

Memoria Práctica E6

Modulador FM-AM configurable basado en FPGA

David Martínez Esteso

Néstor García García

Tabla de contenido

Sección 1: Descripción del módulo.	3
Sección 2: Interfaz.	5
Sección 3: Recursos hardware.	6
Sección 4: Frecuencia de operación.	7
Sección 5: Verificación.....	8
Sección 6: Resolución de problemas encontrados.	10

Sección 1: Descripción del módulo.

En esta última práctica de la asignatura Procesado Digital de la Señal en FPGA se lleva a cabo la completa implementación del modulador configurable FM-AM desarrollado en las prácticas anteriores, y la verificación del funcionamiento de éste a través de la tarjeta DE2-115.

La estructura del proyecto completo es la siguiente:

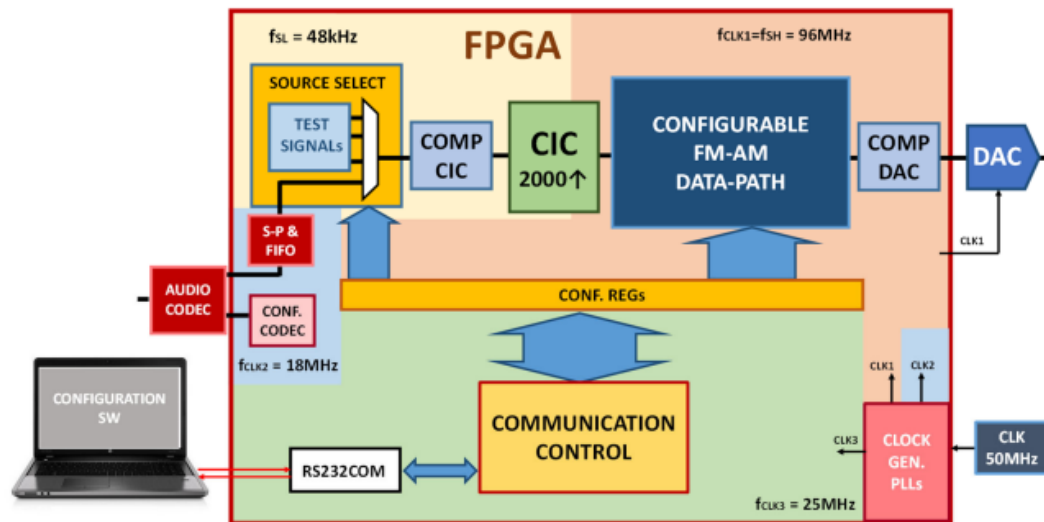


Ilustración 1: estructura completa modulador configurable

La estructura del bloque a implementar para el proyecto completo del modulador, donde se muestran las conexiones entre los bloques desarrollados en las prácticas anteriores junto a los formatos numéricos de estas conexiones es la siguiente:

