos.
- **Web-Cam:** es una pequeña cámara digital conectada a una computadora, la cual puede capturar imágenes y transmitir las a través de Internet, ya sea a una página web o a otra u otras computadoras de forma privada.

Anexo B

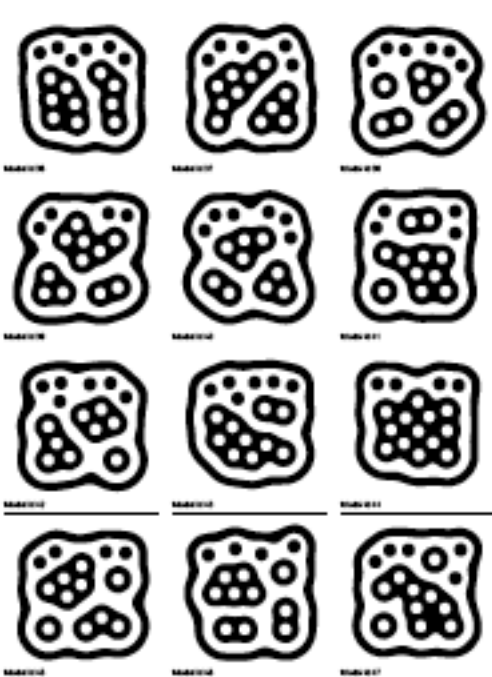
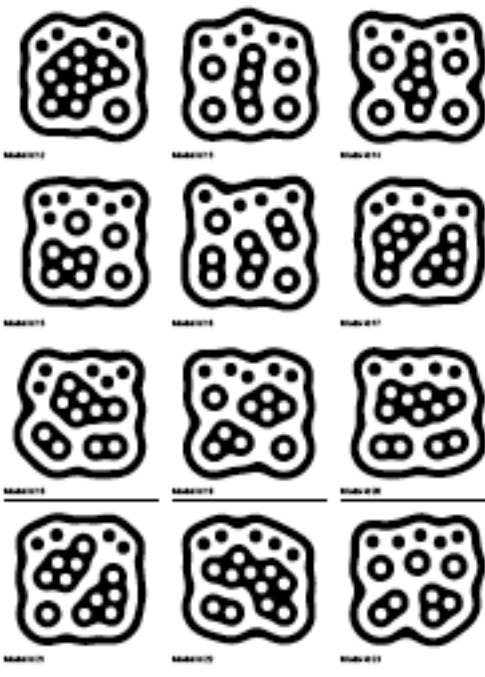
Símbolos Fiduciales

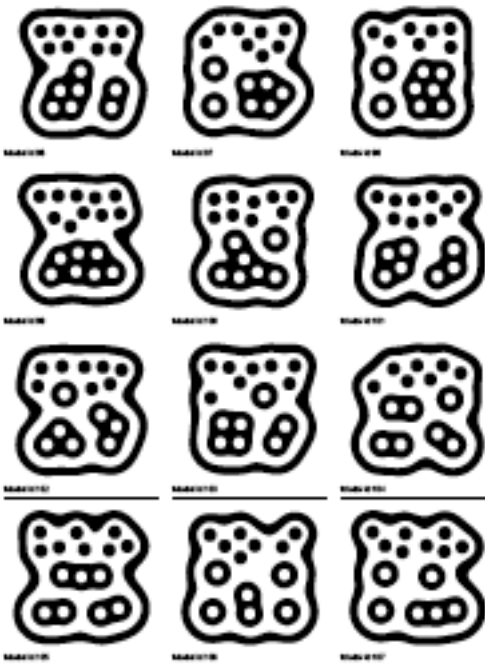


Copyright © 2008 Pearson Education, Inc. All rights reserved. This material is intended solely for the personal use of the individual user and is not to be disseminated broadly.

