Máster Universitario en Ingeniería de Sistemas Electrónicos

**Imagen que contiene Texto

Descripción generada automáticamente**

**Práctica E5: Escritura y lectura de registros de configuración**

Jose Luis, Rocabado Rocha

Gianmarco Leopoldo, Sangoi Da Roza

05/30/2022

# **ÍNDICE**

[ÍNDICE 2](#_Toc102123082)

[ÍNDICE DE FIGURAS 3](#_Toc102123083)

[Descripción del módulo 4](#_Toc102123084)

[Bloques realizados 8](#_Toc102123085)

[Problemas encontrados 8](#_Toc102123086)

# **ÍNDICE DE FIGURAS**

[Fig. 1. Respuesta en frecuencia del filtro CIC de tres etapas implementado en la práctica anterior. 4](#_Toc102123116)

[Fig. 2. Respuesta en frecuencia del filtro compensador de 17 coeficientes. 5](#_Toc102123117)

[Fig. 3. Zoom de la simulación de Simulink que compara el modelo ideal con el modelo cuantificado con precisión completa. 6](#_Toc102123118)

[Fig. 4. Arriba: Plot de los coeficientes del filtro cuantificados. 6](#_Toc102123119)

[Fig. 5. Resultados de la señal filtrada para una señal senoidal de frecuencia 1kHz, 10kHz y 15 kHz. 7](#_Toc102123120)

[Fig. 6. Snippet de código verilog que implementa el multiplicador acumulador. 8](#_Toc102123121)

# Descripción del módulo

En esta práctica se realizará el módulo que nos permitirá comunicar la computadora con el dispositivo FPGA. Este módulo permitirá que podamos escribir y leer los registros de configuración del modulador AM/FM a través de un puerto serie RS232.

El módulo elaborado consta de 4 bloques principales:

Diagrama

Descripción generada automáticamente

* **PLL**

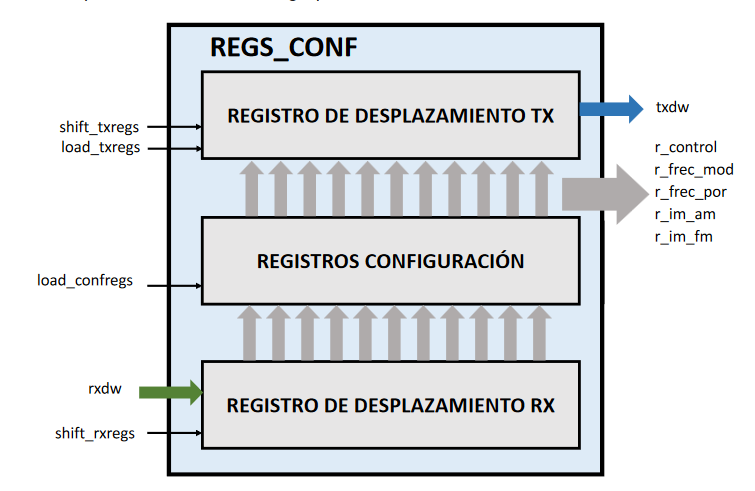
Este bloque es el encargado de proporcionar la frecuencia de reloj de 230400KHz a partir del reloj de 50MHz de la placa DE2-115 (Ya dado por el docente de la asignatura). Su instanciación realmente será afuera del módulo elaborado: CONF\_CONTROL, esto se debe a que se usará un módulo TOP proporcionado por el docente para poder instanciar nuestro modulo junto al PLL, los switches y LEDs de la placa.

* **RS232COM**

Este bloque descrito en HDL es el encargado de realizar la capa física para la comunicación RS232 (Ya dado por el docente de la asignatura).

* **REGS\_CONF**

Este bloque es el que contiene los registros para la configuración del modulador AM/FM. Esta diseñado en base a 3 registros de 11 bytes, desplazamiento de lectura, desplazamiento de escritura y el registro de configuración. En el siguiente diagrama se podrá observar su estructura:



* **CONTROL\_FSMs**

Este bloque se basa en 3 maquinas de estados que controlan el envío y recepción de los bytes del REGS\_CONF:

* Máquina de estado MAIN\_CONTROL (principal):

Imagen que contiene Tabla

Descripción generada automáticamente

Imagen que contiene diferente, tabla, diversos, pequeño

Descripción generada automáticamente

* Máquina de estado RD\_CONTROL (escritura):

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja

Un grupo de personas con instrumentos musicales y micrófonos en un escenario

Descripción generada automáticamente con confianza baja

* Máquina de estado WR\_CONTROL (lectura):

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

Diagrama

Descripción generada automáticamente

El funcionamiento conjunto de las máquinas de estado es de la siguiente manera:

Primero el bloque se encarga de detectar el primer byte transmitido desde la PC (MATLAB) a nuestra FPGA, este nos indica si el procedimiento es el de lectura o escritura del REGS\_CONF:

Diagrama, Forma, Rectángulo

Descripción generada automáticamenteDiagrama, Forma, Rectángulo

Descripción generada automáticamente

Una vez que se interpreta este comando se procede a realizar el desplazamiento de bytes de los registros de lectura y escritura que posee el bloque REGS\_CONF, lo hace en base al siguiente orden:

Diagrama

Descripción generada automáticamente

Si el procedimiento es de escritura primero realiza el desplazamiento de los 11 bytes al registro de escritura y luego los carga en el registro de configuración para poder configurar nuestro modulador AM/FM. Si el procedimiento es de lectura, primero se carga los bytes en el registro de lectura desde el registro de configuración y luego se desplazan 1 a 1 para poder ser leídos.

# Interfaz

Texto, Tabla

Descripción generada automáticamenteTexto

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Tabla

Descripción generada automáticamenteTexto, Tabla

Descripción generada automáticamente con confianza media

# Recursos estimados

Los recursos de nuestro modulo CONF\_CONTROL fueron los siguientes:

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

# Frecuencia máxima

La placa utilizada en el laboratorio de prácticas consiste en una Cyclone IV DE-115 cuya frecuencia máxima es de 250MHz.

Mediante la herramienta *TimeQuest Timing Analyzer*, podremos calcular la frecuencia máxima a la que trabajará el modelo diseñado. Utilizando el *wrapper* (instanciación con registros en las entradas y salidas) creado, hemos sido capaces de obtener una frecuencia máxima de trabajo de ~217.2MHz tal y como se observa en la siguiente figura.

# Camino critico

El camino critico es aquel en que la señal tarda más desde su ingreso hasta su destino. Mediante el *Technology Map Viewer* podremos encontrar en que parte del circuito se encuentra.

El camino critico de nuestro módulo se encuentra en la etapa de los filtros integradores, en esta etapa hay sumas de 38 bits que producen un acarreo que tiene que recorrer 38 posiciones, esto hace mucho más largo el recorrido de la señal en la etapa INT que en la etapa de filtros peine (COMB).

# Verificación

Para la verificación de esta entrega se realizaron 3 tipos de verificaciones distintas:

1. Verificación Testbenchs:

El primer método de verificación utilizado fue el de testbenchs, se procedió a realizar un testbench de cada modulo para poder simular en MODELSIM cada módulo, finalmente el módulo top del proyecto fue simulado con las entradas que proporciona el fichero de MATLAB dado por el docente, este fichero contenía los 11 bytes de testeo a enviar mas el byte de selección de modo (lectura/escritura).

Diagrama, Esquemático

Descripción generada automáticamente

1. Verificación física:

Al diseñar nuestra maquina de estados se le asigno a cada estado un valor de 4 bits que representa 4 LEDs de la placa. Esto se hizo para que en conjunto con el modulo top proporcionado por el docente (top\_DE2115) y el archivo de asignación de pines se pudiera verificar el funcionamiento del módulo físicamente.

Diagrama, Esquemático

Descripción generada automáticamente

Como se puede observar los verdes LEDs (del 0 al 8) nos dicen el estado, 4 switches seleccionan el byte a mostrar por 8 LEDs rojos (del 0 al 7) y los LEDs rojos de 8 al 15 nos indican el byte recibido por el puerto serie.

1. Verificación con SignalTap:

Esta verificación se hace mediante la herramienta *SignalTap* que proporciona *Quartus*, con ella podemos observar la transmisión y recepción de datos que realiza nuestro modulo simplemente cargando el diseño en la placa y corriendo el fichero de Matlab con los bytes de configuración de prueba.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

El reloj utilizado para la configuración de esta herramienta fue el que arroja la salida del bloque PLL (25KHz aprox.), y para poder visualizar la recepción y transmisión se utilizaron las señales rxrdy y txena respectivamente como señales de disparo cuando están a “1”. Al cargar el fichero en Matlab con los bytes a trasmitir se puede observar como el módulo recibe y transmite la información en 11 bytes.

# Resolución de problemas encontrados

En cuanto a los problemas encontrados el principal fue que al realizar la simulación en *Modelsim* de nuestro modulo Top CONF\_CONTROL se vio como había un problema, esto era debido a que a la hora de elaborar el código verilog para el módulo de REGS\_CONF no se había hecho de una manera óptima (se colocaban 3 registros de 87 bits) lo cual no solamente nos arrojaba un error, sino que también dificultaba su testeo, esto fue arreglado cambiando el código por la estructura dada en clase:

Diagrama

Descripción generada automáticamente