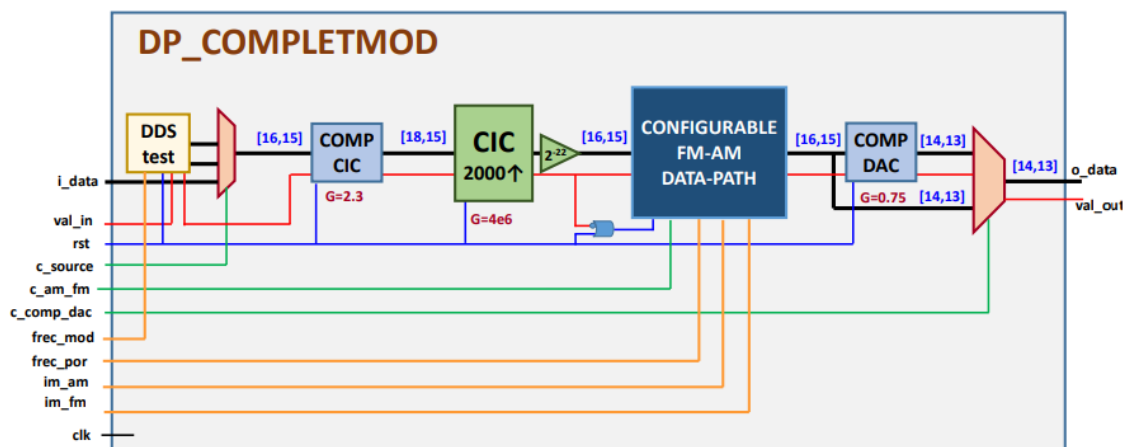


Ilustración 2: estructura bloque proyecto completo DP_COMPLETOMOD

En nuestro caso no se ha desarrollado el filtro FIR compensador del DAC por lo que la salida total del bloque DP_COMPLETOMOD viene dada por la salida del bloque del data-path.

A continuación se muestra el diagrama de interconexión entre los bloques de instanciación en el código MOD_COMP entre las líneas 95 y 174:

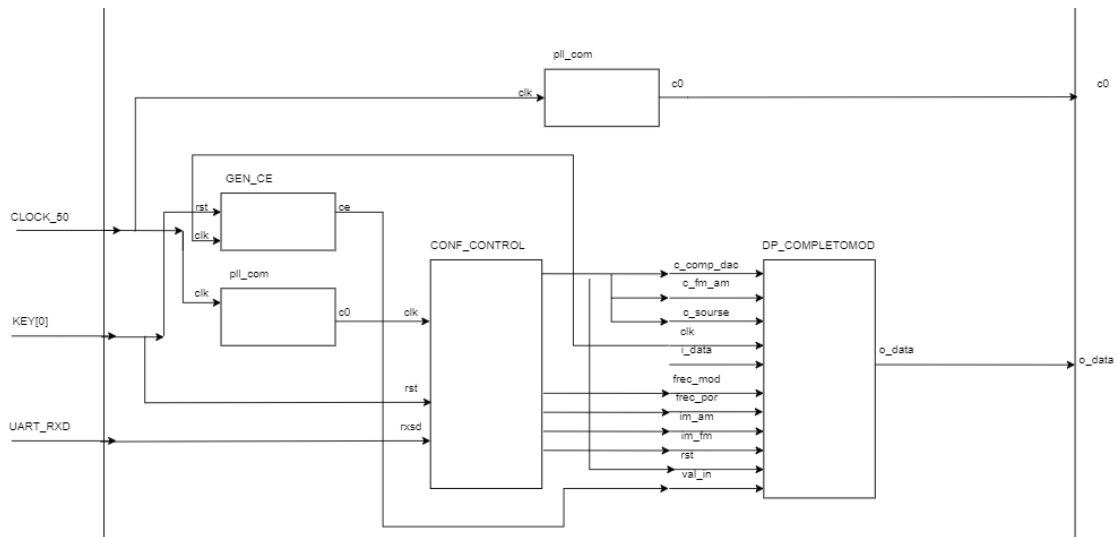


Ilustración 3: diagrama de interconexión bloques código MOD_COMP

El bloque GEN_CE se ocupa de generar la señal de habilitación para el módulo DP_COMPLETOMOD, utilizando una frecuencia de reloj de 48 kHz, es decir habilita la señal cada 2000 ciclos del reloj global.

Sección 2: Interfaz.