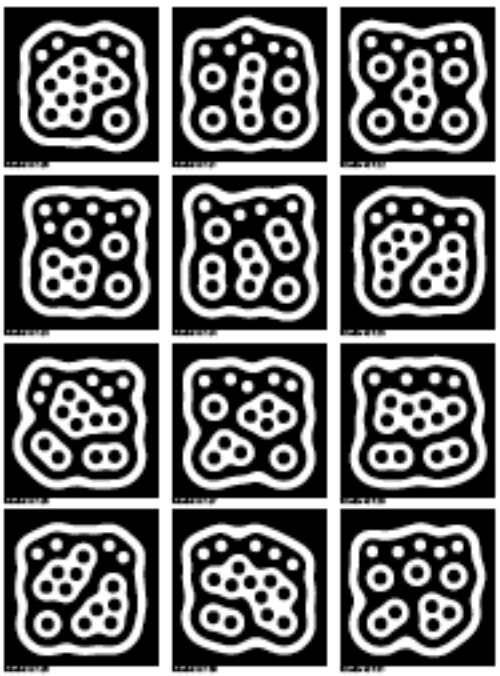


Copyright © 2008 Pearson Education, Inc. All rights reserved. This material is intended solely for the personal use of the individual user and is not to be disseminated broadly.

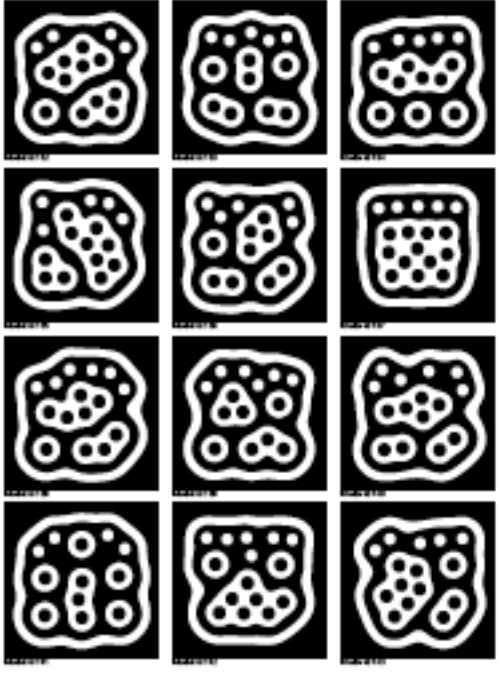
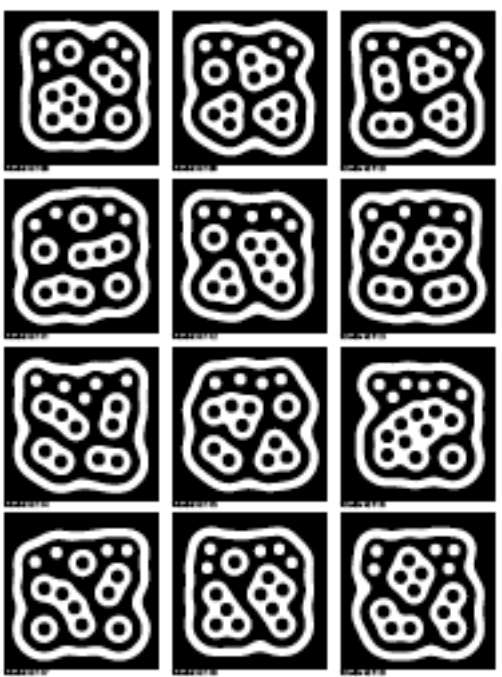


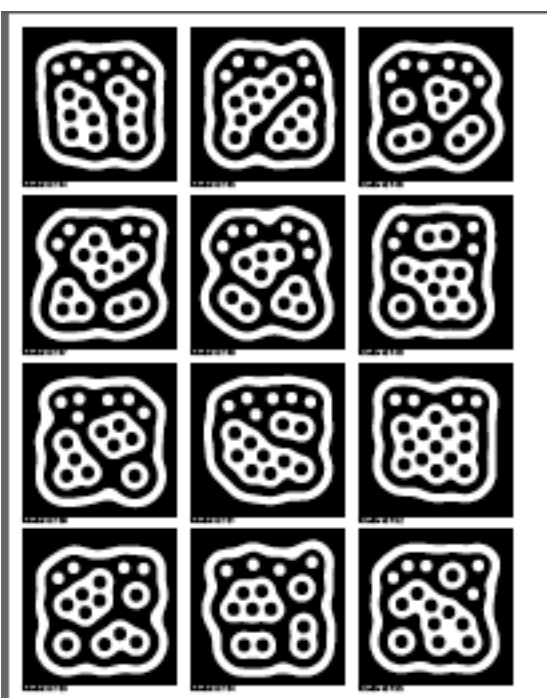


Copyright © 1999 Pearson Education, Inc. This book is not to be used in conjunction with the software (Pearson Education, Inc.) of the same name.

