

UNIVERSIDAD NACIONAL DE LA MATANZA

Departamento de Ingeniería e Investigaciones Tecnológicas

**Ingeniería en Electrónica**

**Técnicas digitales III**

Pedal Multiefecto para Guitarra Eléctrica

**Integrante:** Rosmirez, Jorge

**Profesores:** Lic. Carlos Maidana

Ing. Guillermo Buranits

Ing. Mauro Cipollone

2020

**Tabla de contenidos**

[1 Descripción general 2](#_Toc53326060)

[2 Introducción teórica 2](#_Toc53326061)

[2.1 Que es el Sonido 2](#_Toc53326062)

[2.2 conceptos básicos del sonido 3](#_Toc53326063)

[2.3 Los parámetros del sonido 4](#_Toc53326064)

[2.3.1 Tono o frecuencia 4](#_Toc53326065)

[2.3.2 Sonoridad 4](#_Toc53326066)

[2.3.3 Timbre 4](#_Toc53326067)

[2.3.4 Percepción de duración 4](#_Toc53326068)

[2.3.5 Envolvente o articulación 5](#_Toc53326069)

[2.3.6 Difusión 5](#_Toc53326070)

[3 Estado del Arte 5](#_Toc53326071)

[3.1 Clasificación de los efectos 6](#_Toc53326072)

[3.1.1 Efectos de rango dinámico 6](#_Toc53326073)

[3.1.2 Distorsiones 6](#_Toc53326074)

[3.1.3 Ecualizadores 6](#_Toc53326075)

[3.1.4 Efectos de modulación 7](#_Toc53326076)

[3.1.5 Efectos de transposición de tono 7](#_Toc53326077)

[3.1.6 Efectos basados en retardos 7](#_Toc53326078)

[3.1.7 Efectos simuladores 8](#_Toc53326079)

[3.1.8 Reductores de ruido 8](#_Toc53326080)

[4 Análisis y desarrollo de los efectos 8](#_Toc53326081)

[4.1 Delay 8](#_Toc53326082)

[4.1.1 Introducción 8](#_Toc53326083)

[4.1.2 Implementación 9](#_Toc53326084)

[4.1.3 Estructura básica de un Delay 10](#_Toc53326085)

[4.2 Fuzz 10](#_Toc53326086)

[4.2.1 Introducción 10](#_Toc53326087)

[4.2.2 Implementación 11](#_Toc53326088)

[4.2.3 Procesamiento de la señal 12](#_Toc53326089)

[4.3 Wah-Wah 13](#_Toc53326090)

[4.3.1 Introducción 13](#_Toc53326091)

[4.3.2 Implementación 14](#_Toc53326092)

[5 Descripción técnica del proyecto. 15](#_Toc53326093)

[5.1 Hardware 15](#_Toc53326094)

[5.1.1 Protocolo I2S 16](#_Toc53326095)

[5.1.2 Driver CS43L22 17](#_Toc53326096)

[5.1.3 ADC 18](#_Toc53326097)

[5.1.4 TIMER 18](#_Toc53326098)

[5.1.5 Configuración de Clock 18](#_Toc53326099)

[5.1.6 Configuraciones de pines 19](#_Toc53326100)

[5.1.6.1 Pulsadores para el usuario 19](#_Toc53326101)

[5.1.6.2 ADC 19](#_Toc53326102)

[5.1.6.3 Configuración con el Códec de DAC 19](#_Toc53326103)

[5.1.6.4 Transmisión de audio al códec del DAC 19](#_Toc53326104)

[5.1.7 Diagrama en bloques 19](#_Toc53326105)

[5.1.8 Esquemático 20](#_Toc53326106)

[5.1.8.1 Jack 20](#_Toc53326107)

[5.1.8.2 Divisor resistivo 20](#_Toc53326108)

[5.1.8.3 Potenciómetro 21](#_Toc53326109)

[5.1.8.4 Filtro Pasa Banda 21](#_Toc53326110)

[5.1.8.5 Circuito impreso 22](#_Toc53326111)

[5.1.8.6 Modelo 3D: 23](#_Toc53326112)

[6 Software 23](#_Toc53326113)

[6.1.1 Descripción del funcionamiento de cada rutina 23](#_Toc53326114)

[6.2 Software adicional requerido 31](#_Toc53326115)

[6.2.1 Matlab 32](#_Toc53326116)

[6.2.1.1 Implementación para efecto Fuzz 32](#_Toc53326117)

[6.2.1.2 Implementación para el efecto WahWah 32](#_Toc53326118)

[7 Modo de operación 32](#_Toc53326119)

[7.1 Luces led indicadoras de efecto 32](#_Toc53326120)

[8 Conclusiones 33](#_Toc53326121)

[8.1 Audio Weaver 33](#_Toc53326122)

[8.2 Relevamiento de experiencias con alinealidad característica del Fuzz 34](#_Toc53326123)

[8.3 Vicisitud con efecto WahWah 34](#_Toc53326124)

[8.4 Relevamiento de experiencias por necesidad de conversión de formatos 35](#_Toc53326125)

[9 Proyecto finalizado 35](#_Toc53326126)

[10 Referencias 36](#_Toc53326127)

**Tabla de ilustraciones**

[Figura 1 – Esquema basico de conexión 2](#_Toc53303795)

[Figura 2: El sonido como una onda de presión 3](#_Toc53303796)

[Figura 3 – Clasificación de efectos 8](#_Toc53303797)

[Figura 4 – Esquema basico delay 9](#_Toc53303798)

[Figura 5 – Ecuacion 1 Delay 10](#_Toc53303799)

[Figura 6 - Ecuacion 2 Delay 10](#_Toc53303800)

[Figura 7 – Gráfico del desfasaje de señal con Delay 10](#_Toc53303801)

[Figura 8: Diagrama de efecto Fuzz [4] 11](#_Toc53303802)

[Figura 9 – Una señal sin afectar y afectada por el efecto Fuzz 11](#_Toc53303803)

[Figura 10 – Salida de una señal con Overdrive 12](#_Toc53303804)

[Figura 11 – Formula tipica Fuzz 12](#_Toc53303805)

[Figura 12 – Alinealidad de la curva 13](#_Toc53303806)

[Figura 13 – Salida de una señal con Fuzz 13](#_Toc53303807)

[Figura 14: Conexión serie de en bloque y filtros Peak 14](#_Toc53303808)

[Figura 15 Wah Wah: Filtro pasa banda variable en el tiempo 14](#_Toc53303809)

[Figura 16: Filtro Wah-Wah 15](#_Toc53303810)

[Figura 17: efecto con filtro M-fold wah-wah 15](#_Toc53303811)

[Figura 18 – Grafico I2S 16](#_Toc53303812)

[Figura 19 – Clock & Configuration (CUBE) 18](#_Toc53303813)

[Figura 20 – Diagrama en bloques 19](#_Toc53303814)

[Figura 21 – Esquemático general 20](#_Toc53303815)

[Figura 22 – Jack de entrada 20](#_Toc53303816)

[Figura 23 – Divisor resistivo 21](#_Toc53303817)

[Figura 24 – Conexionado Potenciometro 21](#_Toc53303818)

[Figura 25 – Ecuacion de filtro pasa bajo 22](#_Toc53303819)

[Figura 26 – Routeo de pistas 23](#_Toc53303820)

[Figura 27 – modelo 3D Placa 23](#_Toc53303821)

[Figura 28 – Maquina de estado de efectos 24](#_Toc53303822)

[Figura 29 - Señal sin nivelar y señal nivelada 25](#_Toc53303823)

[Figura 30 - Señal sin shiftear y señal shiteada 25](#_Toc53303824)

[Figura 31 - Señal de salida shifteada 25](#_Toc53303825)

[Figura 32 – Realimentación con desplazamiento (eco) 28](file:///C:\Users\Juan%20Rosmirez\Documents\Jorge\Home\Pedal-Multiefecto\Informe%20final%20-%20Jorge%20Rosmirez_v04.docx#_Toc53303826)

[Figura 33 – Posición Actual del vector 28](#_Toc53303827)

[Figura 34 – Posición actual y atrasado 28](#_Toc53303828)

[Figura 35 – Posición actual y un retraso mas alejado 29](#_Toc53303829)

[Figura 36 – Diagrama implementacion Efecto Fuzz 29](#_Toc53303830)

[Figura 37 – Diagrama de flujo de WahWah 31](#_Toc53303831)

[Figura 38 – Interfaz AudioWeaver 33](#_Toc53303832)

[Figura 39 – Imagen superior del prototipo 34](#_Toc53303833)

[Figura 40 – imagen frontal del prototipo 34](#_Toc53303834)

**Tabla de ecuaciones**

[Ecuación 1 16](#_Toc52017464)

[Ecuación 2 17](#_Toc52017465)

[Ecuación 3 17](#_Toc52017466)

[Ecuación 4 17](#_Toc52017467)

[Ecuación 5 17](#_Toc52017468)

# Descripción general

El proyecto consiste en la implementación de un pedal multiefecto para guitarra eléctrica. El sistema será puesto en funcionamiento en una placa donde el usuario pueda interactuar con un potenciómetro simulando un pedal de expresión, y a través de un botón el cambio de los efectos. Además, se informará el efecto seteado a partir de un color de una luz led.

El proyecto implementará 3 efectos de audio: *Delay*, *Fuzz* y *Wah-Wah*. Pudiendo cambiar entre cada uno de ellos mediante un botón de forma secuencial, mientras que con un potenciómetro se variarán los parámetros característicos del efecto previamente elegido. El sistema será capaz de procesar en tiempo real la señal, aplicando un efecto.

El proyecto no implementará display para informar el efecto, el efecto seleccionado se informará mediante leds (los leds integrados en el board), un color por efecto.

El footswitch permitirá al usuario el efecto conocido como loopback que interconecta la entrada de señal con la salida de manera pasiva, cuando el pulsador este apagado.

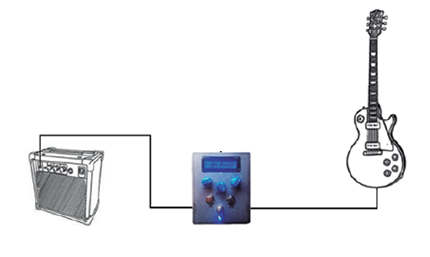


Figura 1 – Esquema basico de conexión

# Introducción teórica

## 2.1 Que es el Sonido

Una definición que describe la naturaleza senoidal del sonido es la siguiente: “Vibración transmitida a través de un material elástico, con frecuencias en el rango aproximado de 20 a 20000 Hercios” [1]. Gracias a esta definición los físicos y los ingenieros utilizan el “sonido” para referirse a una onda de presión que se propaga por el aire, algo que puede ser fácilmente medido, digitalizado y analizado. Por otra parte, desde un punto de vista perceptual del sonido puede referirse como: “Sensación producida en el órgano del oído por el movimiento vibratorio de los cuerpos, transmitido por un medio elástico, como el aire” [2]. A diferencia de la primera, esta definición se refiere a la percepción o sensación que ocurre dentro del oído, algo que es mucho más difícil de cuantificar. Basándonos entonces en la primera definición, las ondas sonoras se pueden representar como se muestra en la Figura 1, donde las zonas de altas presiones aparecen por encima de la “presión nominal” y las zonas de bajas presiones se muestran debajo. Esta forma de onda particular (onda senoidal) se puede caracterizar por tres términos matemáticos:

frecuencia, amplitud y fase. La frecuencia de una sinusoide es el número de oscilaciones completas que suceden en un segundo. Así si una onda con frecuencia de 100 Hz oscila

100 veces cada segundo. En lo correspondiente a una onda sonora las moléculas de aire se comprimen y estiran 100 veces en cada segundo [10].

