

ádiz1

# Stringless Guitar - Proyecto Final

Cuervo B., Nicolas y Horta C. Juan, D  
 {ncuervob,jdhortaca} @unal.edu.co

**Abstract**—We’ve been working along the semester developing the course project: the stringless guitar. As we said in the first review, we’ve already reached the goal of generating sounds via pwm. From there, the effort was directed to generate the different notes of the guitar, and also, to emulate the guitar’s sound. To accomplish this, FIR filters were very useful, as functions for the surrounding wave (which give us the similarity of the guitar sound). The inputs of the system were sent as interrupts via IRQ-GPIO using software developing; meanwhile, the synthesis of the sound is made as a hardware task to make it faster. An additional peripheral speaker is connected to a jack plug to hear the guitar, making it a digital “electro-acoustic look alike” device.

**Index Terms**—FPGA, guitar, PWM, sounds

## I. PROPONENTES

- Nicolás Cuervo Benavides. 261539
- Juan Daniel Horta. 260996

## II. ÁREA

Electrónica Digital. Programación y desarrollo hardware - software.

## III. INTRODUCCIÓN

En el presente escrito se presenta el desarrollo que ha tenido la realización del proyecto final del curso: la *Stringless Guitar*. Se ha trabajado a lo largo del semestre en la generación de sonidos a través del procesador; ésto se ha logrado mediante una modulación por ancho de pulso (PWM por sus siglas en inglés). Para ésto se desarrollaron osciladores de diferentes frecuencias (para cada nota) con forma de diente de sierra (dado a que éste tipo de onda es rico en armónicos). Las entradas del sistema vienen de un tratamiento de señal efectuado a rayos de luz interrumpidos, simulando así las cuerdas de la guitarra; digitalmente se envían a través del GPIO del procesador, y se toman como interrupciones. La salida del sistema es una señal de audio, que es posible escuchar a través de unas bocinas externas. El desarrollo del proyecto de curso

permitió conocer las funcionalidades del procesador LM32, y cómo debe ser manipulado al entender su arquitectura.

## IV. OBJETIVOS

**General:** Presentar el dispositivo *Stringless guitar* en funcionamiento.

**Específicos:**

- Generar señales que puedan representar un sonido real mediante programación
- Filtrar la señal generada con el fin de obtener un sonido semejante al producido en una guitarra
- Implementar la estructura de interfaz de usuario

## V. CRONOGRAMA

El cronograma propuesto y las modificaciones realizadas a lo largo del desarrollo del proyecto se encuentran en el anexo 1.

## VI. COMPORTAMIENTO

La idea consiste en emular una guitarra, sólo que prescindiendo de sus cuerdas. El dispositivo en vez de cuerdas cuenta con unos pares emisores-receptores de luz, de manera que una interrupción en el trayecto del rayo lumínico envía una señal al procesador, indicándole que debe generar el sonido. La forma física del dispositivo no difiere a la de una guitarra común.

De cada par emisor-receptor el procesador recibe una señal diferente, generando así mismo un sonido a diferente frecuencia. Además se ha dispuesto de 4 pulsadores que emulan los trastes que al ser pulsados cambian la frecuencia del sonido generando, percibiéndose una señal más aguda. El flujo de datos que describe el sistema se aprecia en la fig. 1.

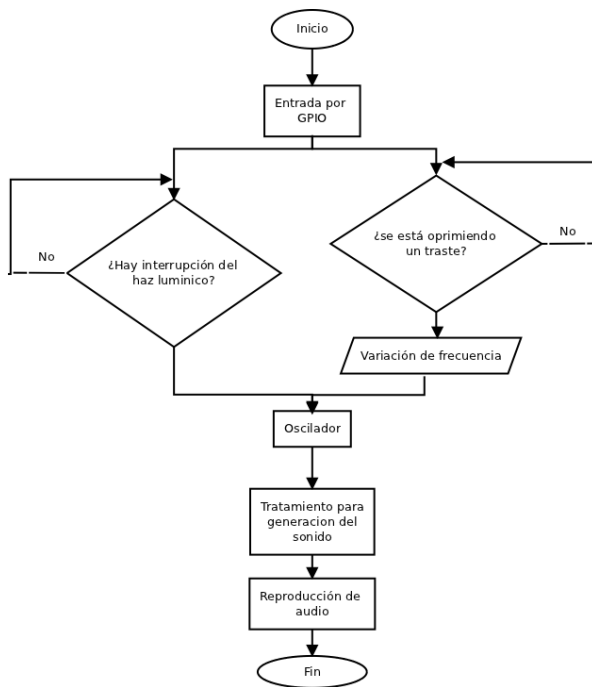


Fig. 1: Diagrama de flujo de señal

## VII. ESPECIFICACIONES

**Temporales:** El dispositivo genera sonidos en tiempo real, como respuesta a una interrupción física entre el emisor de luz y el receptor. Además, también modifica la frecuencia del sonido emitido al pulsar los trastes; el delay que le toma al procesador efectuar esas operaciones no es perceptible al oído humano.

