

Procesamiento de señales para aplicaciones y efectos de audio en tiempo real con FPGA

Félix Garro Martínez; Diego Costa; Carlos Sosa Páez (*)

Laboratorio de Electrónica, Investigación y Servicios
 Facultad de Ciencias Físico, Matemáticas y Naturales / Universidad Nacional de San Luis
 San Luis, Argentina
 e-mail: garrofelix@gmail.com; {dec, sosapaez}@unsl.edu.ar

Resumen— Se diseñó un sistema de procesamiento de señales en tiempo real para producción musical con un afinador y efectos de de trémolo y digital delay. Se implementó en FPGA describiendo el circuito en VHDL bajo estándar WISHBONE para integrarlo a un trabajo de referencia que contiene otros efectos de supresión de ruido, compresor, chorus y flanger en cascada, con una interfaz HM para el ajuste y visualización de los parámetros.

Palabras clave: Efectos de audio, afinador, trémolo, digital delay, eco, reverberancia, tiempo real, FPGA, VHDL, WISHBONE.

I. INTRODUCCIÓN.

En el presente trabajo se diseñó un procesador de señales que implementa una aplicación consistente en un afinador para guitarras, y efectos de trémolo y *digital delay* [1]. Se utilizó como trabajo de referencia [2] un rack de efectos de compresión, *chorus*, *flanger* y supresión de ruido de línea, el cual contenía una interfaz de control y visualización para selección de efectos y ajuste de parámetros además del sistema de conversión con circuitería de acondicionamiento y filtrado. El trabajo se realizó bajo las especificaciones WISHBONE [3] en 16 bits. Se hizo un diseño modular [4] lo que permitió probarla en dos FPGA de Actel [5] con leves modificaciones para adecuarlas a los recursos en cada caso.

II. ELEMENTOS EMPLEADOS.

A. Herramientas de diseño.

Para depuración del diseño se utilizó el Libero 9.1 con la herramienta Smart Gen para generar *cores*, Synplify DSP AE para la síntesis, Designer para posicionamiento-ruteo, Flash Pro para programación y ModelSim para simulación pre-síntesis, post-síntesis y post-layout [6]. Los circuitos se realizaron en VHDL describiendo directamente el código, con excepción del bloque de reloj que se hizo con Smart Gen mediante librerías de diseños parametrizados.

B. Dispositivos empleados.

Una de las placas usadas fue la RVI Prototype Board del ICTP (**) que posee un FPGA ProASIC3E A3PE1500 de Actel, pulsadores, LEDs y *displays* de siete segmentos. Dicha placa trabaja con la expansión LP Data Conversion Daughter Board (**) que está provista de un convertor A/D AD9201 de 10 bits y un convertor D/A LTC1654 serie de 14 bits.

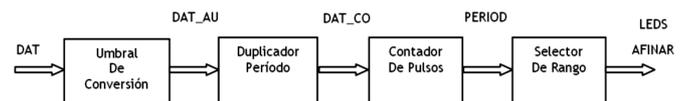


Figura 1. Módulos del afinador.

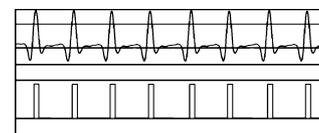


Figura 2. Señal de entrada DAT y señal umbralizada DAT_AU.

La otra placa empleada fue la Fusion Embedded Development de la misma firma. Posee un FPGA Fusion M1AFS1500 con encapsulado 484FBGA, y contiene LEDs y pulsadores. La placa porta un convertor A/D y potenciómetro, pero no tiene convertor D/A por lo que se le acopló la misma LP Data Conversion Daughter Board. Además, se adjunta un programador desconectable con puerto USB para bajar el código de la PC al FPGA.

El reloj se definió en 25 y 20 MHz para cada placa. De allí se derivó la frecuencia de muestreo de 44169 Hz.

III. DISEÑO.

A. Afinador.

Se implementó un afinador para guitarra eléctrica con detección automática de la cuerda que ha sido pulsada. En la Figura 1 se muestra el diagrama en bloques.