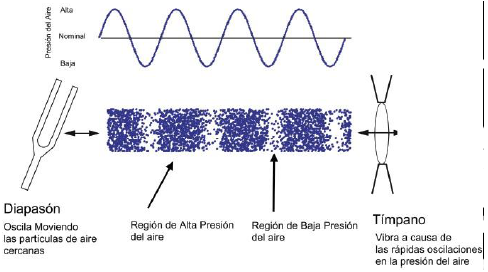


Figura 2: El sonido como una onda de presión

## 2.2 conceptos básicos del sonido

Como onda, el sonido responde a las siguientes características:

1. Es una onda mecánica: Las ondas mecánicas no pueden desplazarse en el vacío, necesitan hacerlo a través de un medio material (aire, agua, cuerpo sólido). Además, de que exista un medio material, se requiere que éste sea elástico. Un medio rígido no permite la transmisión del sonido, porque no permite las vibraciones. La propagación de la perturbación se produce por la compresión y expansión del medio por el que se propagan. La elasticidad del medio permite que cada partícula transmita la perturbación a la partícula adyacente, dando origen a un movimiento en cadena.
2. Es una onda longitudinal: El movimiento de las partículas que transporta la onda se desplaza en la misma dirección de propagación de la onda.
3. Es una onda esférica Las ondas sonoras son ondas tridimensionales, es decir, se desplazan en tres direcciones y sus frentes de ondas son esferas radiales que salen de la fuente de perturbación en todas las direcciones. El principio de Huygens afirma que cada uno de los puntos de un frente de ondas esféricas puede ser considerado como un nuevo foco emisor de ondas secundarias también esféricas, que como la originaria, avanzarán en el sentido de la perturbación con la misma velocidad y frecuencia que la onda primaria [12].

## 2.3 Los parámetros del sonido

Existen 6 propiedades o atributos que pueden describir lo que un ser humano experimenta al escuchar un sonido:

### 2.3.1 Tono o frecuencia

Se refiere a la rapidez de ocurrencia de las vibraciones que componen un tono. Cuando la frecuencia aumenta, también lo hace el tono.

### 2.3.2 Sonoridad

Indica cuan fuerte o suave se escucha un sonido, y se refiere a la amplitud de la onda vibratoria. La cantidad de compresión que sucede en las moléculas de aire, cuando el sonido viaje a través de ellas, no implica la rapidez a la que se desplazan, sino cuanta energía está almacenada en ellas.

### 2.3.3 Timbre

Se refiere a la calidad del sonido. Tanto como los instrumentos musicales y la voz humana producen sonidos que son ricos en vibraciones de diferente frecuencia y que son percibidos al mismo tiempo. La combinación de todas esas vibraciones, con diferentes amplitudes también, es lo que se percibe como timbre. Una forma sencilla de comprender este concepto es imaginar una flauta y la voz de una persona emitiendo un tono de la misma frecuencia. Fácilmente puede notarse la diferencia entre ambos sonidos. Colectivamente, el cerebro escucha lo anterior como un cambio en el color del sonido. El timbre es uno de los atributos del sonido más interesantes y complejos de comprender. Pero es en el timbre donde radica la naturaleza del diseño digital del sonido, pues tiene una gran facilidad para cambiar el color y el timbre del sonido rápidamente. Para hacer lo anterior en el mundo acústico, se tendría que cambiar algo del instrumento para cambiar el timbre. Gracias al diseño digital del sonido, la mayor parte de la música que actualmente escuchamos ha pasado por un proceso digital de refinación.

### 2.3.4 Percepción de duración

La duración no es un elemento fijo, sino algo que se percibe. Un ser humano puede escuchar un sonido, proveniente de cualquier lado, que dura unas pocas milésimas de segundo, o bien, varios minutos. Cuando se habla de lo que se está experimentando, cualquiera puede decir que percibió un tiempo lento o un tiempo rápido. Si se escucha una canción, ¿cómo se determina si la canción tiene un ritmo lento o rápido? Para ello, existe un punto estándar relativo, como referencia del tiempo, basado en los latidos del corazón. Los latidos del corazón rondan entre los 60, 70, y a veces hasta 80 latidos por minuto si es nervioso. Un sonido se percibirá como de tiempo rápido si va más rápido que los latidos y de tiempo lento, si va más lento. Así que cuando se habla de tiempo rápido y de tiempo lento, el reloj interno del cuerpo es el encargado de proveer la información.

### 2.3.5 Envolvente o articulación

Muestra la forma del sonido en el dominio del tiempo basada en la forma en que se interpreta un instrumento musical, incluyendo a la voz humana. La articulación se refiere a lo que sucede en los primeros milisegundos de un sonido, y a la cantidad de energía que se aplica a la nota. Por ejemplo, con un violín se puede aplicar una gran cantidad de energía al tocar una nota, si se ejerce una presión inicial mayor con el arco sobre la cuerda. Luego, se libera la mayor parte de la presión haciendo que el sonido se estabilice hasta que suavemente desaparece. Con un piano, la articulación es diferente, pues inicialmente se escucha un martillazo, y luego, la nota que suavemente desaparece. Así, el piano tiene una articulación percusiva, que gráficamente, representa un impulso inmediato. La envolvente es importante para el diseño digital del sonido. Pues conociendo las envolventes de cada instrumento, se pueden imitar los mismos. Y entonces, por medio de una computadora, se puede tener todos los instrumentos musicales.

### 2.3.6 Difusión

Se refiere a capacidad del cerebro para localizar espacialmente la fuente de un sonido. El cerebro tiene la capacidad de procesar simultáneamente múltiples sonidos de diferentes fuentes y, ubicar espacialmente los mismos consciente o inconscientemente.

Por ejemplo, si se está en la calle hablando con un amigo, uno le pone mayor atención a la conversación. Cuando se llega a una intersección, inconscientemente se procesan sonidos de ciertos automóviles provenientes de ciertas direcciones. Y en ése momento, se está teniendo la conversación, se están ubicando las fuentes de sonido de los coches, y se está determinando si los mismos se están acercando o alejando. La difusión es un atributo importante en el diseño digital del sonido, por ejemplo, cuando se está escuchando una película en el cine. La experiencia auditiva resulta ser más real en el cine por que el sonido proviene de diferentes fuentes. Esto hace que una persona sienta que forma parte de la acción, y que sienta que forma parte de la película. Aquí, el sonido es representado por una configuración de canales (2.1, 5.1, 7.1). Una configuración 5.1 implica la existencia de 6 fuentes de sonido; ; y una de 2.1, sólo 3 fuentes. Ésta última configuración es mejor conocida como sonido estéreo (altavoces izquierdos, derecho y subwoofer). [11]

# Estado del Arte

En este apartado se definirán de forma teórica los la basta cantidad de efectos de un pedal y la manipulación matemática que sufre una señal de información para modificarla y obtener la señal esperada. Por otro lado, también se pretende que el usuario empiece a familiarizarse con los efectos antes de empezar a leer más sobre ellos. De tal manera que con una breve descripción pueda ser capaz de clasificarlos y ver sus diferencias características respecto a otros efectos. También se detallarán en este apartado algunos conceptos que van ligados a la implementación de los efectos ya que algunos efectos para poder quedar terminados como tal necesitan alguna pequeña ayuda, modulación, filtrado etc. Para poder obtener el efecto deseado.

## Clasificación de los efectos

Un efecto de guitarra es un circuito analógico o digital que audiblemente modifica una señal de entrada producida por un instrumento para producir una salida deseada [10].

Los efectos en general se pueden clasificar en categorías específicas [9]:

* Efectos basados en el rango dinámico
* Distorsiones
* Ecualizadores
* Efectos de Modulación
* Efectos de Transposición de tono
* Efectos de retrasos
* Simuladores
* Reductores de ruido

### Efectos de rango dinámico

El rango dinámico de un equipo de sonido, es la diferencia existente entre los sonidos más potentes y menos potentes que el equipo es capaz de reproducir. Los efectos tales como compresor y expansor son efectos basados en el rango dinámico [3,4].

### Distorsiones

Cuando hablamos de la distorsión de un amplificador o una pedalera nos referimos a la distorsión armónica. La “distorsión armónica” es un parámetro técnico utilizado para definir la señal de audio que sale de un sistema. Dicha distorsión se produce cuando la señal de salida no equivale a la señal que entro en él. Esta falta de linealidad afecta a la forma de onda, porque el equipo introduce armónicos que no estaban en la señal de entrada. Puesto que son armónicos, es decir múltiplos de la señal de entrada esta distorsión no es tan disonante y a veces es más difícil de detectar. Efectos como distorsiones pueden ser tales como el Fuzz, Bitcrusher, PhaserDistorsión, distorsión, así como otros tipos de distorsiones derivados es una distorsión simple, aunque este tipo de distorsiones suelen ser más específicas para el tipo de sonido que se quiera buscar [10].

### Ecualizadores

Dentro de los ecualizadores tenemos dos tipos, ecualizadores activos y pasivos.

Los ecualizadores pasivos son los que se pueden encontrar en los amplificadores. Donde con pocos parámetros son capaces de tratar la señal. Los graves dan cuerpo a la señal, es decir dan peso, para una guitarra los graves nos es una frecuencia que nos deba preocupar excepto si tenemos ausencia de una señal de bajo [8]. Los agudos nos dan brillo a la señal, es decir, cuanta más distorsión tengamos más destacaran los agudos. Por el contrario, cuanta menos distorsión tengamos menos debemos de preocuparnos por los agudos [7]. Los medios es la frecuencia que más afecta al sonido de una guitarra. Para el sonido de una guitarra cuantos menos medios tengamos obtendremos un sonido más suave, más redondo, menos ruidoso, aunque se necesitaran más vatios en un equipo de sonido para poder oírlos con claridad. Por el contrario, si contamos con más medios el sonido sonará como más “duro” y algo más ronco a la vez que nos facilitara la mezcla de ese sonido con otros. Efectos basados en ecualización los más conocidos son el “wha-­wha” y su derivado el “auto-­wha” [10].

### Efectos de modulación

Modulación es un efecto no lineal en el cual varias señales interactúan unas con otras para producir nuevas señales con frecuencias que no estaban presentes en las señales originales. Hay muchas formas de modulación, pero para este caso solo utilizaremos dos de ellas, la modulación de frecuencia y de amplitud. La modulación de frecuencia (FM) es la variación en frecuencia de una señal, debido a la influencia de otra señal, generalmente de frecuencia más baja. La modulación de amplitud (AM) es un tipo de modulación lineal que consiste en hacer variar la amplitud de la señal portadora de forma que esta cambie de acuerdo con las variaciones de nivel de la señal que contiene la información que se desea transmitir, llamada señal moduladora o modulante. [4] También hay que indicar que los efectos de modulaciones hacen uso de un LFO, Low Frecuency Oscilator (Oscilador de baja frecuencia). Los LFO's son primariamente usados como moduladores para otras señales, y no para generar audio en sí [9]. En el mundo de los osciladores se considera "*baja frecuencia*" cualquier frecuencia por debajo de los 20 Hz, que es la frecuencia más baja que puede escuchar un humano. Aunque los LFO's no están destinados para ser usados como fuente de sonido, en algunos sintetizadores pueden operar a frecuencias más allá de los 20 Hz.

Efectos basados en modulación nos encontramos con efectos como el Chorus, el Flanger, el Phaser, el Tremolo o el Vibrato [6].

### Efectos de transposición de tono

Para transponer el tono, la señal que entra en un equipo se muestrea y posteriormente se reproduce más deprisa para subir el tono o más despacio para bajarlo. Efectos basados en la transposición tenemos el Octavador que lo que hace es añadir a la señal octavas por arriba o por abajo. El Armonizador el cual añade a la señal su copia transpuesta o el Detuner el cual desafina la señal y la añade a la señal original [7].

Hay que indicar que estos efectos no son de los más utilizados debido a que se necesitan grandes niveles de composición y armonización para poder usarlos correctamente.

### Efectos basados en retardos

Este tipo de efectos están basados en el uso de muestras y/o salidas pasadas para generar la salida en el momento actual. Hay que señalar que muchos efectos, aunque no se encuentre dentro de esta clasificación también usan retardos como puede ser el Chorus. Los efectos más puros basados en retardos son el Delay, el Eco, el Flanger y el Reverb o Reverberación [7].

