

## Arquitecturas de Sistemas Digitales 2013-2014

### Trabajo – Sintetuzo V1.0

### Sintetizador digital de audio

#### Objetivos del trabajo:

El objetivo de este trabajo consiste en el diseño de un sintetizador digital de audio polifónico (nº de canales a determinar) y basado en arquitectura clásica de síntesis substractiva.

Para ello, el alumno debe analizar distintas alternativas para solucionar el problema basándose en dispositivos como microcontroladores de bajas prestaciones (tipo AVR, como ATmega644) y FPGAs de pequeño tamaño (tipo Xilinx Spartan 3).

El principal objetivo es la toma de decisiones en cuanto al de diseño arquitectural del sistema, pudiendo llegar a una solución puramente *software*, puramente *hardware*, o mixta HW-SW. El grupo decidirá qué elementos serán los encargados de realizar cada tarea (generación de las ondas, envolventes, etc.) y dónde serán mapeados; si en el uC, la FPGA o en ambos.

#### Recursos disponibles

Los principales recursos disponibles son:

1. Una placa (llamada Nonaina) que incorpora un microcontrolador ATmega644, una FPGA XC3S200, conector USB para conectar a un PC, conectores de programación JTAG, un conector de expansión de 34 pines, y conectores compatibles con *shields* de Arduino (ver Figura 1)

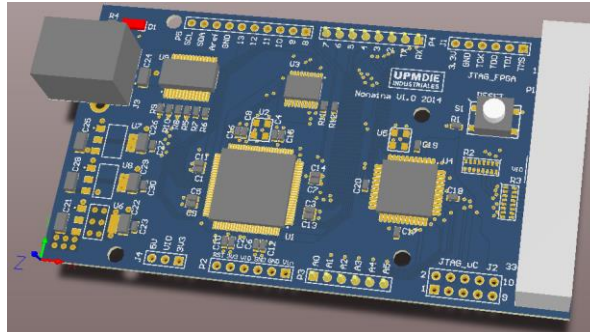


Figura 1: Placa de procesamiento.