En la versión realizada para el FPGA ProASIC3E A3PE 1500, se utilizó un *display* de siete segmentos para mostrar la cuerda digitada en nomenclatura americana y tres LEDs para indicar si la frecuencia es mayor, menor o es exacta. En la versión para el FPGA Fusion sólo se utilizaron los LEDs.

Para el diseño del sistema de afinación se probaron dos métodos: La detección de cruce por cero y la detección de picos. La forma de onda típica del sonido de la guitarra posee, en cada período, múltiples cruces por cero, y además de un máximo absoluto, tiene varios máximos locales. El primer método produce errores por los múltiples cruces por cero. Por eso fue elegido el segundo método aunque requiere diferenciar el pico mayor de los picos menores que deben descartarse.

(*) Pertenecientes al Proyecto "Instrumentación Virtual Reconfigurable" financiado por Ciencia y Técnica de la UNSL.

(**) Esta placa ha sido provista por el ICTP (International Center For Theoretical Physics) y fue desarrollada por A. Cicuttin, M. L. Crespo y A. Shapiro.

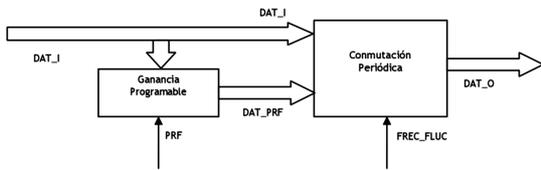


Figura 3. Módulos del trémolo.

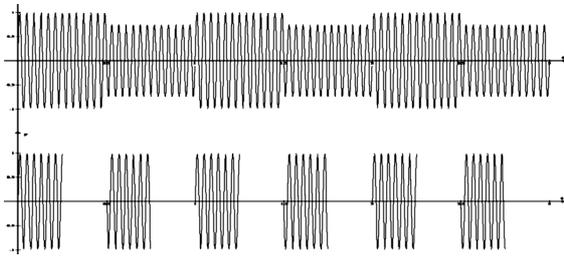


Figura 4. Señal de salida DAT_O para fluctuación = 1 y profundidad = 2; y fluctuación = 3 y profundidad = 7.

Para esa detección de picos se eligió un mecanismo sencillo y estable que consiste crear una señal digital, mostrada en la Figura 2, de igual período que la entrada, por umbralización mediante un comparador. A dicha señal se le duplica el período y corrige el ciclo de trabajo llevándolo al 50%. El conteo de pulsos se realiza durante medio ciclo. Este método fue útil para depurar el sistema y evitar disparos erróneos del contador. Un detector de rango compara la cuenta de pulsos con un mínimo y un máximo según los intervalos de frecuencia correspondientes mostrados en la Tabla 1. Cada intervalo se definió centrado en la frecuencia correspondiente a la nota estándar de cada cuerda. En dicha tabla se detalla el valor enviado al control de pantalla, para indicar la cuerda, en la versión de la RVI Prototype Board que posee ese recurso.

TABLA 1. INTERVALOS PARA LA DETECCIÓN DE LA CUERDA DIGITADA Y SU CORRESPONDIENTE SEÑALIZACIÓN.

Cuerda	Nota	Octava	Intervalo (Hz)	Señalización
6°	Mi (E)	2	[70; 96)	E (101)
5°	La (A)	2	[96; 128)	A (100)
4°	Re (D)	3	[128; 171)	D (011)
3°	Sol (G)	3	[171; 221)	G (010)
2°	Si (B)	3	[221; 288)	B (001)
1°	Mi (E)	4	[288; 365]	E (000)

Una vez detectada la cuerda digitada, el sistema realiza una segunda comparación, más precisa, acotando el intervalo de frecuencias a un entorno cercano de cada valor estándar de afinación, según se detalla en la Tabla 2. Dichos intervalos se definieron para una detección con un error menor al 1%. Esto permite distinguir el estado de la afinación de la cuerda, señalando si está dentro del entorno válido, si está por encima o por debajo. Esas situaciones se codifican con '01', '10' y '00' respectivamente en la señal que comanda los LEDs, la cual se refresca cada cierto tiempo para dar estabilidad en la visualización.

TABLA 2. INTERVALOS ACOTADOS PARA DISTINGUIR ESTADO DE AFINACIÓN.