### Efectos simuladores

Los efectos basados en simuladores no son de gran interés, ya que simplemente emulan tipos de amplificadores, altavoces, tipos de guitarras o de pastillas. Son especialmente utilizados en pedaleras digitales. Estos efectos carecen de gran interés de ahí su poca utilización y venta de este tipo de pedaleras [9].

### Reductores de ruido

Los efectos de reducción de ruido “cierran” el paso de toda señal que no supere un determinado umbral fijado por el usuario. Son muy útiles en situaciones de “directo” en las que hay multitud de micrófonos que pueden captar lo mismo que el principal, y tratamos de que la señal sólo entre por el principal. También nos ayudan a “recortar” todos aquellos ruidos no deseados que se han colado en una grabación (toses, respiraciones, rozamientos de ropas, ruidos de ambiente), siempre que no se mezclen con la señal principal. El efecto más conocido de este tipo es el Noise Gate o puerta de ruido, el cual es muy utilizado para tocar en directos [9].

En el cuadro comparativo dela figura 3, se pueden ver los efectos según su clasificación de efecto.



Figura 3 – Clasificación de efectos

# Análisis y desarrollo de los efectos

En este apartado vamos a concentrarnos en el análisis de los efectos que nos competen para el desarrollo de nuestro pedal multiefecto de guitarra eléctrica, cabe remarcar que los mismos son: Delay, Fuzz y Wah-Wah.

## Delay

### Introducción

El delay es el efecto más básico y simple de los efectos basados en retardos, siendo uno de los efectos más clásicos y comunes en muchas aplicaciones de audio, estando presenta hasta en la producción musical.

Se utiliza en todo tipo de música desde música electrónica hasta electroacústica, así como en los sistemas de refuerzo de sonido. Pudiendo ofrecer gran variedad de aplicaciones que van desde fines puramente artísticos hasta otras aplicaciones de mejora del sonido [3].

A pesar de ser un efecto muy simple es un efecto muy importante, no solo como efecto sino como elemento potencial para otros efectos como el chorus o el flanger como se vio anteriormente, así como para la reverberación [4].

### Implementación

Para la implementación del delay no tiene mayor complicación, ya que básicamente se trata de coger una muestra se la señal original desplazada y escalada y sumada a la señal original. La figura 4 muestra el diagrama de bloques de un efecto Delay.

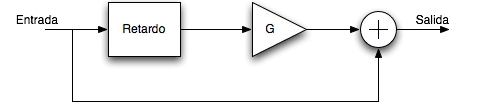


Figura 4 – Esquema basico delay

Como puede apreciarse en el diagrama de bloques a la señal retrasada se le aplica una amplificación, normalmente suele ser para atenuar un poco el sonido de la señal haciéndose notar más o menos la intensidad de las muestras repetidas. El tiempo de delay para retrasar nuestra señal puede variar desde unos pocos milisegundos hasta varios segundos. Dependiendo del tiempo de retraso elegido el efecto delay sonara de una manera u otra, esto es con un tiempo de retraso de unos 40-­80 ms el efecto delay

sonara como un efecto de coro poco elaborado. Es con un tiempo de retraso entre 100 y 120 ms cuando sonara como un efecto delay. También indicar que para tiempos mayores a 1 segundo permite al músico tocar melodías sobre sí mismo.

Tras esto podemos decir que el efecto delay toma una señal de audio y la reproduce de nuevo después de un tiempo de delay, escalada y mezclada con la señal actual. Siendo las ecuaciones que definen al delay [10]:

*señal\_de\_salida(n) = señal\_de\_entrada(n) + amplitud\* señal\_de\_entrada(n-­retraso)*

Los retrasos se pueden experimentar en espacios acústicos. Una onda de sonido reflejada por una pared se superpondrá a la onda de sonido en la fuente. Si la pared está lejos, tal como un acantilado, oiremos un eco. Si la pared está cerca de nosotros, notaremos los reflejos a través de una modificación del color del sonido.

Pueden aparecer reflejos repetidos entre límites paralelos. En una habitación, tales reflexiones serán un eco más plano. La distancia entre los límites determina el retraso que se impone a cada una de la onda de sonido reflejada. En un cilindro, se desarrollarán reflexiones sucesivas en ambos extremos. Si el cilindro es largo, escucharemos un patrón iterativo, mientras que, si el cilindro es corto, oiremos un tono agudo.

Los equivalentes de estos fenómenos acústicos se han implementado como unidades de procesamiento de señales. La variación del tono se debe al hecho de que la distancia entre la fuente y nuestros oídos está siendo variada al variar la distancia, para nuestra aplicación, sería el equivalente a variar el retraso del tiempo. Si seguimos variando periódicamente el tiempo de retraso produciremos una variación periódica del tono. Esto es sencillamente un efecto de vibrato.

### Estructura básica de un Delay

La red que simula un solo retraso se llama Comb Filter[[1]](#footnote-1) (filtro de peine FIR). La entrada

la señal se retrasa por una duración de tiempo dada. El efecto será audible solo cuando

la señal procesada se combina (agrega) a la señal de entrada, que actúa aquí como

una referencia. Este efecto tiene 2 parámetros de ajuste: la cantidad de tiempo de retraso “T” y la amplitud relativa de la señal retrasada a la de la señal de referencia.

La ecuación de diferencia y la función de transferencia están dadas por:

Figura 5 – Ecuacion 1 Delay

Con:

Figura 6 - Ecuacion 2 Delay

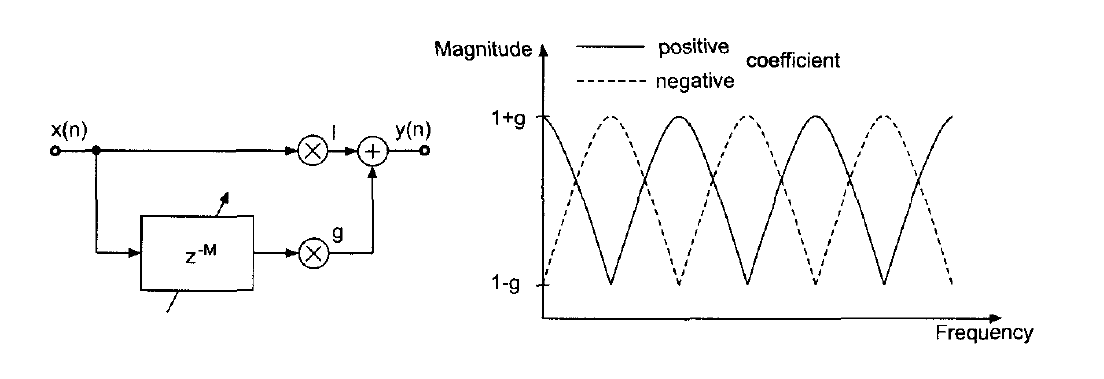


Figura 7 – Gráfico del desfasaje de señal con Delay

## Fuzz

### Introducción

Una guitarra eléctrica distorsionada es esencial en la música rock. El Fuzz es un efecto clave en este sentido. El Fuzz fue uno de los primeros efectos en aparecer, usado por músicos como Eric Clapton o Jimmy Hendrix. Dicho efecto trabaja sobre la ganancia de la señal y se caracteriza por tener un sonido suave y un poco más áspero en comparación con una distorsión, es muy similar al sonido que pude producir una guitarra eléctrica sonando por un amplificador saturado con los altavoces rotos [5].

### 4.2.2 Implementación

La implementación de este efecto se basa simplemente en recortes. A partir de un umbral indicado. Consiguiendo así convertir una señal de entrada sinusoidal en una forma de onda mucho más parecido a una onda cuadrada creando así un efecto áspero en el sonido, creando un efecto Fuzz clásico. Dándole a la canción un sonido más sintético que una distorsión normal.

La amplitud de la señal de entrada se recorta por arriba y por debajo a partir de cierto umbral establecido, y posteriormente automáticamente será amplificada para compensar cualquier perdida de la señal. Dicho umbral de la señal estará comprendido entre 0-­1 al ser introducido por el usuario, pero realmente representa un porcentaje de valor absoluto del máximo de nuestra señal.

La figura 8 muestra el esquema de un efecto Fuzz. La amplitud la cual se amplificará la señal vendrá dada como la inversa de nuestro umbral menos uno. Asegurando así que nuestra amplificación estará comprendida entre 0.99 y -­0.99. Quedando, así como únicos parámetros a introducir por el usuario el umbral de la señal [10].

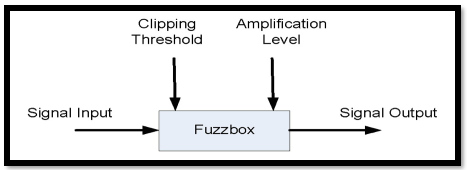


Figura 8: Diagrama de efecto Fuzz [4]

El efecto fuzz básicamente aplica sobre la señal un recorte (clipping) como el que haría un amplificador extremadamente saturado. Los picos de la señal de entrada quedan recortados, y los valles aplanados, generando el característico sonido distorsionado.

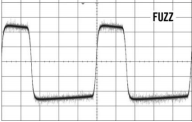
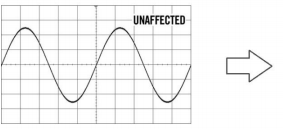


Figura 9 – Una señal sin afectar y afectada por el efecto Fuzz

El fuzz es un overdrive[[2]](#footnote-2) exagerado, en la figura 10 se puede ver como es la forma de onda de un overdrive, con una mayor distorsión de la onda que hace que existan más picos en los armónicos. Esto genera un sonido más grueso y denso que se focaliza en las frecuencias bajas y medias, pero no tanto en las más agudas (un fuzz es más grave y menos chillón que una distorsión). Sin duda es un sonido menos nítido y más sobrecargado. De hecho, muchos guitarristas, con tendencia a un sonido más definido, huyen de este efecto, mientras que otros lo prefieren.

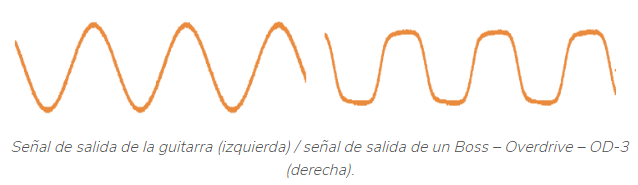
[[3]](#footnote-3)

Figura 10 – Salida de una señal con Overdrive

### 4.2.3 Procesamiento de la señal

**Symmetrical Soft Clipping** (Recorte suave simétrico): para simulaciones de overdrive, se debe realizar un recorte suave asimétrico de los valores de entrada. Un posible enfoque para una saturación suave no lineal viene dado por:

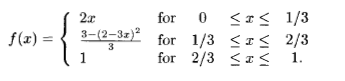


Figura 11 – Formula tipica Fuzz

La relación entrada / salida estática se muestra en la figura 11 hasta el umbral de 1/3, la entrada se multiplica por dos y la curva característica se encuentra en su región lineal. entre valores de entrada de 1/3 hasta 2/3, la curva característica produce una compresión suave descrita en el término medio de la ecuación. Por encima de los valores de entrada de 2/3, el valor de salida se establece en uno.

La figura 12 muestra la forma de onda de una simulación con la curva característica descrita anteriormente y una sinusoide de 1Khz.

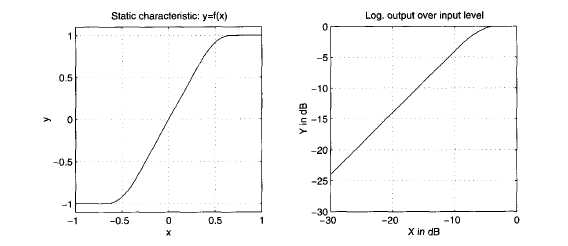


Figura 12 – Alinealidad de la curva

Desde el punto de vista analógico, normalmente el circuito se compone de dos o tres transistores capaces de impulsar la señal a un nivel alto antes de limitarla, haciendo que se vuelva más cuadrada en lugar de curva. El twang[[4]](#footnote-4) resultante se crea especialmente de frecuencias medias y altas.