En la siguiente sección se define el interfaz, sus formatos y sus parámetros del proyecto Verilog:

Módulo DP_COMPLETOMOD: INTERFAZ			
Nombre	Tipo	Formato	Descripción
i_data	in	S[16,0]	Entrada de datos a modular
rst	in	bit	Reset síncrono, activo a nivel alto
clk	in	bit	Entrada de reloj
val_in	in	bit	Señal de validación de la muestra de entrada
c_fm_am	in	bit	Entrada configuración modulador FM/AM
c_source	in	U[2,0]	Señal de selección primer multiplexor
c_comp_dac	in	bit	Señal de selección segundo multiplexor
frec_mod	in	U[24,0]	Paso del DDS para generar la frecuencia de las señales de test
frec_por	in	U[24,0]	Entrada del valor de configuración de la frecuencia portadora
im_am	in	U[16,0]	Entrada de configuración del índice de modulación de AM
im_fm	in	U[16,0]	Entrada de configuración del índice de modulación de FM
o_data	out	U[14,0]	Salida de datos modulados
val_out	out	bit	Señal de validación de la muestra de salida

Sección 3: Recursos hardware.

Teniendo en cuenta los numerosos ficheros utilizados en el desarrollo del proyecto se estima el uso de una gran cantidad de elementos lógicos (LEs).

Una vez compilado el proyecto en Quartus se ha ejecutado el sumario del uso de recursos total como se muestra a continuación:

Resource	Usage
Estimated Total logic elements	3,030
Total combinational functions	1870
Logic element usage by number of LUT inputs	
-- 4 input functions	713
-- 3 input functions	692
-- <=2 input functions	465
Logic elements by mode	
-- normal mode	1380
-- arithmetic mode	490
Total registers	2368
-- Dedicated logic registers	2368
-- I/O registers	0
I/O pins	101
Total memory bits	761901
Embedded Multiplier 9-bit elements	8
Total PLLs	3
-- PLLs	3

Ilustración 4: sumario recursos proyecto Quartus

Comentando la tabla anterior el proyecto hace uso de 1387 registros y un total de 1740 elementos lógicos (LEs), como hemos supuesto se hace uso de una gran cantidad de recursos en el presente proyecto.

En cuanto a las memorias utilizadas se muestra a continuación el listado de estas:

Name	Type	Mode	Port A Depth	Port A Width	Port B Depth	Port B Width	Size
DP_COMPLETMOD:U1 DDS_testdds_test rom_syncram_a471:auto_generated ALTSYNCRAM	AUTO	ROM	8192	14	--	--	114688
DP_COMPLETMOD:U1 DP_MOD:dp_mod DDS.D...ncram_6671:auto_generated ALTSYNCRAM	AUTO	ROM	8192	16	--	--	131072
DP_COMPLETMOD:U1 SEC_FILTER:sec_filter a...d altsyncram_1131:altsyncram_4 ALTSYNCRAM	AUTO	Simple Dual Port	3	15	3	15	45
audio_codec:codec Altera_UP_Audio_In_Des...pfifo altsyncram_9tb1:FIFOram ALTSYNCRAM	AUTO	Simple Dual Port	128	24	128	24	3072
audio_codec:codec Altera_UP_Audio_Out_Se...pfifo altsyncram_9tb1:FIFOram ALTSYNCRAM	AUTO	Simple Dual Port	128	24	128	24	3072
audio_codec:codec Altera_UP_Audio_Out_Se...pfifo altsyncram_9tb1:FIFOram ALTSYNCRAM	AUTO	Simple Dual Port	128	24	128	24	3072
slid_signaltap:auto_signaltap_0 slid_signalt...altsyncram_ge24:auto_generated ALTSYNCRAM	AUTO	Simple Dual Port	16384	31	16384	31	507904

Ilustración 5: sumario memorias proyecto Quartus

Sección 4: Frecuencia de operación.

La frecuencia de operación de los distintos relojes utilizados en el proyecto son las siguientes:

Slow 1200mV 85C Model				
	Fmax	Restricted Fmax	Clock Name	Note
1	89.28 MHz	89.28 MHz	pll_inst1 altpll_component auto_generated pll1 clk[0]	
2	149.81 MHz	149.81 MHz	clock_m	
3	191.24 MHz	191.24 MHz	pll_inst2 altpll_component auto_generated pll1 clk[0]	

Se observa una alta frecuencia en cada uno de los relojes lo cual no supondrá ningún problema en la ejecución y funcionamiento de nuestro proyecto.