Cuerda	Nota	Octava	Frecuencia (Hz)	Intervalo (Hz)
6°	Mi (E)	2	82,41	[82,58; 82,24]
5°	La (A)	2	110,00	[110,30; 109,70]
4°	Re (D)	3	146,83	[147,37; 146,29]
3°	Sol (G)	3	196,00	[196,58; 195,43]
2°	Si (B)	3	246,94	[247,86; 246,03]
1°	Mi (E)	4	329,63	[330,72; 328,55]

B. Trémolo.

El efecto de trémolo es la modulación en amplitud de la señal de audio con una señal moduladora de forma de onda cuadrada. El diagrama de bloques se muestra en la Figura 3.

En la Tabla 3, se exponen los parámetros ajustables:

- La profundidad PRF define la diferencia relativa entre la amplitud de la señal atenuada con respecto a la original.
- La fluctuación FREC_FLUC define la frecuencia con la que conmuta la amplitud modulada entre los valores de la señal original y el seleccionado con profundidad.

Los valores nominales de los parámetros se fijaron empíricamente a fin de que cada cambio en FREC_FLUC o PRF tenga un efecto distinguible. El bloque Ganancia Programable asigna a DAT_PRF la entrada DAT_I afectada por una ganancia. Esta depende del parámetro PRF según el algoritmo de la Tabla 4. Se observa que la representación es de 16 bits para respetar el estándar WISHBONE. El bloque de Comutación Periódica Programable alterna las señales DAT_I y DAT_PRF en su salida DAT_O a una frecuencia dada por el parámetro FREC_FLUC. La Figura 4 ejemplifica el efecto.

TABLA 3. VALORES NOMINALES PARA LOS PARÁMETROS PRF Y FREC_FLUC.

PRF	1	2	3	4	5	6	7
Ganancia	0,90625	0,75	0,625	0,5	0,375	0,125	0
FREC_FLUC	1	2	3	4	5	6	7
Período (ms)	1	0,75	0,5	0,25	0,16	0,12	0,1

TABLA 4. ALGORITMO DE PROFUNDIDAD DEL TRÉMOLO.

PRF	Señal de salida
1	$(DAT_I/2) + (DAT_I/4) + (DAT_I/8) + (DAT_I/32) + 3072$
2	$(DAT_I/2) + (DAT_I/4) + 8192$
3	$(DAT_I/2) + (DAT_I/8) + 12288$
4	$(DAT_I/2) + 16384$
5	$(DAT_I/4) + (DAT_I/8) + 20480$
6	$(DAT_I/8) + 28672$
7	32768

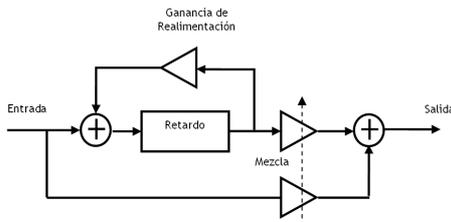


Figura 5. Diagrama de bloques del digital delay.

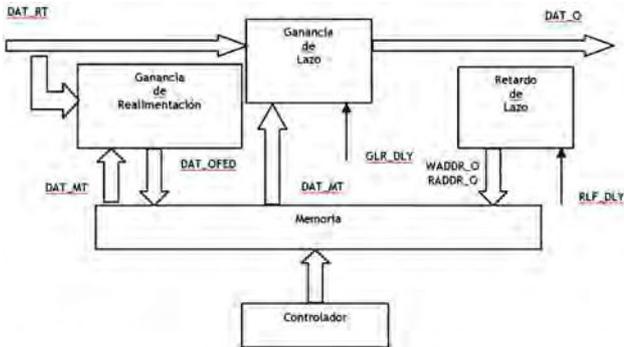


Figura 6. Módulos del digital delay.

C. Digital delay (eco y reberverancia)

Se implementó un Delay Base Realimentado, modificado como se expone en la Figura 5, con realimentación fija y control de mezcla que suma ponderadamente ambas señales a modo de balance. El efecto tiene dos parámetros ajustables:

- El retardo de lazo RLF_DLY controla el tiempo de la realimentación para los valores dados en la Tabla 5.
- La ganancia de lazo GLR_DLY controla la proporción de amplitud entre la señal directa y la retardada.