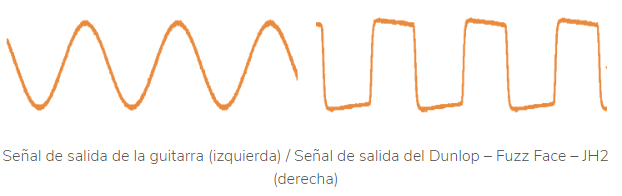
[[5]](#footnote-5)

Figura 13 – Salida de una señal con Fuzz

En general, el fuzz produce diferentes niveles de salida dependiendo de los circuitos, pero estas frecuencias son siempre bastante sensibles.

## Wah-Wah

### Introducción

El wah-wah es un tipo de efectos que altera el tono y la frecuencia de la señal de guitarra para crear un sonido distintivo, imitando la [voz humana](https://es.qwe.wiki/wiki/Human_voice) diciendo que el [onomatopéyico](https://es.qwe.wiki/wiki/Onomatopoeia) nombre de "wah-wah". El efecto se extiende por el pico de respuesta de un filtro de frecuencia arriba y hacia abajo en frecuencia para crear el sonido, un [deslizamiento espectral](https://es.qwe.wiki/wiki/Spectral_glide) , también conocido como "el efecto wah". Su origen en la música jazz, pues era el nombre que se le daba al sonido de la trompeta cuando se usaba la sordina. Esto fue luego simulado con circuitos electrónicos para la guitarra eléctrica cuando se inventó el pedal wah-wah. El mismo está controlado por el movimiento del pie del guitarrista en un pedal de oscilación conectado a un [potenciómetro](https://es.qwe.wiki/wiki/Potentiometer).

### Implementación

A diferencia de los filtros pasa altos y pasa bajos, que atenúan el espectro del audio por encima o por debajo de una frecuencia de corte. Los ecualizadores dan forma al espectro de audio al mejorar ciertas bandas de frecuencia mientras que otras permanecen afectadas. Están construidos mediante una conexión en serie de filtros limitadores de primer y segundo orden y filtros de pico, que son controlado independientemente, ver figura 14. Los filtros limitadores aumentan o reducen el nivel bajo o alto bandas de frecuencia con la frecuencia de corte de parámetros fc y ganancia G. Los filtros de pico impulsan o cortan bandas de frecuencia media con parámetros de frecuencia de corte fc, ancho de banda fb y ganancia G. Un tipo de filtro usado con frecuencia es el filtro de pico Q constante. El factor Q se define por la relación del ancho de banda a la frecuencia de corte Q = k. El corte La frecuencia de los filtros de pico se ajusta, manteniendo el factor Q constante. Esta significa que el ancho de banda aumenta cuando aumenta la frecuencia de corte y viceversa. [13]

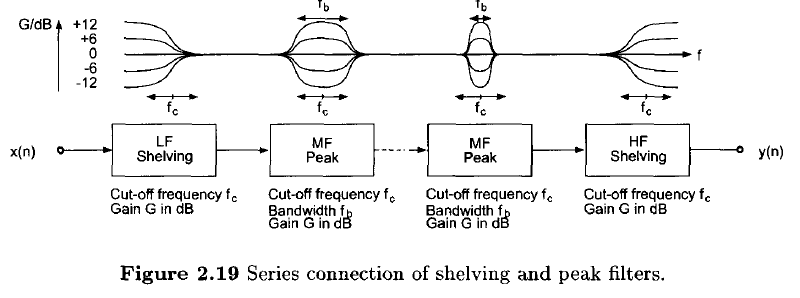


Figura 14: Conexión serie de en bloque y filtros Peak

Los filtros paramétricos permiten controlar los parámetros del filtro, la ganancia, frecuencia de corte y ancho de banda o factor Q.

El efecto es producido por procesadores de señal que son controlados por el pie. Contienen un filtro de banda ancha que posee una frecuencia central de resonancia variable y muy selectiva (un pequeño ancho de banda). El mover el pedal se está modificando la frecuencia de corte central de la banda de paso. En consecuencia, el wah-wah se mezcla con la señal que es conectada de forma directa, como muestra la figura 15. Este efecto provoca una formación del espectro similar al habla humana y produce un discurso como el sonido "wah-wah".

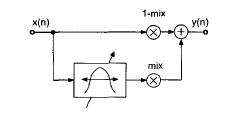


Figura 15 Wah Wah: Filtro pasa banda variable en el tiempo

Si la variación de la frecuencia central es controlada por la señal de entrada, se usa un oscilador de baja frecuencia para cambiar la frecuencia central. Tal efecto se llama filtro auto-wah. Si el efecto se combina con una variación de amplitud de baja frecuencia, que produce un trémolo, el efecto se denomina filtro de tremolo-wah. Reemplazar la unidad de retardo en el filtro de paso de banda por un M tap delay implica implementar un filtro M-fold wah-wah. Un M-fold wah-wah, se puede describir como un efecto en el que se aplican múltiples filtros wah wah a una señal, cada uno en un cierto rango de frecuencia como se muestra en la figura 16.

Los filtros de paso de banda M son extendidos por todo el espectro y cambian simultáneamente su frecuencia central. La figura 17, contiene una tabla con varias configuraciones de parámetros para diferentes efectos.

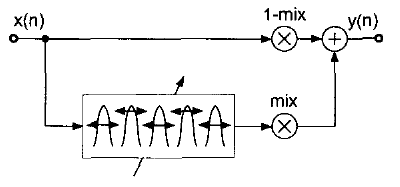


Figura 16: Filtro Wah-Wah

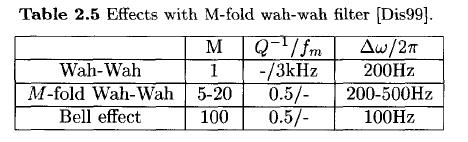


Figura 17: efecto con filtro M-fold wah-wah

# Descripción técnica del proyecto.

## Hardware

En esta sección se detallará el funcionamiento interno de los protocolos y drivers implementados para el proyecto.

Con la Placa STM32F411 se ha podido cubrir la mayoría de las necesidades asociadas al proyecto. Con la salvedad de implementar un filtro pasa bajo pasivo de orden 2 para oficiar de filtro anti-alias, además un divisor resistivo para montar la señal en continua y un Jack de ingreso para el audio.

### Protocolo I2S

El protocolo **I2S** conocido también como **Inter-IC Sound**, **Integrated Interchip Sound**, es un estándar eléctrico de bus serial usado para interconectar circuitos de audio digital. El bus I2S separa las señales de datos y de reloj, lo que resulta en menores cantidades de fluctuación (jitter en inglés) de la señal que en sistemas que recuperan el reloj de la señal de datos. Ver figura 18.

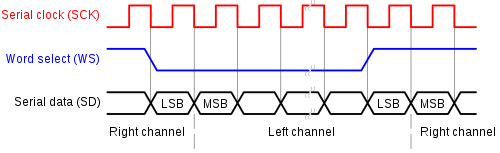


Figura 18 – Grafico I2S

Multimedia 🡪 I2S3 🡪 GPIO settings

Para que se escuche un sonido libre de glitches, es necesario que el I2S para el DAC y el ADC, trabajen a frecuencias lo más cercanas posible.

A continuación, se describirán los pines que necesita la comunicación I2S

* **MCK:** (Main Clock) es el clock principal que contiene la mayor frecuencia, este es enviado por el master y recibido por el/los esclavos, en este caso el master es el micro STM32F411 y el esclavo es el CS43L22 (driver), su frecuencia es N\*WS; siendo N un numero estandarizado (ej 128, 256, 512) *Ver sección 7.4.2, página 39 manual*.
* **SCK**: (system clock) es el BitClock (menor frecuencia que el primero) frecuencia de C/Bit, su frecuencia es Fs\*(#de bits por canal) \*2.
* **WS:** también se lo llama LRCK (left right clock) este sería el Fs que se menciona en el "SCK", cuando está en estado alto se realiza la conversión del canal izquierdo y cuando está en estado bajo en el canal derecho.
* **SD:** (Serial Control Data) Son los datos.

Para comunicarse con el DAC "CS43L22", debe generarle el Master Clock para que pueda filtrar correctamente ya que se trata de un DAC delta-sigma.

Según la hoja de datos, cuando el Master Clock del I2S es generado, La Fs se calcula con la siguiente fórmula:

Ecuación 1

Por otro lado, I2SxCLK se calcula como:

Ecuación 2

Para una Fs=32KHz recomienda:

Ecuación 3

Ecuación 4

Ecuación 5

Este seteo genera una Fs=32001.20117Hz. (con un error del 0.0038% respecto a los 32KHz). El Master Clock (MCK) esta pre configurado a una frecuencia igual 256\*Fs= 8.192Mhz [16]. Como el I2S está configurado para 16 bits, SCK = WS\*16 => SCK = 512.019 KHz. El I2S en el micro está configurado como master, es decir, genera el Master Clock.

### Driver CS43L22

El CS43L22 es un convertidor digital / analógico (DAC) estéreo de baja potencia con auriculares integrados y amplificadores de altavoces de Clase D que funcionan desde un núcleo analógico y digital de bajo voltaje. La ruta de salida DAC incluye un motor de procesamiento de señal digital con varios controles de función fija, como el control de tono y volumen, e incluye el énfasis, funciones limitantes y un generador de pitidos que ofrece tonos seleccionables en un rango de dos octavas. El amplificador de auriculares estéreo se alimenta de un suministro positivo separado, y la fuente de carga integrada proporciona un suministro negativo que permite una salida analógica centrada en tierra para eliminar la necesidad de condensadores de bloqueo de CC. El altavoz estéreo de clase D no requiere un filtro externo y puede alimentarse directamente de una batería [14]. Para el Proyecto se implementó el Driver CS43L22 integrado al Board de la placa:

Connectivity 🡪 I2C1

El DAC necesita primero de la configuración del I2C para establecer la comunicación entre el DAC externo y el micro, y así configurar los parámetros pertinentes: frecuencia, canales, volumen, mute, etc. Luego se comunica por I2S para enviar los datos.

* **I2S**

El driver está configurado como Slave, es decir, no genera Master Clock, es generado por el microcontrolador.

### ADC

* El ADC utiliza el Timer 3 para ser disparado. El Timer 3 está seteado a 32kHz
* 12 bits
* 2 canales (para el audio y el potenciómetro)
* La frecuencia de Sampling es seteada por el Timer 3 el cual depende del APB1 Timer Clock
* Fs=APB1 Timer Clock/(Prescaler+1)/(Counter Period+1)
* Fs=32Khz=96Mhz/1000/3.

De esta manera el ADC e I2S (Controla el DAC) trabajan a la misma frecuencia.

Teniendo en cuenta que se dispone de un solo ADC y se requieren medir dos entradas (ADC y potenciómetro), se deben configurar 2 canales para medir secuencialmente:

* + Resolución = 12 [bits]
  + Sampling\_time (ADC) = 3 [ciclos]
  + Sampling\_time (potenciómetro) = 3 [ciclos]
  + DMA: Modo=Circular, Palabra=HalfWord (16bits), Frame=2048 samples (por canal)

### TIMER

El **Tim3** referenciado con APB1[16] para el Clock & Configuration del Cube, fue seteado a 32khz para disparar el ADC como se menciona en la sección *5.1.3 ADC*.

El **Tim9** referenciado con APB2[16] para el Clock & Configuration del Cube, fue seteado a 1hz para que el parpadeo lógicamente de 1 segundo del led Rojo sea el testigo de que el proyecto está corriendo de forma correcta.