**Físicas:** El dispositivo se ha instalado en un prototipo de madera con forma de guitarra cómo se pudo observar en la fig. 1, con unas dimensiones aproximadas de 56cm x 21cm x 9cm.

**Funcionales:** Las emisiones de luz se harán mediante lasers indicadores, mientras que fotoceldas tendrán la función de receptores. El estado estable del dispositivo consiste en un flujo ininterrumpido entre los emisores y los receptores. Una interrupción efectuada sobre el haz de luz envía una señal al procesador y genera el sonido. Además, a partir de pulsadores que emulan los trastes se obtienen variaciones en los osciladores, generando sonidos más agudos al ser pulsados.

**Eléctricas:** El dispositivo se alimentará con una batería de 3.7v.

## VIII. ARQUITECTURA DEL SISTEMA

Las tareas del dispositivo se dividen en la manera en que son instanciadas, ya sea hardware o software. Éstas son:

### Tareas hardware

El criterio principal para determinar las tareas que se deben efectuar por medio de hardware es básicamente la velocidad. Dado a que se está generando audio en tiempo real, es requerido que éste se obtenga sin demora al momento en que efectúa la solicitud. La instanciación hardware permite generar esta interfaz necesaria para que los delays de procesamiento no sean notorios para el usuario. Las tareas efectuadas en este apartado son:

- Recepción de las señales externas desde los pulsadores
- Generación del oscilador y su tratamiento para obtener los sonidos
- Reproducción del sonido generado.

### Tareas software

Las tareas efectuadas en software son meramente de control. La programación software fue vital para manejar los procesos de interrupciones a partir de las entradas externas, así como el control del proceso de generación de audio en general.

- Control general de periféricos y generación de audio

## IX. ESTRUCTURA

Básicamente la operación del dispositivo puede dividirse en etapas, caracterizadas por la función que ejercen dentro del proceso. Estas etapas se pueden apreciar en la fig. 2.

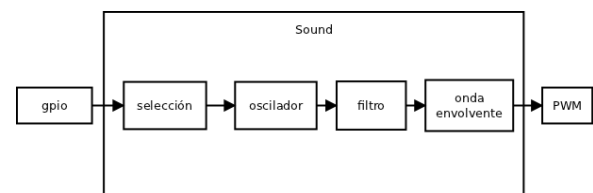


Fig. 2: Diagrama de bloques del dispositivo

Siguiendo el flujo de datos de izquierda a derecha en la fig. 2, el primer bloque corresponde a la entrada de datos mediante el periférico de *entradas y salidas de propósito general* (GPIO, por sus siglas en inglés). En éste periférico se toman las entradas externas que provienen de las interrupciones en el haz de luz o de los pulsadores. Estas entradas van al módulo de *Sound* donde se sintetiza toda la parte de generación de audio. Cómo se puede apreciar éste segmento se compone del bloque de la selección, el oscilador, el filtrado y la onda envolvente.

### Selección

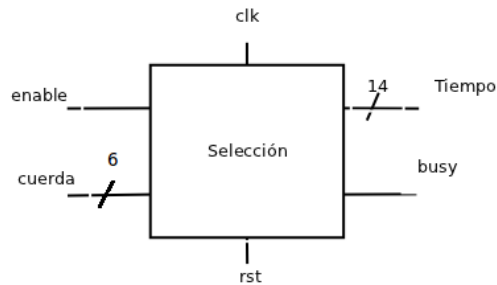


Fig. 3: Entrada/salida del módulo de selección

Básicamente, cómo su nombre lo indica, este módulo tiene la función de seleccionar las cuerdas de manera que el oscilador tenga el parámetro correspondiente a partir de las entradas recibidas del GPIO.