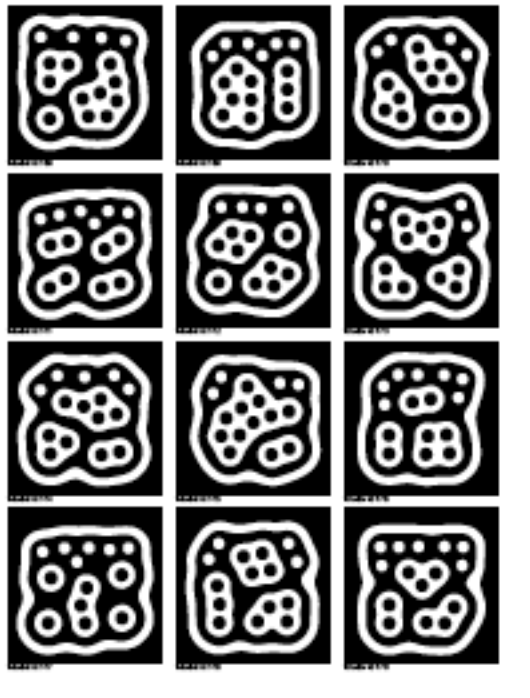


Copyright © 1999 Pearson Education, Inc. This book is not to be used in conjunction with the software (Pearson Education, Inc.) of the same name.

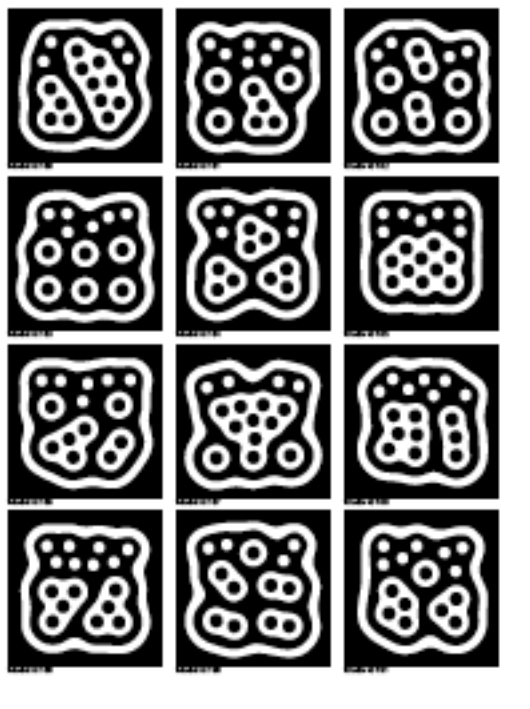
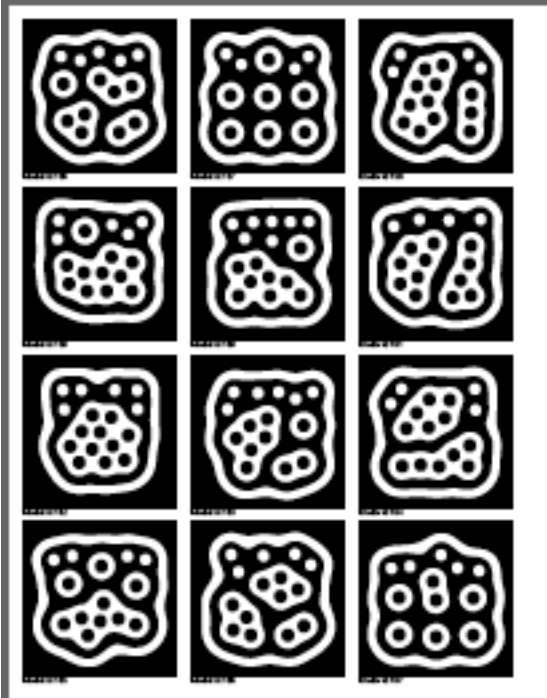