### Configuración de Clock

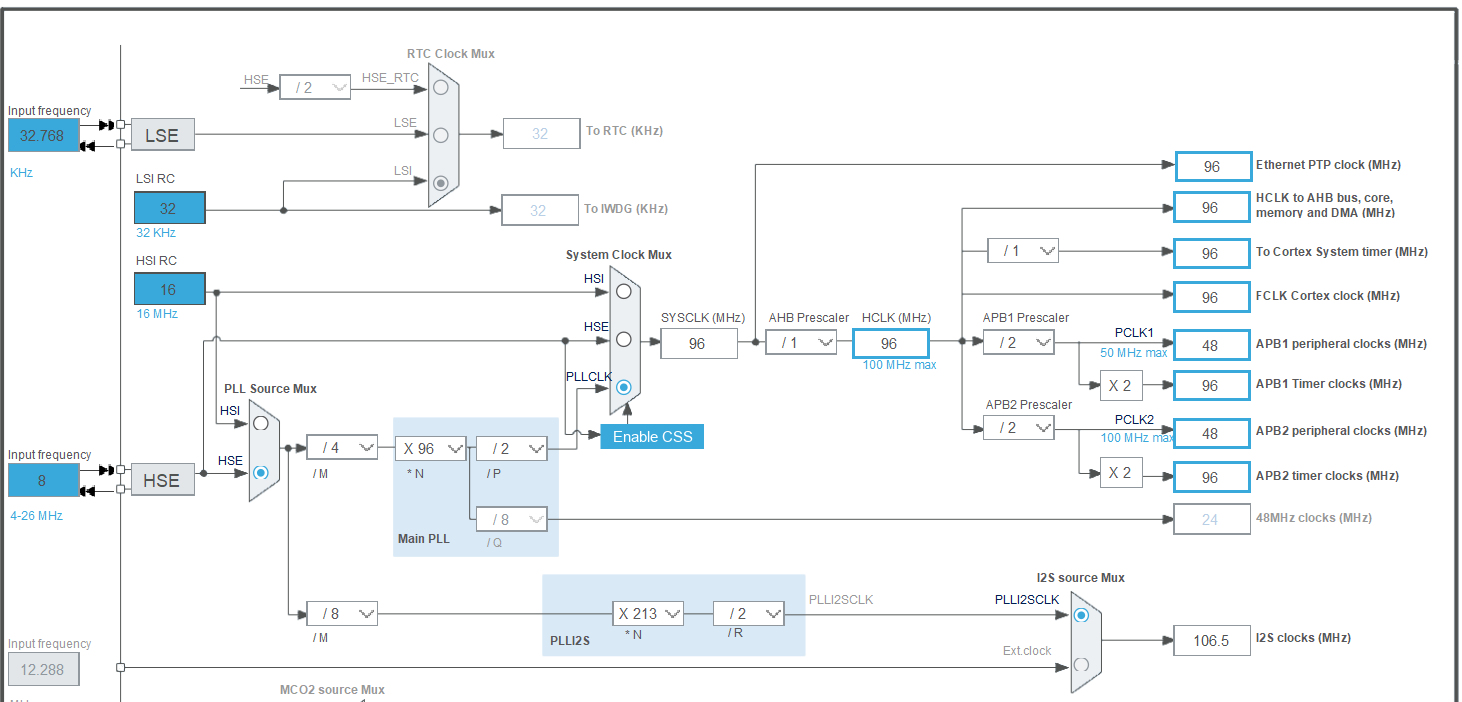


Figura 19 – Clock & Configuration (CUBE)

### Configuraciones de pines

En esta sub sección se mostrarán los pines empleados por el microcontrolador.

#### Pulsadores para el usuario

*PA0:* (Pulsador interno para modificar efecto)

*PD9:*(Pulsador Interno para hacer un reset)

#### ADC

*PA1*: (Entrada de Audio)

*PA3*: (Control de expresión de efectos)

#### Configuración con el Códec de DAC

*I2C1\_SCL:* PB6

*I2C1\_SDA:* PB9

#### Transmisión de audio al códec del DAC

*I2S3\_WS***:** PA4

I2S3\_*MCK***:** PC7

I2S3\_SCK**:** PC10

I2S3\_SD: PC12

### Diagrama en bloques

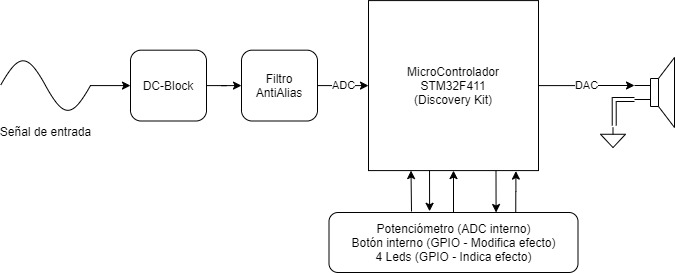


Figura 20 – Diagrama en bloques

El conjunto de este diagrama en bloques, explica el funcionamiento de un pedal multiefecto. El mismo tiene una adaptación en la señal de entrada para poder ingresar al ADC del microcontrolador, seguido de un filtro AntiAlias, para eliminar el contenido armónico no deseado. El microcontrolador es el encargado de realizar el procesamiento digital de la señal, agregando los efectos (LoopBack, Delay, Fuzz y Wah-wah).

### Esquemático

En esta sección se hará la descripción de forma separada de cada uno de los bloques que conforman el sistema.

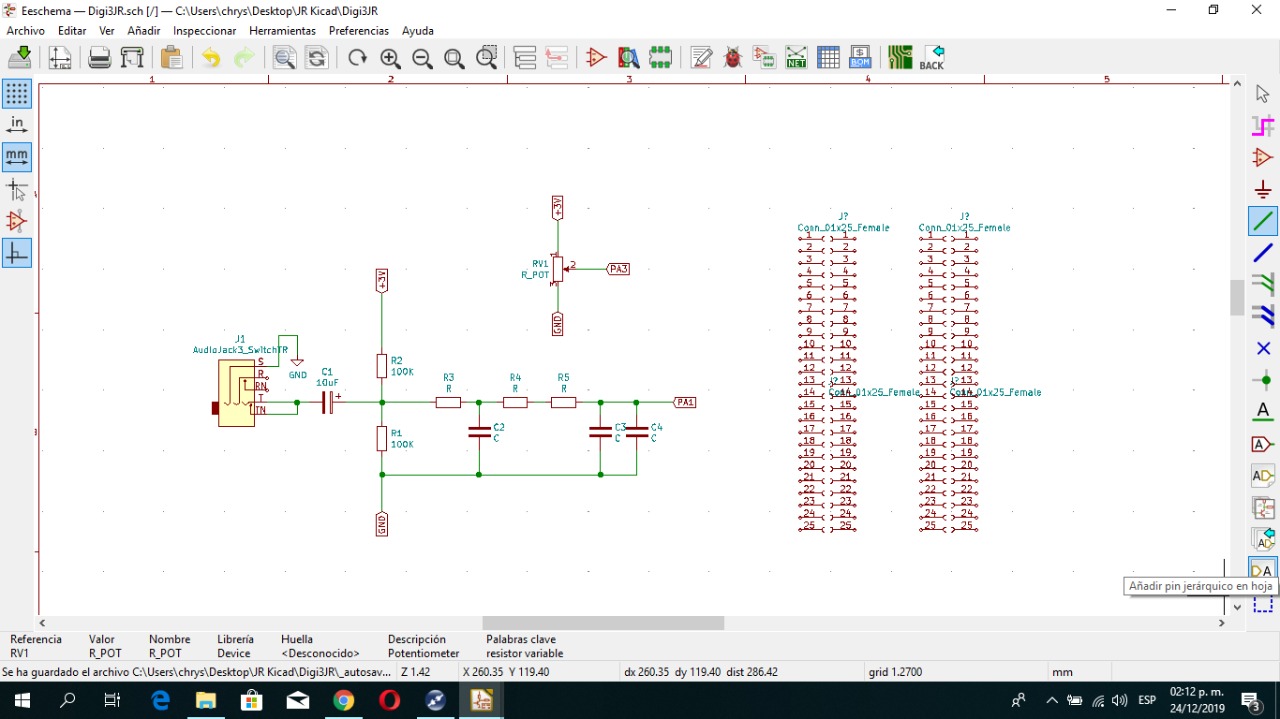


Figura 21 – Esquemático general

#### Jack

Comenzaremos con la descripción de este conector, es el conector de audio​ de señales analógicas, que se utiliza para Inyectar la señal de entrada. La misma es un conector mono canal dado que la aplicación es para instrumentos.

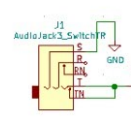


Figura 22 – Jack de entrada

#### Divisor resistivo

Este divisor se implementó para que el punto medio de la señal inyectada esté en 0 (en el centro), además se incluyó un capacitor para bloquear la componente continua de la señal.

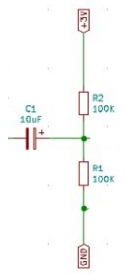


Figura 23 – Divisor resistivo

#### Potenciómetro

Este potenciómetro se implementó para realizar las expresiones de los efectos.

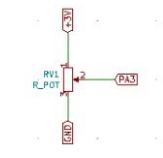


Figura 24 – Conexionado Potenciometro

#### Filtro Pasa Banda

Se experimentó la necesidad de incluir un filtro pasa bajo de orden 2, con frecuencia de corte en 8khz, dado que el ADC interno esta seteado a 32Khz, al tener este valor la performance del audio no es buena, incluyendo un nivel de ruido notable.

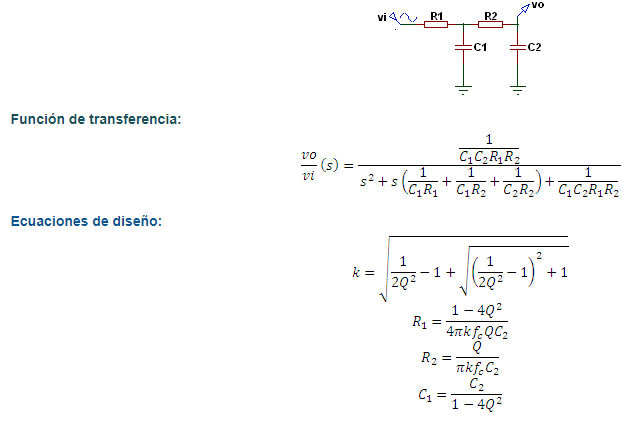


Figura 25 – Ecuacion de filtro pasa bajo

##### Valores del Filtro pasa bajos Pasivo:

* Frecuencia de corte Fc= 8 KHz
* Factor de calidad Q= 0.33

##### Componentes del filtro Pasa bajos:

* R1= 6288 [Ω]
* R2=4853 [Ω]
* C1= 1.77 [nF]
* C2= 1 [nF]

#### Circuito impreso

Este proyecto incluye un hardware adicional muy sencillo ya que solo integra el Jack para inyectar la señal de entrada, un filtro pasa bajos y un potenciómetro oficiando como pedal de expresión.