El camino crítico obtenido del proyecto es el siguiente:

Slow 1200mV 85C Model									
Command info		Summary of Paths							
	Slack	From Node	To Node	Launch Clock	Latch Clock	Relationship	Clock Skew	Data Delay	
1	34.771	CONF_CONTROL:U2 CONNECT_FSMs:C3 RD_CONTROL:C3 state.desplaza	CONF_CONTROL:U2 REGS_CONF:C2 reg_array_txregs[3][2]	pll_inst_1 clk[0]	pll_inst_1 clk[0]	40.000	-0.084	5.146	
2	34.771	CONF_CONTROL:U2 CONNECT_FSMs:C3 RD_CONTROL:C3 state.desplaza	CONF_CONTROL:U2 REGS_CONF:C2 reg_array_txregs[4][2]	pll_inst_1 clk[0]	pll_inst_1 clk[0]	40.000	-0.084	5.146	
3	34.771	CONF_CONTROL:U2 CONNECT_FSMs:C3 RD_CONTROL:C3 state.desplaza	CONF_CONTROL:U2 REGS_CONF:C2 reg_array_txregs[5][2]	pll_inst_1 clk[0]	pll_inst_1 clk[0]	40.000	-0.084	5.146	
4	34.771	CONF_CONTROL:U2 CONNECT_FSMs:C3 RD_CONTROL:C3 state.desplaza	CONF_CONTROL:U2 REGS_CONF:C2 reg_array_txregs[6][2]	pll_inst_1 clk[0]	pll_inst_1 clk[0]	40.000	-0.084	5.146	
5	34.864	CONF_CONTROL:U2 CONNECT_FSMs:C3 RD_CONTROL:C3 state.desplaza	CONF_CONTROL:U2 REGS_CONF:C2 reg_array_txregs[2][2]	pll_inst_1 clk[0]	pll_inst_1 clk[0]	40.000	-0.086	5.051	
6	34.864	CONF_CONTROL:U2 CONNECT_FSMs:C3 RD_CONTROL:C3 state.desplaza	CONF_CONTROL:U2 REGS_CONF:C2 reg_array_txregs[3][6]	pll_inst_1 clk[0]	pll_inst_1 clk[0]	40.000	-0.086	5.051	
7	34.864	CONF_CONTROL:U2 CONNECT_FSMs:C3 RD_CONTROL:C3 state.desplaza	CONF_CONTROL:U2 REGS_CONF:C2 reg_array_txregs[4][6]	pll_inst_1 clk[0]	pll_inst_1 clk[0]	40.000	-0.086	5.051	
8	34.864	CONF_CONTROL:U2 CONNECT_FSMs:C3 RD_CONTROL:C3 state.desplaza	CONF_CONTROL:U2 REGS_CONF:C2 reg_array_txregs[5][6]	pll_inst_1 clk[0]	pll_inst_1 clk[0]	40.000	-0.086	5.051	
9	34.864	CONF_CONTROL:U2 CONNECT_FSMs:C3 RD_CONTROL:C3 state.desplaza	CONF_CONTROL:U2 REGS_CONF:C2 reg_array_txregs[7][6]	pll_inst_1 clk[0]	pll_inst_1 clk[0]	40.000	-0.086	5.051	
10	34.866	CONF_CONTROL:U2 CONNECT_FSMs:C3 RD_CONTROL:C3 state.desplaza	CONF_CONTROL:U2 REGS_CONF:C2 reg_array_txregs[0][6]	pll_inst_1 clk[0]	pll_inst_1 clk[0]	40.000	-0.087	5.048	

Se encuentra entre el módulo de conexiones de las máquinas de estados y el módulos de los registros para los bytes de configuración.