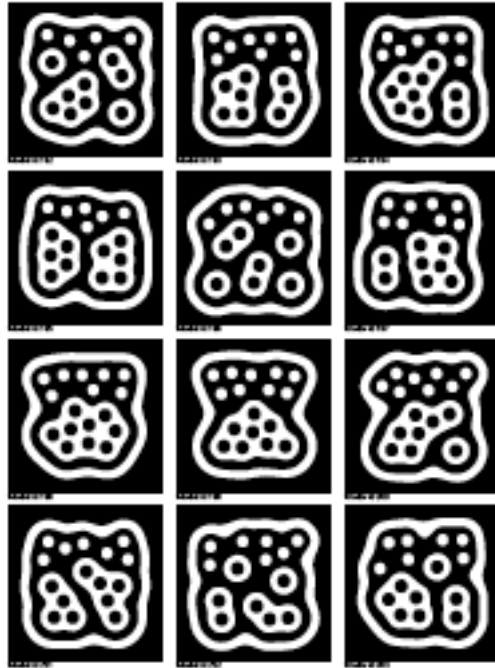


Copyright © 1988 Pearson Education, Inc. All rights reserved. This material is intended solely for the personal and internal use of the individual user and is not to be disseminated, distributed, or used in any other manner.

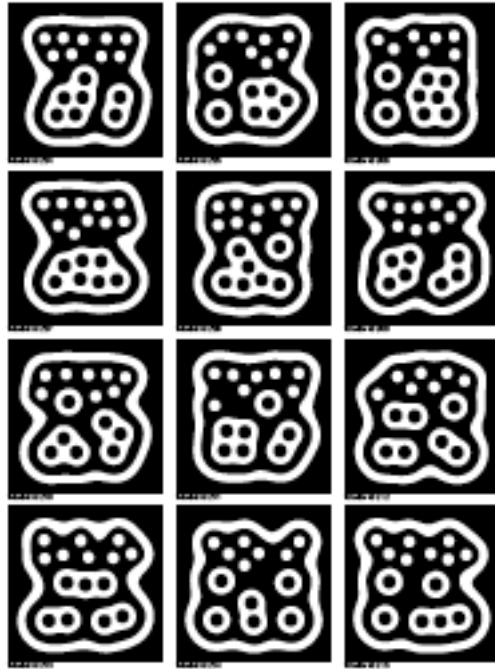


Copyright © 1988 Pearson Education, Inc. All rights reserved. This material is intended solely for the personal and internal use of the individual user and is not to be disseminated, distributed, or used in any other manner.





Copyright © 2008 Pearson Education, Inc. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or by any information storage or retrieval system, without permission in writing from Pearson Education, Inc.



Anexo C

Datos Biográficos Breves

- **Luigi Russolo:** (30/04/1885 – 04/02/1947) fue un pintor futurista y compositor italiano. Entre sus obras más interesantes destaca “Los Relámpagos” (1909-10). En 1910 firmó el Manifiesto Futurista y tuvo una participación activa en este grupo, su música y sus instrumentos para hacer ruidos contribuyeron significativamente al movimiento futurista. Además, expuso sus principios en el libro El arte de los ruidos de 1916.
- **León Theremin:** (15/11/1896 – 03/11/1993) fue un inventor ruso. Es muy famoso por la invención del theremín, uno de los primeros instrumentos musicales electrónicos. Desde joven estudió con gran pasión el mundo de la música y de la electricidad.
- **Maurice Martenot:** (14/10/1898 – 8/10/1980), fue un sargento, violonchelista y radiotelegrafista francés, conocido por descubrir en 1928 las ondas que llevan su nombre a partir de la pureza de las vibraciones de los tubos radiales, lo que le llevó a una serie de investigaciones en el campo del sonido y la electrónica.
- **Robert A. Moog:** (23/05/1934 - 21/08/2005) fue un inventor estadounidense. Es famoso por crear, junto con otras personas el sintetizador, instrumento musical con el que se pueden crear una gama prácticamente infinita de sonidos.
- **Pierre Henri Marie Schaeffer:** (14/08/1910 – 19/08/1995) compositor francés considerado el creador de la música concreta. Es autor del libro titulado Tratado de los objetos musicales, en donde expone toda su teoría sobre este tipo de música.
- **Herbert Eimert:** (1897 - 1972) compositor y teórico musical alemán, defensor de la música dodecafónica, fue el iniciador de la música electrónica y colaboró con K. Stockhausen.
- **Herbert A. Deutsch:** (1933) compositor, inventor y maestro de Estados Unidos. Actualmente profesor de música electrónica y composición en la Universidad de Hofstra. Su más grande logro fue co-inventar el “Moog Modular” con Robert Moog.
- **John Cage:** (5/09/1912 – 12/08/1992) fue un compositor e instrumentista estadounidense, revolucionó la música contemporánea dotándola de un lenguaje caótico. Una de sus influencias fue el Zen, el cual le condujo en lo musical a usar silencios interminables, sonidos desconectados, casuales y atonales con un volumen, duración y timbre aleatorios.
- **Vladimir Kirilovitch Ussachevsky:** (03/09/1911 – 02/01/1990) compositor musical, conocido por su trabajo acerca de la música electrónica.