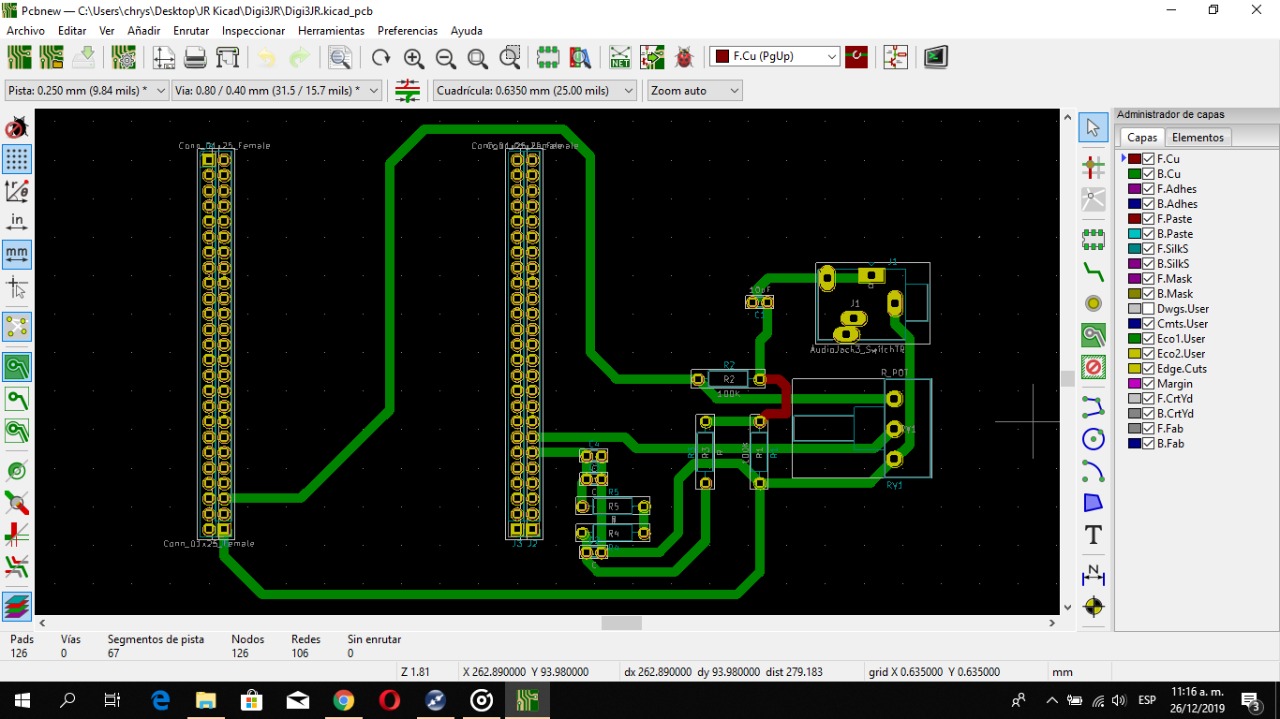


Figura 26 – Routeo de pistas

#### Modelo 3D:

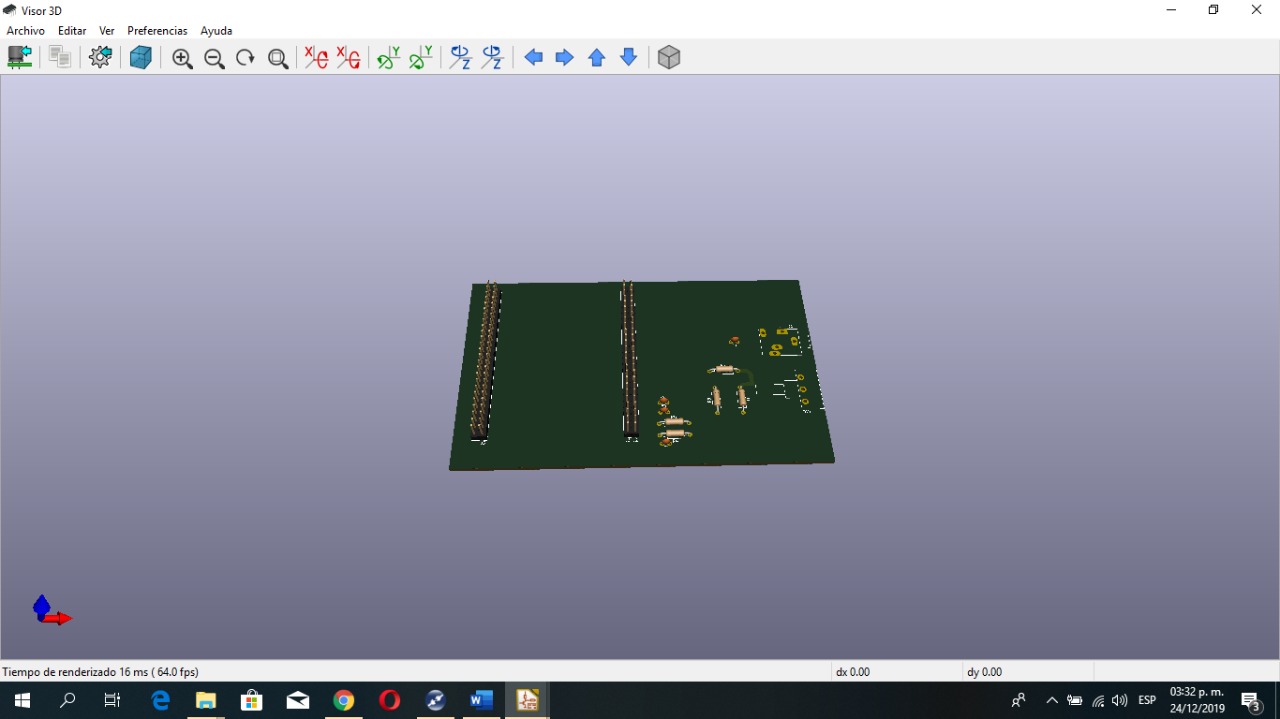
****

Figura 27 – modelo 3D Placa

# Software

En esta sección se verán las distintas rutinas y FSM (Finite State Machine) utilizadas junto con un diagrama de flujo y su explicación.

### Descripción del funcionamiento de cada rutina

En esta sección se desea mostrar la enumeración de todas las rutinas incluidas en el programa y la función que cumplen en el mismo en forma resumida.

* Rutina main ()

En el main\_init inicializo y disparo los periféricos necesarios para este proyecto. Inicializa y dispara Timers (TIM3 y TIM9), configura el DAC CS43L22 por I2C e inicializa la comunicación por I2S, configura los GPIO, genera buffers de nivelación de la señal y configura los filtros para el efecto Wah-Wah. Luego se llama a la función main\_loop () dentro de un While (1). Esta función contiene un FSM para aplicar el efecto seleccionado.

* main\_loop ()

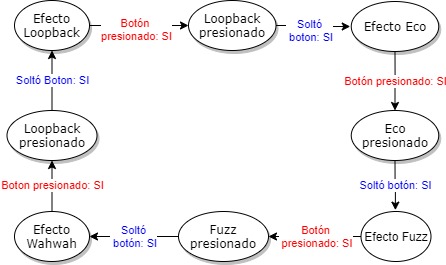


Figura – Maquina de estado de efectos

Dentro de main Loop esta Finite State Machine (FSM) con los 4 efectos 2 estados por efecto, uno para cuando se pulsa el botón y el otro cuando se suelta.

Esta FSM solamente actúa cuando recibe muestras del ADC (transmit\_ready==DMA\_ADC\_READY) y durante su ejecución mantiene transmit\_ready=DMA\_LOGIC\_BUSY para indicarle a la FSM del DMA que el buffer de entrada está siendo procesado. Al finalizar lo libera con transmit\_ready=DMA\_LOGIC\_READY (Esto no fue incluido en el diagrama para una interpretación más sencilla).

Al comienzo y final del Main\_loop () se agregaron dos métodos, main\_loop\_start () y main\_loop\_end () respectivamente, comunes a todos los efectos.

**main\_loop\_start ()** 🡪 Obtiene el valor del potenciómetro, duplica el audio de entrada hacia ambos canales (Right y Left). Ademas, se crea un buffer para el efecto del wahwah a partir de solo un canal ya que el mismo asi lo requiere.

Por otro lado, se resta el buffer de entrada *buffer\_DMA* (es el buffer donde se guardan las muestras que provienen del ADC) al buffer de nivelación *nivelacion\_in*. Teniendo en cuenta que el ADC trabaja en 12 bits dicho buffer se completa con 0x0800, luego se desplaza la señal a 16 bits para aprovechar todo el rango. Finalmente se inicializa el buffer utilizado para el efecto Eco.

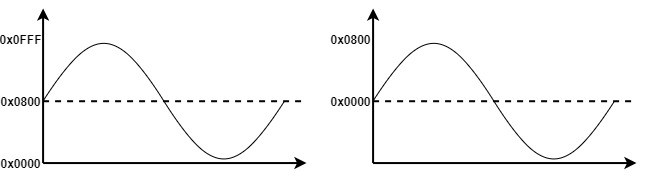


Figura - Señal sin nivelar y señal nivelada

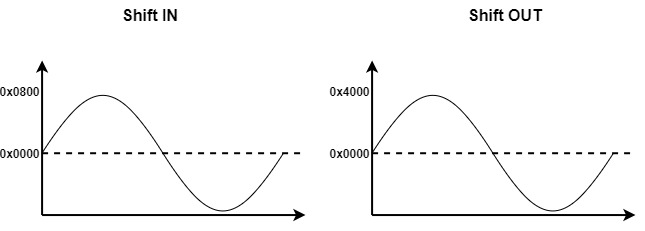


Figura - Señal sin shiftear y señal shiteada

**main\_loop\_end ()** 🡪 Suma al buffer de entrada otro buffer de nivelación que teniendo en cuenta que la salida es en 16 bits, se completa con 0x4000.

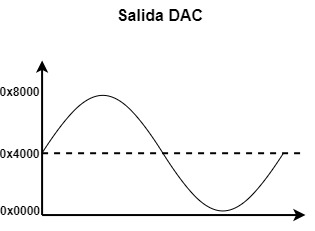


Figura - Señal de salida shifteada

* FSM del DMA (DMA\_FST\_t)

En primer término, se definieron 5 estados para representar la máquina de estados del DMA con un *enum.* A continuación, se describe la función de cada estado:

* **DMA\_INIT:** Esta listo para completar el buffer.
* **DMA\_ADC\_READY:** cambia este estado cada vez que termina de convertir el ADC. Esto indica que el ADC tiene completo el buffer y está listo para procesar.
* **DMA\_LOGIC\_BUSY:** Cambia este estado cuando comienza la lógica del programa, este estado permite detectar si el tiempo de procesamiento llegase a insuficiente. No deberían llegar muestras nuevas para que no se solapen.
* **DMA\_LOGIC\_READY:** ya terminó el procesamiento y está listo para mandar por el DAC. El tiempo de procesamiento queda confinado entre *DMA\_LOGIC\_BUSY* y *DMA\_LOGIC\_READY*.
* **DMA\_DAC\_READY:** El DMA envió los datos del buffer al DAC.

Si llega el estado **DMA\_ADC\_READY** mientras la máquina de estado se encuentra en **DMA\_LOGIC\_BUSY** (antes de cambiar a **DMA\_LOGIC\_READY**)**,** significa que el tiempo de procesamiento es demasiado grande y pueden producirse solapamiento entre las muestras procesadas y las muestras leídas del ADC.

En ese caso se debe omitir la transmisión de bloques para ADC y el I2S hasta que el tiempo de procesamiento vuelva a la normalidad.

* Manejo de Buffers del DMA

El DMA se configuró con dos buffer 8192 posiciones (entrada y salida) 🡪 [4096\*2(canales)], este tiene una interrupción cuando completa la primera mitad del buffer y otra cuando completa la segunda mitad, debido a esto se definió *DMA\_HALF\_SIZE=2048.* Es por ello que la técnica empleada para ocupar las posiciones del buffer, y al mismo tiempo poder procesarlas se las conoce como *Ping Pong.*

En primera instancia es preciso saber que se definieron 3 buffers:

* \_dma\_out: representa el Buffer de la salida DAC.
* \_dma\_in: representa el Buffer de la entrada ADC.
* \_dma\_aux: se utiliza para procesar las muestras (requiere la mitad de muestras)

Cabe aclarar que las muestras que almacena el DMA son asignadas al buffer \_dma\_in.

Por otro lado, las muestra del I2S las obtiene del \_dma\_out.

En esta máquina de estados se distinguen dos aspectos fundamentales, uno que se encarga del ADC y otro que se encarga del DAC:

* **ADC:** Se asigna la primera mitad de las posiciones del buffer \_dma\_in al buffer auxiliar \_dma\_aux cuando llega la interrupción *HAL\_ADC\_ConvHalfCpltCallback*.

Cuando llega la interrupción *HAL\_ADC\_ConvCpltCallback*, se asigna la otra mitad del buffer \_dma\_in al buffer \_dma\_aux.

Se utiliza un puntero buffer\_DMA global para poder apuntar al buffer actual desde cualquier parte del código.

Se utiliza esta técnica para mientras se almacenan las muestras que provienen del ADC, se pueda realizar el procesamiento de señales.

Como detalle es válido aclarar que en el código se utilizó una función de CMSIS para copiar un buffer a otro, arm\_shift\_q15 con un desplazamiento en 0, para optimizar el código.