El diagrama de los módulos se ve en Figura 6. El bloque Ganancia de Lazo pondera los valores de la señal directa DAT_RT y la retardada DAT_MT. Su salida es DAT_O. El bloque Ganancia de Realimentación pondera los valores de DAT_RT y DAT_MT según el algoritmo de la Tabla 6 y escribe su valor en memoria. La Ganancia de Realimentación se fijó en un valor igual a $DAT_MT/2 + DAT_RT/2$. El bloque Retardo de Lazo calcula la dirección de memoria en la que se lee el valor de entrada DAT_MT según el parámetro RLF_DLY. El Controlador de Memoria gestiona el funcionamiento de lectura y escritura. La memoria se configuró en dos puertos, y dividida en dos partes, una destinada a escritura y otra a lectura de datos. El tamaño se definió en 16384 palabras de 16 bits. Para la frecuencia de muestreo de 44169 Hz se logran 371 ms de retardo máximo.

D. Interfaz humana

El usuario interactúa con el sistema diseñado a través de la interfaz que permite seleccionar cada uno de los parámetros de los distintos efectos y asignarle el valor deseado. Se usó el esquema de navegación del proyecto de referencia, con las modificaciones del caso. La máquina de estado maestro con un estado por efecto y función, llama a máquinas de estado esclavas, donde cada estado representa un parámetro.

TABLA 5. VALORES NOMINALES PARA EL PARÁMETRO RLF_DLY.

RLF_DLY	1	2	3	4	5	6	7
Retardo (ms)	68	113	181	226	271	317	362

TABLA 6. ALGORITMO DE MEZCLA DEL DELAY.

GLR_DLY	Ganancia
1	$DAT_MT/128 + (DAT_RT/2 + DAT_RT/4)$
2	$DAT_MT/64 + (DAT_RT/2 + DAT_RT/4)$
3	$DAT_MT/32 + (DAT_RT/2 + DAT_RT/4)$
4	$DAT_MT/16 + (DAT_RT/2 + DAT_RT/4)$
5	$DAT_MT/8 + DAT_RT/2$
6	$DAT_MT/4 + DAT_RT/2$
7	$DAT_MT/2 + DAT_RT/2$

En la versión para la Fusion, se utilizan tres pulsadores: Reset general, selección de parámetro y aumento del valor del parámetro seleccionado. En la versión de la RVI Prototype Board, se agregan dos pulsadores más: Selección de efecto y disminución del valor del parámetro. Adicionalmente, los cuatros displays de siete segmentos muestran los mnemónicos de los efectos y de los parámetros a editar con el valor almacenado, y también la cuerda digitada indicando la letra en notación americana para el afinador.

IV. RESULTADOS.

Se logró implementar con éxito un afinador y dos efectos con excelente funcionalidad, integrándolas a un diseño previo. Los reportes de la síntesis en la FPGA ProASIC3E (junto al diseño de referencia) y en la Fusion (con cada aplicación sintetizada por separado) se detallan en la Tabla 7.

TABLA 7. REPORTE DE RECURSOS

Recurso	ProASIC3E	Fusion
CORE cells	5981 de 38400 (16%)	2532 de 38400 (7%)
IO(W/clocks)	36	74 de 252 (29%)
RAM/FIFO	60 de 60 (100%)	60 de 60 (100%)

V. REFERENCIAS.

- [1] Documentos técnicos de <http://www.harmony-central.com>.
- [2] M. González, D. Costa, C. Sosa Páez, "Procesamiento de Señales para Efectos de Audio con Lógica Programable", Congreso de uEA 2010, Tercer Milenio. ISBN 978-978-9374-65-8
- [3] Specification for the WISHBONE System-on-Chip (SoC), Interconnection Architecture for Portable IP Cores; Revision: B.3, Released: September 7, 2002
- [4] A. Cicuttin, M. L. Crespo, A. Shapiro y N. Abdallah, "A block-based open source approach for a reconfigurable virtual implementation platform using FPGA technology", IEEE International Conference on Reconfigurable Computing and FPGA's ReConFig 2006, pp 1 a 8.
- [5] Documentos técnicos de Actel Corporation, <http://www.actel.com>
- [6] Manuales de uso del Libero, Designer, Flash Pro y ModelSim.