- **Wendy Carlos** (14/11/1939) compositora de música electrónica estadounidense., fue una de las primeras intérpretes famosas de música electrónica que usaron sintetizadores.
- **Pink Floyd:** grupo de rock británico que cosechó gran popularidad gracias a su música psicodélica que, con el paso del tiempo, fue evolucionando hacia el rock progresivo y el rock sinfónico.
- **Tangerine Dream:** banda alemana de música electrónica perteneciente a la escena Kraut Rock, fue pionera del ambient, del space rock y del uso de secuenciadores.
- **King Tubby:** (28/01/1941 – 06/02/1989) en Jamaica fue un ingeniero de sonido, conocido por su influencia en el desenvolvimiento de la música dub en las décadas de 1960s y 1970s. El es siempre citado como el inventor del remix, y así puede ser visto como un antecedente directo de la música electrónica, también es considerado el pionero del Dub.
- **Kraftwerk (central energética):** grupo musical vanguardista alemán decisivo para el desarrollo de la música electrónica. Las técnicas musicales y los instrumentos introducidos por ellos son hoy básicos, sin embargo, fueron realmente innovadores en su tiempo. Por las experimentaciones realizadas y por constituir la base sobre la que luego se asentó el género, son considerados como los padres de la música electrónica, ya que fueron los primeros en tocar música basada íntegramente en la utilización de instrumentos electrónicos.
- **Jean-Michel André Jarre:** (24/08/1948) compositor e intérprete francés de música electrónica. Experimentó en diversos frentes: "The Dustbins", "La Cage", "Erosmachine", "Hypnose", "Happiness is a Sad Song".
- **Evangelos Odysseas Papathanassiou:** (29/03/194) mejor conocido como Vangelis, es un famoso teclista y compositor de música electrónica. Su música se caracteriza por el uso de sintetizadores y ocasionales instrumentos acústicos (sobre todo de percusión) para crear atmósferas de sonido envolvente, en un tono generalmente grandioso y solemne.
- **DJ Kool Herc (Clive Campbell):** (16/04/1955) en Kingston, Jamaica, es un músico y productor jamaicano-americano, conocido como "The father of Hip Hop" en la década de los 70.
- **Frankie Knuckles:** (18/01/1955) es un Dj y productor estadounidense, es aclamado por muchos admiradores como el "padre" de la música House, que surgió a fines de los años 70. Como DJ su trabajo desde Chicago ayudó a popularizar el estilo no solo tocando sino también por el aporte de sus producciones y remixes.