* **DAC:** Se asigna las posiciones del buffer auxiliar \_dma\_aux a la primera mitad de buffer \_dma\_out cuando llega la interrupción *HAL\_I2S\_TxHalfCpltCallback*.

Cuando llega la interrupción *HAL\_I2S\_TxCpltCallback*, se asigna las posiciones del buffer auxiliar \_dma\_aux a la segunda mitad de buffer \_dma\_out cuando llega la interrupción *HAL\_I2S\_TxCpltCallback*.

En esta instancia, el puntero buffer\_DMA=NULL, para indicar que el buffer ya no es válido para procesar.

Se utiliza esta técnica para mientras se trasmite las muestras que ya fueron procesadas al I2S, se pueda copiar las muestras procesadas del siguiente buffer.

Como detalle es válido aclarar que en el código se utilizó una función de CMSIS para copiar un buffer a otro, arm\_shift\_q15 con un desplazamiento en 0, para optimizar el código.

* main\_loopback ()

Ya que el ADC convierte 2 canales en el buffer del DMA van a llegar las muestras de ambos canales intercalados por los tanto hay que hacer un procesamiento para separarlos y poder procesarlos y enviarlos al I2S. Este procesamiento es realizado en main\_loop\_start ().

* main\_eco ()

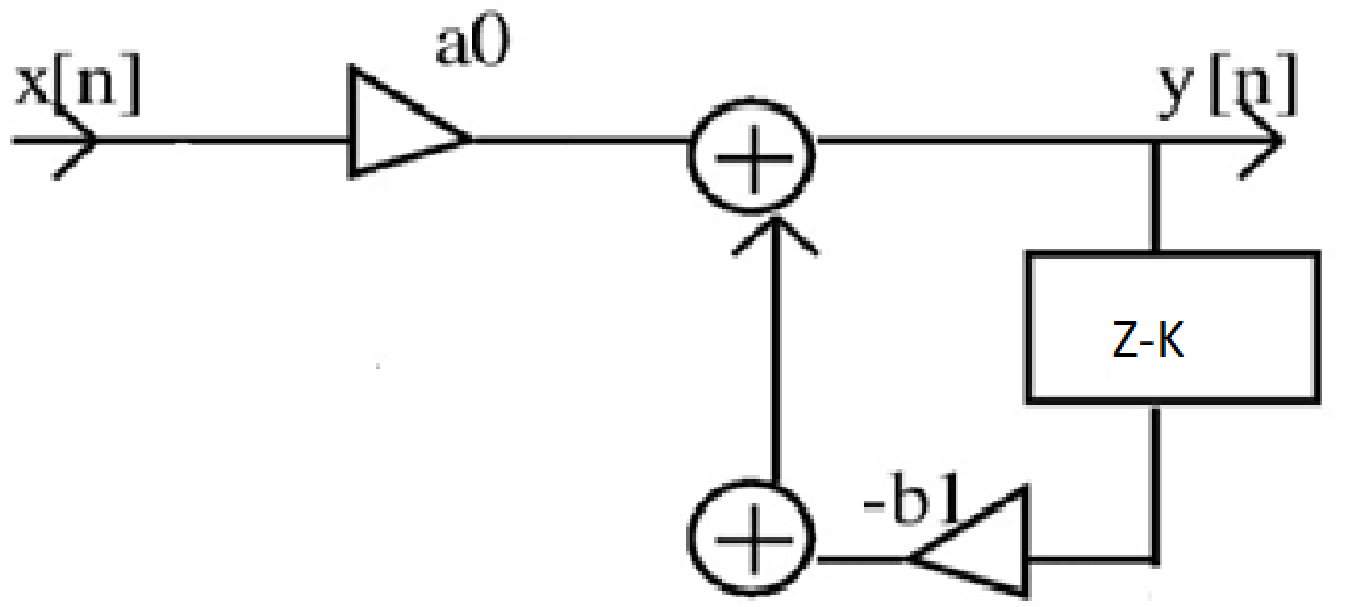


Figura 32 – Realimentación con desplazamiento (eco)

Partiendo de un vector llamado eco\_all de 8 Buffers de ADC (Cada Buffer es de 2048 muestras por canal), Se va llenando cada buffer de ese vector según van llegando. El *efecto eco* se implementa utilizando eco\_all como buffer circular.

Ejemplo:

**eco\_all:**

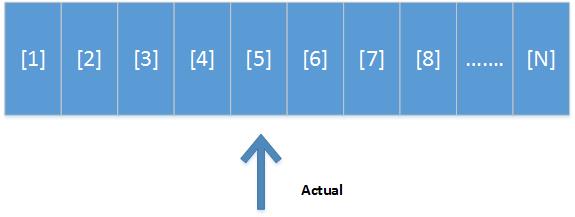


Figura – Posición Actual del vector

El puntero "actual" va a ir avanzando hacia la derecha conforme vayan llegando nuevas muestras.

Luego tengo otro puntero que apunta al mismo vector pero unas posiciones más atrasadas.

Ej:

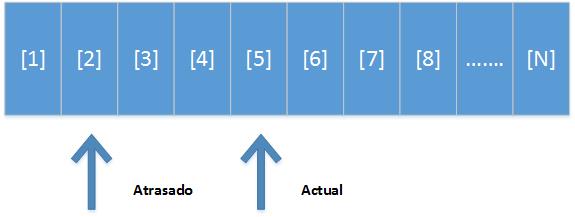


Figura – Posición actual y atrasado

Luego yo voy a ir sacando las muestras al DAC sumando la muestra apuntado por "actual" más la apuntado por “atrasado" que es lo que le da el efecto de Delay (eco). El puntero "actual" y "atrasado" se van a ir moviendo a la misma velocidad para que la frecuencia del eco no se vea afectada, y la distancia entre el puntero "actual" y "atrasado" es lo que le da el tiempo de retardo. El máximo retardo se da cuando "atrasado" está en la posición más atrasada en el vector.

**eco\_all:**

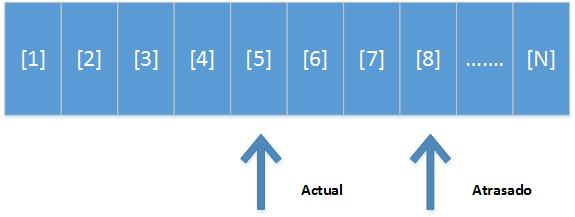


Figura – Posición actual y un retraso mas alejado

En ese caso, el retraso es de (N-1) /fs segundos. En este proyecto N=2048\*2\*8=32768 muestras, entonces el máximo retraso es de 1.02 [segundos].

* main\_fuzz ()

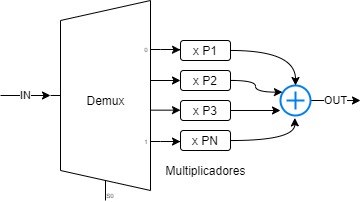


Figura 36 – Diagrama implementacion Efecto Fuzz

El efecto fuzz se lleva a cabo con una curva de salida alineal el cual se realizó codificando una función por partes. El bloque Demux esta implementado con una cadena de *if-else* que representa cada sección de la curva.

Cuando el nivel de la señal supera un de los umbrales (threshold) la señal es alterada por una pendiente.

Se utilizó codificación *4q12* para obtener mayor rango en la parte entera, y así poder utilizar pendientes más grandes, lo que me permite apreciar una mayor distorsión en la salida.

* main\_wahwah ()

Para la implementación de este efecto se acudió a la estructura de filtros Biquad. Para una mejor compresión de cómo se llevó a cabo este efecto se expondrán los 3 métodos principales del efecto wahwah: filter\_calculate (), wahwah\_init () y wahwah ().

* filter\_calculate ()

Para el cálculo de las celdas Biquad fue requisito la utilización de un software adicional especializado en cálculos matemáticos *Matlab* que es detallado en la sección 6.2.X.

A lo que código se refiere, el cálculo se realizó con variables flotantes, a diferencia del resto del código que usamos *q\_15*, para tener mayor precisión. Esto no influye en la performance, ya que estos cálculos se realizan una vez al comienzo del programa.   
Al final del cálculo se realizó la conversión a *q\_15*, para utilizarlos en los filtros que nos provee la librería de CMSIS.

* wahwah\_init ()

Se utilizó una Array de 20 filtros biquad de una etapa tipo *Peak,* con distinta frecuencia central. Que se desplaza dentro del espectro seteado.

Dichos filtros se implementaron utilizando las funciones de CMSIS *arm\_biquad\_cascade\_df1\_init\_q15*, que requiere *coeffs* (son los coeficientes calculados con la función filter\_calculate ()) y *pState* (son variables de estado que requiere la función de CMSIS).

Con lo anteriormente calculado se crean estructuras del tipo *arm\_biquad\_casd\_df1\_inst\_q15* [17], que se utilizan para establecer los filtros Peak, siendo iterados de forma secuencial, provocando la sensación de que la frecuencia central se desplaza (como si fuese un ecualizador paramétrico[[6]](#footnote-6)) a lo largo del espectro definido.

* wahwah ()

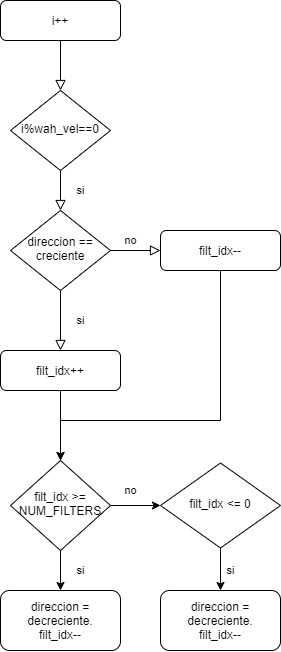


Figura 37 – Diagrama de flujo de WahWah

En este proyecto el efecto citado se ha implementado utilizando 20 filtros pasa banda, cada uno de estos filtros tiene una frecuencia central distinta, entonces conforme sucede el tiempo se recorren los filtros (de derecha a izquierda, es decir, cuando dirección=0 y viceversa). Se activa y desactivan los filtros en forma secuencial. Con el potenciómetro fijo la velocidad modificando la variable *wah\_vel* de variación de activación y desactivación de los filtros, percibiendo de esta forma el efecto deseado.

## Software adicional requerido

### Matlab

Se utilizó Matlab tratando de replicar el funcionamiento del micro de la siguiente forma

1. Se abre un archivo de audio
2. Se toma un bloque de 2048 muestras
3. Se procesa el bloque
4. Se guarda el bloque procesado en un vector de salida
5. Repite desde el punto 2 hasta el fin del archivo
6. Exporta vector de salida en otro archivo

#### Implementación para efecto Fuzz

Simulamos el efecto del Fuzz con una función por partes, conformada por rectas de distintas pendientes. Cada pendiente se calcula utilizando el valor del potenciómetro.

#### Implementación para el efecto WahWah

Este efecto se implementó usando redes Biquad de una etapa. Los coeficientes deben estar normalizados y con un cierto formato el cual se encuentra documentado en [18].

A modo de resumen este formato es 2q14, con A1 y A2 negativos.

En primer lugar, Se obtuvieron los coeficientes de los filtros, con el comando **“fdatool”** es una herramienta que nos permite diseñar filtros de forma gráfica dándole al usuario simplicidad a la hora de su síntesis. Luego, se implementó el procedimiento de conversión al formato que precisa el microcontrolador para corroborar el correcto funcionamiento, es decir, normalizado y el formato que nos indica la nota de aplicación [18].

Se hicieron algunas conversiones ya que Matlab trabaja en punto flotante y en el micro estamos usando punto fijo.

# Modo de operación

El modo de operación es muy sencillo, en este producto podemos encontrar cuatro efectos para el usuario (Loopback, Delay, Fuzz y WahWah) como mencionamos al principio. El usuario se informará del efecto que esta seteado a partir de los Leds que incluye el board de la placa con la siguiente estructura.