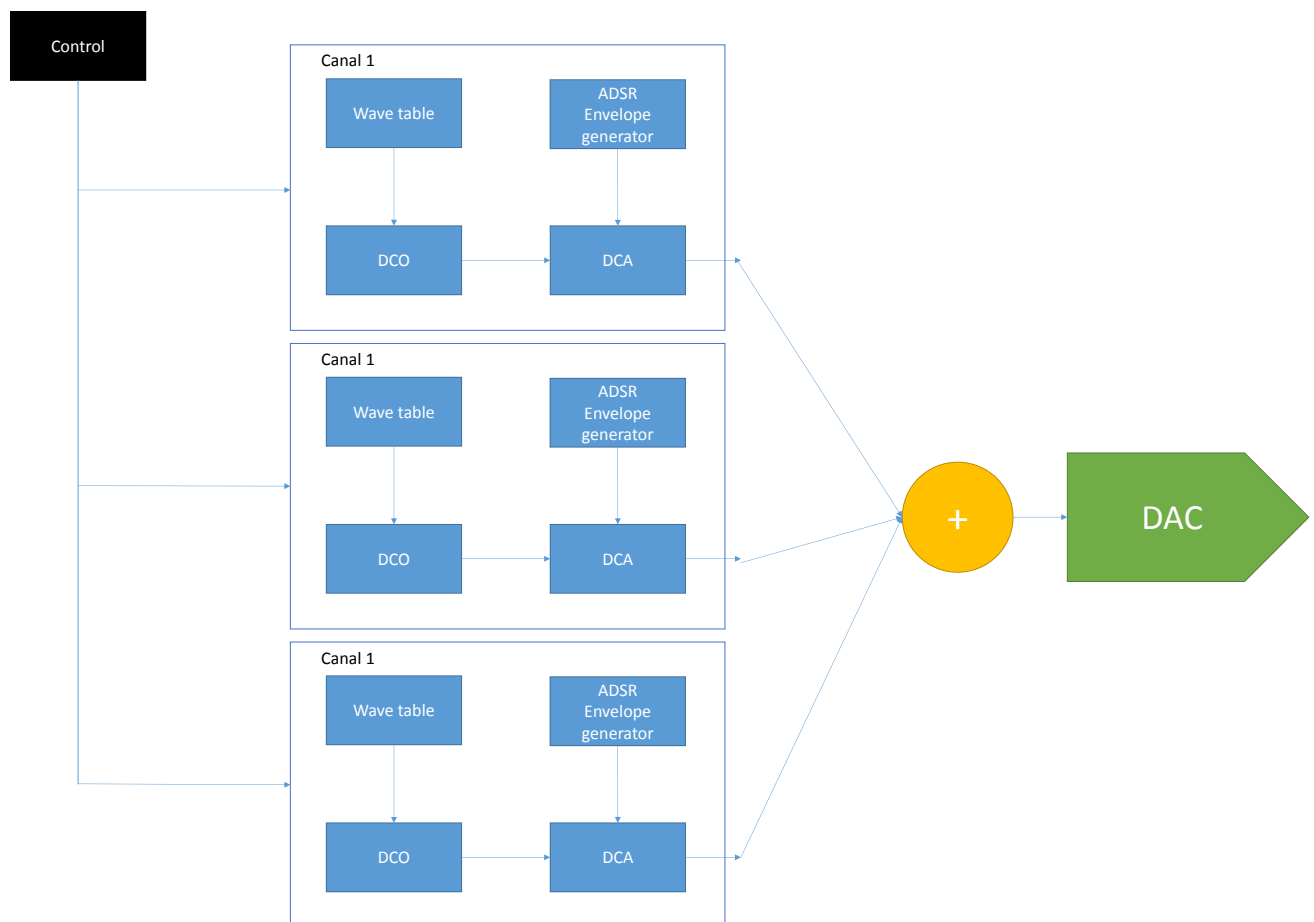
2. Una aplicación (Figura 2) que corre en el PC donde se muestra el panel de control de un secuenciador de audio. Esta aplicación se comunica con la placa Nonaina a través de un puerto serie virtual y de un bridge MIDI (*Musical Instrument Digital Interface*)-puerto serie, a una velocidad de 38400 bps, de modo que la Nonaina pueda recibir los comandos **MIDI** estándar. Este puerto serie se crea gracias a la incorporación de un chip **FTDI** (convertor UART a USB) en la placa Nonaina. De todos los comandos MIDI disponibles, sólo será necesario interpretar los comandos **NOTE\_ON**, **NOTE\_OFF** y **CONTROL\_PARAM**, **CONTROL\_CHANGE**. En cualquier caso se aportará una biblioteca SW en C para la placa de API de Arduino, que se puede utilizar como referencia, o como los alumnos estimen necesario.



Figura 3: Pantallazo de la aplicación que corre en el PC, llamada [LMMS](#).

- Además será necesario descargar la aplicación [loopMIDI](#) para poder conectar aplicaciones MIDI entre sí dentro del PC (en nuestro caso el secuenciador LMMS y el [Bridge MIDI serial](#) con el que podremos enviar desde cualquier aplicación, por el puerto serie y monitorizar los comandos MIDI).
- La salida de audio se generará inicialmente a través de PWM o mediante un bloque  $\Delta\Sigma$  (sigma-delta).

El diagrama de bloques preliminar del sintetizador es el siguiente:



Los mensajes MIDI llegan al control, y éste se encarga de interpretarlos y de generar el audio de acuerdo a los mismos.

### Evaluación

Para la evaluación se considerará el análisis de la solución hecha por cada grupo de alumnos. Se deberá entregar un documento explicativo de por qué se ha elegido la solución planteada, qué partes se realizan en HW o SW y por qué, puntos fuertes y débiles del diseño, etc.

Se entregará también todo el código que se haya realizado, con explicación de cada parte (función, bloque vhdl, etc.). Si se considera necesario, se entregarán pantallazos de simulaciones o capturas de osciloscopio.

No es requisito imprescindible que el sistema funcione totalmente, pueden funcionar partes del mismo. El funcionamiento total se considerará para mejorar la nota final.