- **Kevin Saunderson:** (05/09/1964) músico estadounidense, empezó su carrera como Dj y desarrolló nuevas habilidades que le introdujeron como productor musical que creó su propio sello discográfico KMS Records. Es conocido como el “padre” del Techno en Detroit.
- **Derrick May:** (06/04/1963) a temprana edad empezó a explorar la música electrónica, acompañado de sus compañeros, Juan Atkins y Kevin Saunderson. May es considerado como el creador de una variante futurística de la música House conocida como techno dub.
- **Joseph Saddler (Grandmaster Flash):** (01/01/1958) fue pionero de toda una serie de técnicas como el “cutting” (cambiar de un disco que está sonando en un plato al otro sin perder el ritmo, esto mediante la técnica del "Reloj"), “punch-in” (interpolación breves fragmentos de otro disco interrumpiendo rítmicamente la reproducción de otro), “Duets” (retrasar manualmente los platos para repetir breves muestras de sonido), y “phasing” (superponer dos copias del mismo disco, lo que produce un sonido fluctuante). También fue de los primeros en cambiar los arreglos de los temas utilizando copias duplicadas de discos, editando y repitiendo las partes de clímax (break beat) girando los discos hacia atrás y adelante.
- **Grand Wizzard Theodore** (1962) es ampliamente acreditado como el inventor del “scratch”. Además del “scratch”, logró fama por su dominio de la aguja gotas y otras técnicas que él inventó o perfeccionado.
- **Public Enemy:** grupo de música Rap de Nueva York, Estados Unidos.
- **Patrick Dedo (DJ Noise):** se le considera como uno de los más talentosos mezcladores en vivo, creció con ABBA.
- **Rick Wakeman:** (18/05/1949) empezó siendo un pianista entrenado en música clásica y fue un pionero en el uso de teclados electrónicos. Es considerado por la mayoría de los críticos musicales como el mejor tecladista de rock de la historia.
- **Jean-Claude Risset:** (18/03/1938) compositor francés, conocido por sus contribuciones pioneras a equipo de música. Hizo grabaciones digitales de trompetas y luego estudió su composición timbral utilizando "tono sincrónica" espectro de herramientas de análisis, y reveló que los armónicos de estos instrumentos son muy diferentes dependiendo de la altura, duración y sonoridad. También es mérito de la ejecución de las primeras experiencias en una variedad de técnicas incluyendo la síntesis de FM y tabla de ondas.

- **Charles Dodge:** (5/06/1942) compositor más conocido por su música electrónica, concretamente su equipo de música. Él es un ex-alumno de Darius Milhaud y Gunther Schuller.
- **John M. Chowning:** (1934) compositor, músico, inventor, profesor y más conocido por su trabajo en la Universidad de Stanford y su invención de la síntesis FM.
- **Paul Lansky:** (18/06/1944) Americano creador de música electrónica por computadora que ha estado produciendo obras de la década de 1970 hasta el día de hoy.
- **Luc Ferrari:** (1929-2005) compositor francés, que destacó principalmente por ser uno de los pioneros de la música electro-acústica, y por su personal interpretación de algunas de las corrientes musicales contemporáneas.
- **Denis Smalley Arthur:** (1946) compositor de música electroacústica, con especial interés en la música acusmática. Él describe su enfoque como "Spectro-morfológicas", con el desarrollo de los sonidos en el tiempo.
- **Horacio Vaggione:** (1943) compositor argentino de música concreta y electroacústica que se especializa en micromontaje, síntesis granular y microsonido.
- **Robert Normandeau:** (11/03/1955) compositor de música electroacústica, estudió en la Universidad Laval en Quebec, y en la Universidad de Montreal, donde estudió con Marcelle Deschênes y Francis Dhomont.
- **Luis de Pablo Costales** (28/01/1930) compositor español, realizó sus primeros estudios musicales en Fuenterrabia y posteriormente en Madrid. Gran parte de su formación musical es autodidacta aunque recibió consejos de Maurice Ohana, Max Deutsch y otras personalidades. Fundador de importantes grupos de música contemporánea. Fue el creador también del primer laboratorio de música electroacústica en España.
- **Juan García Esquivel:** (20/01/1918 — 03/01/2002) arreglista, pianista y compositor internacional. Es conocido por crear un estilo de música único en su género, ocasionalmente llamado Lounge o Space Age Pop.

Bibliografía

- The theory and technique of electronic music; Miller Puckette; 2006
- <http://es.geocities.com/erthisa/historiadjing.htm>
- <http://productions.caffix.org.mx/sintetizadores>
- http://es.encarta.msn.com/encyclopedia_761552222/M%C3%BAsica_electr%C3%B3nica.html
- <http://www.aulaactual.com/equipamiento/isequenciadores.htm>
- <http://www.tuio.org/>
- <http://frecuenciamfundamental.blogspot.com/2009/06/glosario-sintesis-aditiva.html>
- <http://www.musica-y-tecnologia.com/Módulos/Artículos/ImprimirArticulo.asp?ID=59>
- <http://tecnicaaudiovisual.kinoki.org/sonido/fisica.htm>
- <http://puredata.info>
- <http://www.taringa.net/posts/info/1762903/Breve-historia-de-la-musica-electronica.html>
- <http://www.reactable.com/>
- <http://reactivision.sourceforge.net/>
- <http://www.dubtable.net/>
- <http://www.jamespatten.com/audiopad/>
- <http://www.d-touch.org/audio/>
- <http://mtg.upf.es/reactable/>