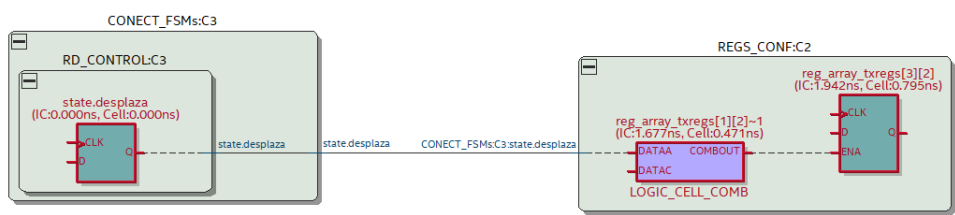


Ilustración 6: Visión RTL del camino crítico del proyecto

Sección 5: Verificación.

La verificación se ha llevado en primer lugar, mediante test bench en ModelSim. Se ha realizado con los siguientes valores de configuración:

Control FM --> 1, AM --> 0

Sampling frequency (MHz) = 100 MHz

Carrier frequency (MHz) = 1 MHz

AM modulation index (range $[0,1[$) = 1

FM modulation index (kHz) = 5000 kHz

Modulator source signal: SIN ->0; RAMP->1; SQR->2; (source_sel = 0)

Modulating frequency (kHz) (fmod= 10)

Reset (rst = 1)

Simulación AM:

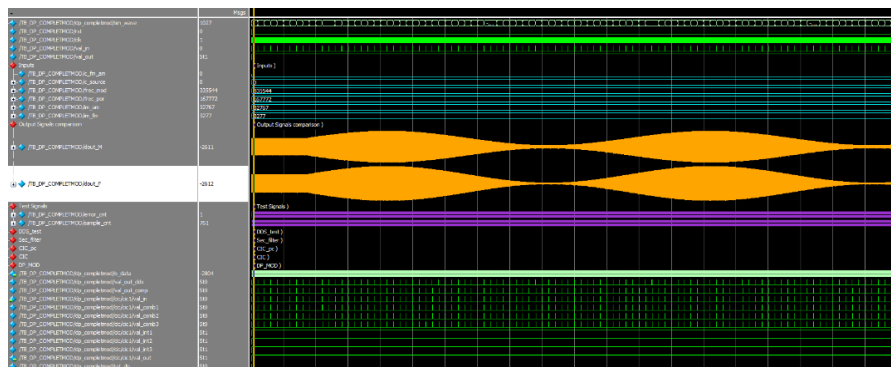


Ilustración 7: formas de onda sistema completo modulación AM

Simulación FM:

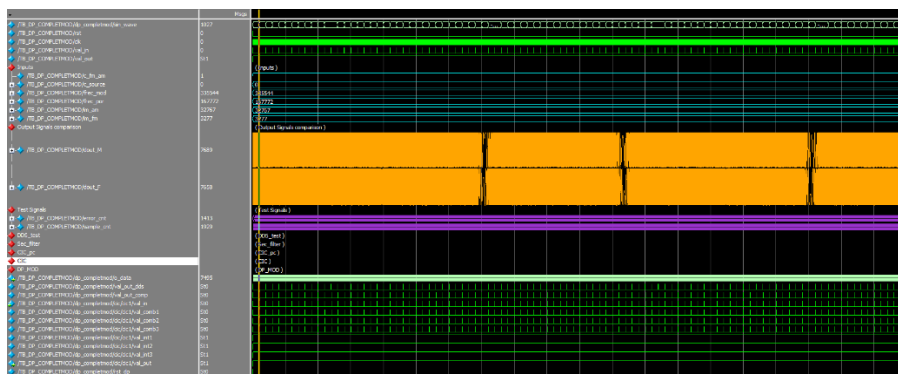


Ilustración 8: formas de onda sistema completo modulación FM

Posteriormente, se ha probado en SignalTap. Para ello se ha seleccionado como señal de reloj la señal c0 del pll1 (generación de reloj de 48 kHz) y se ha seleccionado adquisición continua. Se han obtenido los siguientes resultados:

Modulación AM con índice de modulación AM = 0.5.