### Oscilador

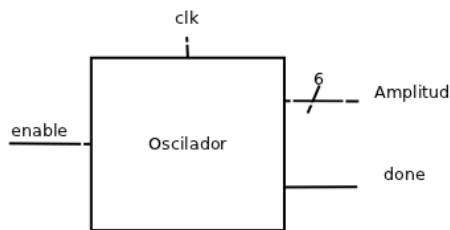


Fig. 4: Entrada/salida del módulo del oscilador

En este módulo se crea por medio de instancia hardware una seal de la forma diente de sierra. Ésto se hace mediante un contador. El módulo internamente tiene unos parámetros que determinan su frecuencia; sin embargo estos son elegidos a partir de la seal de enable proveniente del proceso de selección.

### Filtros

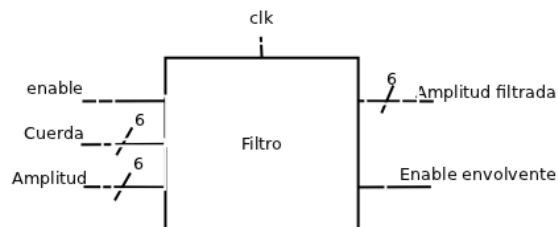


Fig. 5: Entrada/salida del módulo del filtro

La función de este módulo es limitar los valores de frecuencia de la seal obtenida en el oscilador, de manera que los cambios de amplitud no sean abruptos. Para esta sección se ha decidido utilizar filtros FIR, dado a que requieren menos recursos

para operar que su contraparte IIR. Los filtros FIR operan como se aprecia en la fig. 6.

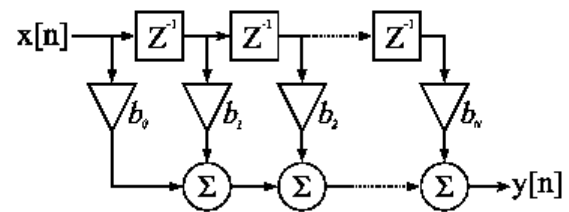


Fig. 6: Diagrama de un filtro FIR [1]

Lo interesante de estos filtros es que no tienen memoria; es decir, su comportamiento depende únicamente de la entrada actual y no de valores pasados, de manera que su respuesta siempre es finita.

Físicamente estos filtros son los que permiten la semejanza a el sonido de la guitarra. Dado a que es un filtro de alto orden (10), así mismo se requieren coeficientes para su funcionamiento. Estos valores de coeficientes al ser variados arrojan un sonido más suave o más fuerte.

### Onda envolvente

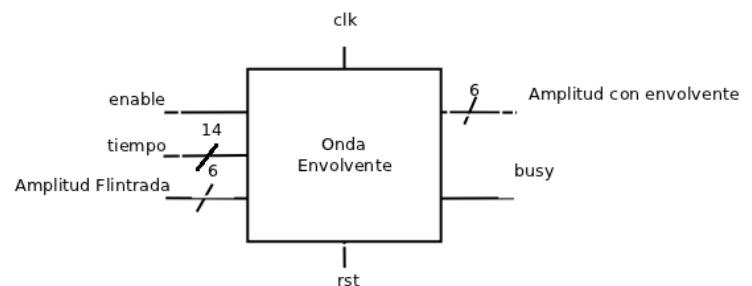


Fig. 7: Entrada/salida del módulo de la onda envolvente

Este módulo se encarga de emular la duración e intensidad del sonido de una guitarra. Es fácil identificar que, al tocar una cuerda, la amplitud de la onda de audio crece rápidamente hasta su valor máximo, mantiene una amplitud de la nota y luego cae en el tiempo. Éste proceso se ha estudiado bastante en el ámbito de la generación de sonidos, y se han identificado las etapas que tiene el sonido al ser generado en: attack, decay, sustain, release.

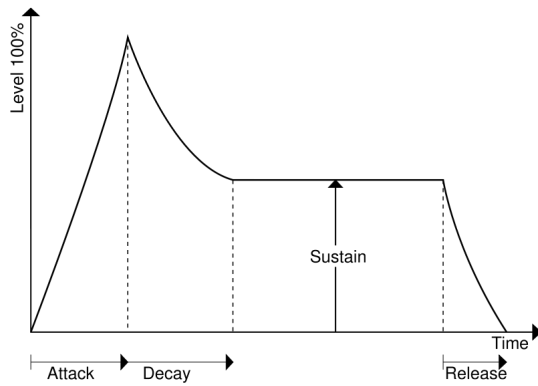


Fig. 8: Forma de onda envolvente, con sus respectivas etapas

Para cada etapa presente en la fig. 8 se desarrolló un submódulo, variando la amplitud de la señal filtrada y de esta manera generando semejanza a el decaimiento temporal de una nota al ser tocada en una guitarra.