## Luces led indicadoras de efecto

* LoopBack -> **Verde**
* Delay (eco) -> **Naranja**
* Fuzz -> **Rojo**
* Wah-Wah -> **Azul**

Podemos navegar por estos efectos a través del botón incluido en el board de la placa (de color azul) para elegirlos de forma secuencial, con el botón de color negro que también está incluido el board, reseteamos y volvemos al LoopBack.

El desarrollo incluye un potenciómetro, para manejar la expresión de los efectos.

# Conclusiones

Entre las experiencias relevadas en el desarrollo del proyecto puedo mencionar, la intención de implementar el Audio Weaver. [15]

## Audio Weaver

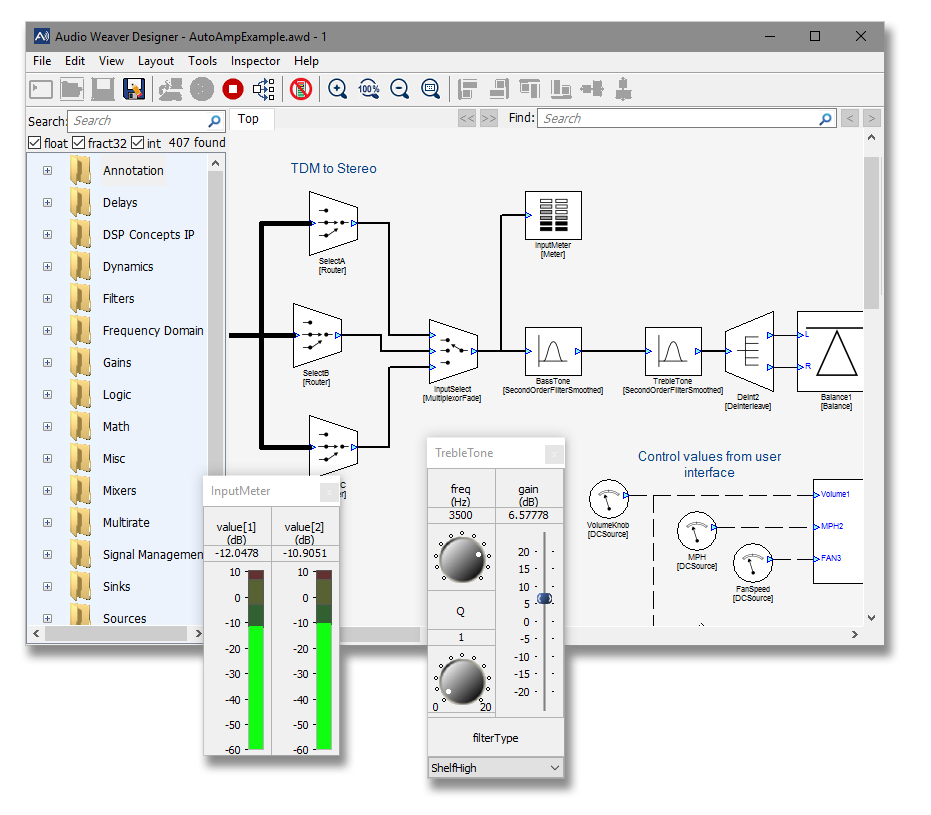


Figura 38 – Interfaz AudioWeaver

Software de desarrollo que permite de forma más rápida agregar funciones de audio con una modalidad de diagrama en bloques.

Tras haber investigado sobre el software por recomendación de la cátedra. A modo de comienzo para la familiarización de la herramienta, pude realizar a través de una conexión USB, la implementación de algunos efectos, como, por ejemplo, delay. El limitante que me encontré con esta herramienta fue la imposibilidad de cargar el código que genera el Audio Weaver a la memoria flash del microcontrolador, para poder implementar un Pedal multiefecto completamente autónomo como requiere la catedra, a causa de la dependencia de la conexión USB.

Por este motivo me vi obligado a desistir en la implementación de la herramienta sugerida. Adicionalmente, en página oficial del proveedor (que se puede visitar a través de este [LINK](https://dspconcepts.com/solutions/audio-weaver)), no explica ni provee información para la implementación que se requiere.

Por otro lado, me vi imposibilitado de implementar un Control de Master Volumen entre el DAC y el Jack de la Placa, dada la limitación física de no poder añadirlo ya que ambos elementos pertenecen al Board de la placa. Por lo tanto, implementé el Footswitch por Software para una simplificación del proyecto.

Otra necesidad en el proyecto fue la necesidad de incluir un filtro Antialias sobre exigido (Frecuencia de corte: 8[KHz]), para eliminar el contenido armónico no deseado, dado que encontré que la performance del ADC que contiene el board de la placa no es lo suficientemente exigente como para realizar el procesamiento de una señal de audio y percibirla con una fidelidad considerable.

## Relevamiento de experiencias con alinealidad característica del Fuzz

Implementar una curva con características alinéales en un microprocesador, no es fácil de realizar, no hay funciones o librerías que permitan describir una curva de estas características. A priori se intentó sintetizar esta curva utilizando raíz cuadrada y la siguiente fórmula [13]

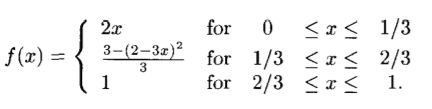


Figura – Fórmula efecto Fuzz

Pero al utilizarlo se obtuvieron las siguientes conclusiones:

1. Es difícil modificar la curva con el valor del potenciómetro.
2. Es difícil realizar las operaciones en punto fijo.
3. Las operaciones se realizan punto a punto, lo cual el pasaje a punto flotante y regresar a punto fijo, no es una opción factible.

Es por este motivo que se decidió implementar el efecto con varias rectas de distinta pendiente.

## Vicisitud con efecto WahWah

Al comienzo fue complicado implementar un filtro ya que requería de un cierto formato el cual no estaba bien documentado. Finalmente se encontró una nota de aplicación de *Texas Instruments* [18] el cual explica este formato.

Por otro lado, una de las dificultades experimentadas fue que, al trabajar con punto fijo, no se obtiene un rango dinámico muy grande, lo que limita de forma considerable el efecto percibido.

Finalmente habiendo sobrepasado los anteriores problemas, se llegó a que al aplicar el filtro *Peak* a la salida se escucha un tono igual a la frecuencia central del filtro. Esto hace creer que el filtro tiene un Q muy elevado y/o una ganancia muy elevada. Sin embargo, modificando estos parámetros no se obtuvieron mejoras.

El efecto final obtenido, fue un tono que varía en frecuencia al ritmo del WahWah.

## Relevamiento de experiencias por necesidad de conversión de formatos

En primer lugar, se propuso utilizar el formato de punto flotante ya que este tiene la ventaja de utilizar números racionales y un alto rango dinámico. A raíz de que este micro *STM32F411* posee FPU, se supuso que no iba a influir en el tiempo de procesamiento. Sin embargo, al realizar las primeras pruebas se concluyó que el tiempo de procesamiento no era suficiente, ni siquiera para implementar un simple Eco. Por lo que me vi obligado a modificar el formato a uno menos “pesado”, este formato fue el q31.

Por otro lado, conforme se avanzó con el código del proyecto, se sorteó la dificultad de los escases en el espacio disponible en memoria (sobre todo con el efecto eco, ya que para su implementación hay que utilizar varios buffers). Debido a esta dificultad nuevamente no hubo alternativa y se tuvo que cambiar de formato, a uno de menor carga, el formato seleccionado fue el q15. Debido a estos cambios se tuvo que cambiar el código de raíz teniendo que reescribir prácticamente todo el código desde cero readecuándolo a este formato (Formato de los buffers, configuración del ADC, configuración del I2S y máquina de estados del DMA).

Finalmente, el formato definitivo fue q15.

# Proyecto finalizado



Figura 40 – Imagen superior del prototipo



Figura 41 – imagen frontal del prototipo

# Referencias

[1]. B. Goldstein, *Blackweel handbook of sensation and perception*.

Oxford:

Blackweel Publishing, 2005.

[2]. ESPASA, *Diccionario de la lengua española*. Pozuelo de Alarcón:

Espasa-­Calpe, 2005.

[3]. PFC, ESTUDIO DE EFECTOS DE AUDIO PARA GUITARRA, E

IMPLANTACIÓN MEDIANTE DSP. Ángel Pérez Rodríguez. Feb 2006.

[4]. PFC, Multi-­Effect Processor for acoustic guitar. Eduardo Fonseca

Montero. Dic 2007.

[5]. PFC, Development of Guitar Music FX to Aid DSP Teaching. Jody

Monahan. Marzo 2010.

[6]. Smith, Julius O. "Chorus Effect." *Center for Computer Research in*

*Music and Acoustics,* CCRMA. 2009. Web. 18 Apr. 2010.

<https://ccrma-­ww.stanford.edu/~jos/waveguide/Chorus_Effect.html>

[7]. Bode, H. "History of Electronic Sound Modification." *Journal of the*

*Audio Engineering Society*32 (1984): 730-­39. Web. 2.0. Feb. 2010.

[8]. Wikipedia, the Free Encyclopedia 12 June 2008. Web. 18 Apr. 2010.

<http://en.wikipedia.org/wiki/>

[9]. Harmony-­ Central Effects Explained. Información sobre efectos

digitales.

[10]. PFC de Iván Palomares Alguacil el cual nos ha servido de punto de inicio de este TFG. IMPLEMENTACION SOFTWARE DE UNA HERRAMIENTA PARA LA GRABACIÓN/EDICIÓN DE MAQUETAS MUSICALES AMATEUR:

[http://avalos.ujaen.es/search~S2\*spi/?searchtype=X&searcharg=Iván+Palomares+Alguacil&searchscope=2&sortdropdown=-](http://avalos.ujaen.es/search~S2*spi/?searchtype=X&searcharg=Iván+Palomares+Alguacil&searchscope=2&sortdropdown=-)­&SORT=DZ&extended=0&SUBMIT=Buscar&searchlimits=&searchorigarg=X\*

[11]. TFG de El-­Alga, Kaoutar -­ Programación de herramientas de

afinación de instrumentos para dispositivos móviles:

http://tauja.ujaen.es/browse?type=author&value=El-­Alga,+Kaoutar

[12]. Acústica Musical: Conceptos básicos sobre el Sonido:

https://www.google.es/ -­

q=https%3A%2F%2Fwww.lpi.tel.uva.es%2F~nacho%2Fdocencia&rct=j

[13]. Udo Zolzer - Digital Audio Effects

Primera edición - Editorial: John Wiley& Sons.

[14] Hoja de dato: CS43L22–Low Power, Stereo DAC w/Headphone & Speak Amps– Cirrus Logic.

[15] Link: DSP Concepts–Audio Weaver

<https://dspconcepts.com/solutions/audio-weaver>

[16] RM0383 Reference manual

[17] Librería CMSIS, estructura de filtros bicuad.  
<https://arm-software.github.io/CMSIS_5/DSP/html/structarm__biquad__casd__df1__inst__q15.html>

[18] Configure the Coefficients for Digital Biquad Filters – Texas Instruments

1. Comb Filter: se produce al sumarle a la señal original una versión retrasada en el tiempo de sí misma, causando así [interferencia](https://es.wikipedia.org/wiki/Interferencia) [constructiva](https://es.wikipedia.org/wiki/Interferencia_constructiva) y [destructiva](https://es.wikipedia.org/wiki/Interferencia_destructiva). [↑](#footnote-ref-1)
2. Overdrive: es un tipo de efecto empleado para guitarra eléctrica (distorsión) [↑](#footnote-ref-2)
3. Boss: Marca reconocida en el mercado de pedal de efectos de guitarra. [↑](#footnote-ref-3)
4. **Twang:** es una onomatopeya utilizada originalmente para describir el sonido de una cuerda de arco vibrante después de soltar la flecha [↑](#footnote-ref-4)
5. **Dunlop:** Marca reconocida en el mercado de pedal de efectos de guitarra. [↑](#footnote-ref-5)
6. Ecualizador paramétrico: permite el control individual de parámetros: ej. su frecuencia central. [↑](#footnote-ref-6)