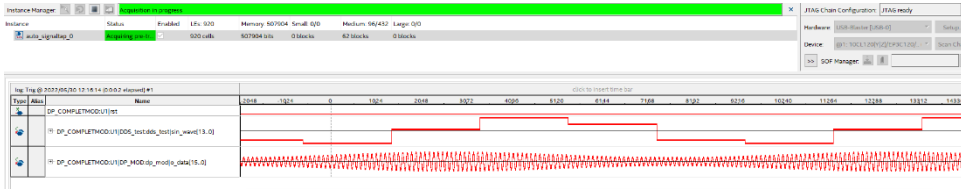


Ilustración 9: simulación SignalTap modulación AM con índice 0.5

Modulación AM con índice de modulación AM = 1.

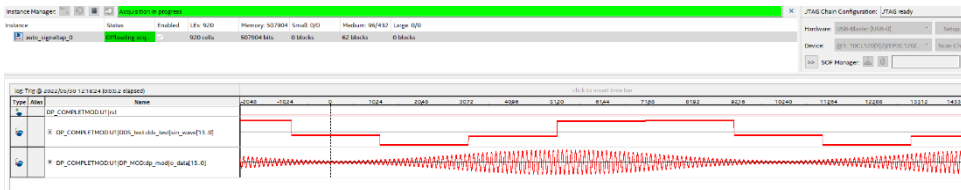


Ilustración 10: simulación SignalTap modulación AM con índice 1

Modulación FM con índice de modulación FM = 2500 kHz

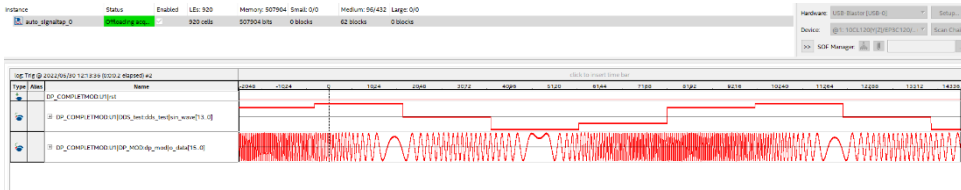


Ilustración 11: simulación SignalTap modulación FM con índice 2500 kHz

Modulación FM con índice de modulación FM = 5000 kHz

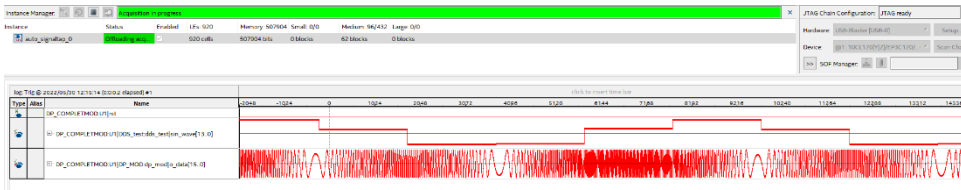


Ilustración 12: simulación SignalTap modulación FM con índice 5000 kHz

Se comprueba en las simulaciones el correcto funcionamiento de la modulación al comprimirse o ensancharse el espectro en cada caso.

Sección 6: Resolución de problemas encontrados.

A continuación, se presentan los problemas encontrados en la realización de la práctica y su solución:

- Se tuvo un problema con el truncado de la salida del CIC, puesto que el truncado se realizaba de forma distinta a como se realizó en la práctica 3.
- La salida no coincide de forma exacta con la de Matlab llegados a cierto punto en la simulación, se considera que puede deberse a errores de redondeo.
- Para la simulación con SignalTap, desconocíamos el uso de adquisición de datos continua y en un principio no conseguíamos resultados.
- En Matlab en el envío de datos, en un principio se realizó en formato binario obteniendo resultados erróneos por lo que se tuvo que cambiar el formato de los datos a decimal.