## PWM

PWM [3] es la sigla en inglés de modulación por anchura de pulsos (pulse-width modulation), y es una técnica en la que se modifica, cómo su nombre lo indica, el ancho del pulso de una señal periódica; este ancho se denomina ciclo de trabajo, tomándose su medida en relación con el periodo de la señal.

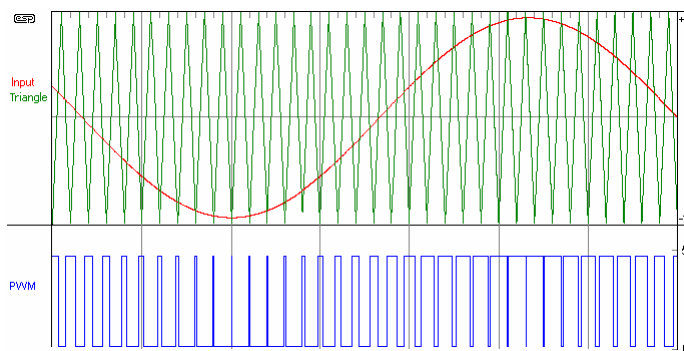


Fig. 9: PWM

Con éste módulo lo que se busca es generar seales a diferentes frecuencias, siendo las frecuencias elegidas las que caracterizan las notas determinadas. Luego, a la salida de la fpga mediante un filtro suavizar la señal de manera que sea audible.

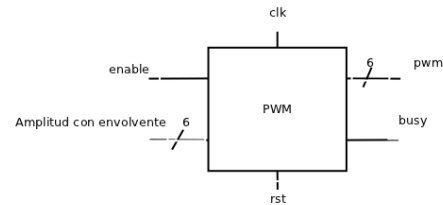


Fig. 10: Módulo de la PWM

Cómo parte final de la generación de audio, se utilizó la modulación por ancho de pulso para efectuar una conversión analógica/digital sin necesidad de un ADC extra. En éste módulo, el ancho del ciclo de trabajo de cada pulso está relacionado con la amplitud de la señal de entrada; es decir, la amplitud de la señal luego de pasar por la onda envolvente. De esta manera se obtiene a la salida del sistema una variación perceptible al conectar una bocina; ésta señal de pwm es la salida principal del dispositivo. La conexión de la salida del dispositivo es directa a unas bocinas externas, cómo se especificará posteriormente en este mismo escrito.

## X. COMUNICACIÓN WISHBONE

La intercomunicación entre módulos y tareas hardware/software se realiza utilizando el protocolo wishbone. Para lograr la comunicación es preciso determinar jerarquías entre bloques del procesador. De esta manera, se dispone de un maestro que es la unidad de procesamiento CPU, y de dos esclavos: GPIO y el módulo Sound. El diagrama de bloques del proceso se aprecia en la fig.

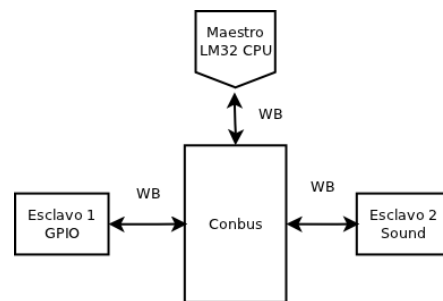


Fig. 11: Intercomunicación utilizando protocolo wishbone

## XI. MONTAJE

Para el montaje físico del dispositivo se desarrolló un circuito al cual se conectan los sensores lumínicos y se envían a la fpga. El esquemático del diseño se aprecia en el Anexo 2. Las señales obtenidas de los sensores son analógicas, de manera

que fue necesario digitalizarlas. Con este fin, se valió del uso de una Schmitt trigger.

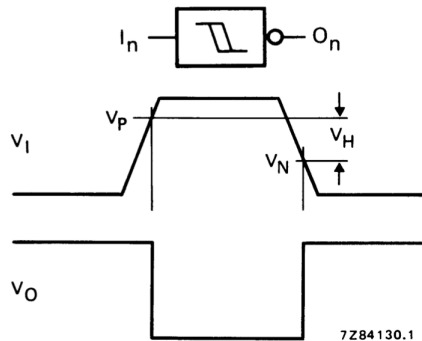


Fig. 12: Formas de onda de entrada y salida de una compuerta Schmitt trigger

El funcionamiento de esta compuerta es bastante sencillo: al sobrepasarse un nivel de tensión de "gatillo", la compuerta arroja a la salida un nivel alto. De manera análoga, al encontrarse a una tensión más baja, la compuerta arroja una salida de nivel bajo. Básicamente digitaliza la señal a 1 bit de información. De ésta manera, esa señal digital funciona como interruptor que va al GPIO.

Luego del tratamiento de señal, la salida del módulo pwm es adquirida a través de un conector Jack de audio para ser conectada a unas bocinas externas; así se evita el tener que adicionar una etapa de amplificación y filtrado de la señal.

El diseño de la pcb se realizó utilizando *Proteus*. El resultado se aprecia en la fig. 13.

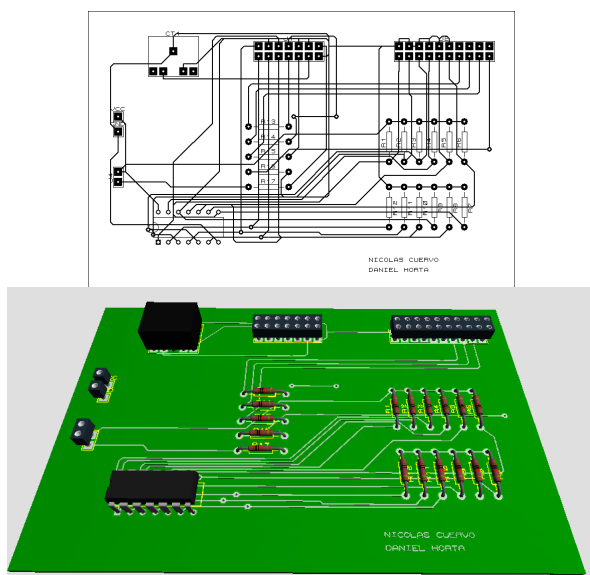


Fig. 13: Circuito Impreso

Cómo producto final se instaló el dispositivo desarrollado en una guitarra mediana.

## XII. ANÁLISIS DE COSTOS

El análisis de costos detallado se encuentra en el anexo 3.

## XIII. CONCLUSIONES

- El producto obtenido representa un desarrollo que se ha venido realizando a lo largo del curso, realizando un manejo de la arquitectura de los procesadores, ms específicamente del LM32. Ms que la generación de audio, el proyecto tiene un enfoque de innovación, donde prima la manera original de presentar dispositivos originales utilizando la tecnología existente.
- El desarrollo de un elemento de ingeniería no se limita a lo obtenido en una sesión de programación. Un dispositivo que sea aceptado en la industria debe tener tanto un alto nivel de ingeniería de desarrollo como una interfaz de usuario llamativa y sencilla. Para este proyecto se imprimió un gran esfuerzo en lograr un elemento que sea llamativo al público en general, elemento que representa una gran fracción del segmento objetivo en la industria profesional.

## XIV. PROYECTOS FUTUROS

El producto final es un elemento bastante llamativo, pero podrían ser aplicadas nuevas tecnologías para optimizarlo. Al introducir técnicas de síntesis de audio más avanzadas se podría emular mejor el sonido de la guitarra. Además, el uso de sensores más sofisticados tanto para las cuerdas como para los trastes podrían dar vía a alternativas interesantes de desarrollo musical digital.

## REFERENCES

- [1] R. Cádiz, *Filtros FIR*, [En Línea] Disponible en: "http://www.rodrigocadiz.com/imc/html/Filtros\_FIR.html" Fecha de consulta: 13-06-2012
- [2] H. Taigen, *Musical Blocks*, [En Línea] Disponible en: "http://people.ece.cornell.edu/land/courses/ece4760/FinalProjects/s2009/jvt6\_th389/jvt6\_th389/finalproject.html" Fecha de consulta: 6-06-2012
- [3] S. Vanheesbeke, *Swapping bits improves performance of FPGA-PWM counter*, EDN Europe: Electronics DEsign, Strategy, News [En Línea] Disponible en: "Swapping bits improves performance of FPGA-PWM counter" Fecha de consulta: 07-05-2012