

# Memoria Práctica E1

## Sintetizador Digital de Frecuencia: DDS

David Martínez Esteso

Néstor García García

Tabla de contenido

**Enunciado ..... 3**

**Sección:1 Descripción del módulo ..... 3**

**Sección 2: Interfaz ..... 4**

**Sección 3: Recursos Hardware ..... 5**

**Sección 4: frecuencia de operación ..... 7**

**Sección 5: Verificación..... 8**

**Sección 6: Resolución de problemas encontrados ..... 9**

## Enunciado

El objetivo de esta práctica es el modelar con el lenguaje Verilog un sintetizador de frecuencias (DDS) parametrizable, que genere 3 formas de onda: sinusoidal, rampa y cuadrada. El modelo del DDS deberá ser sintetizable y podrá ser implementado en un dispositivo FPGA funcionando a una frecuencia de reloj mayor de 125 MHz.

Esquema de implementación del DDS:

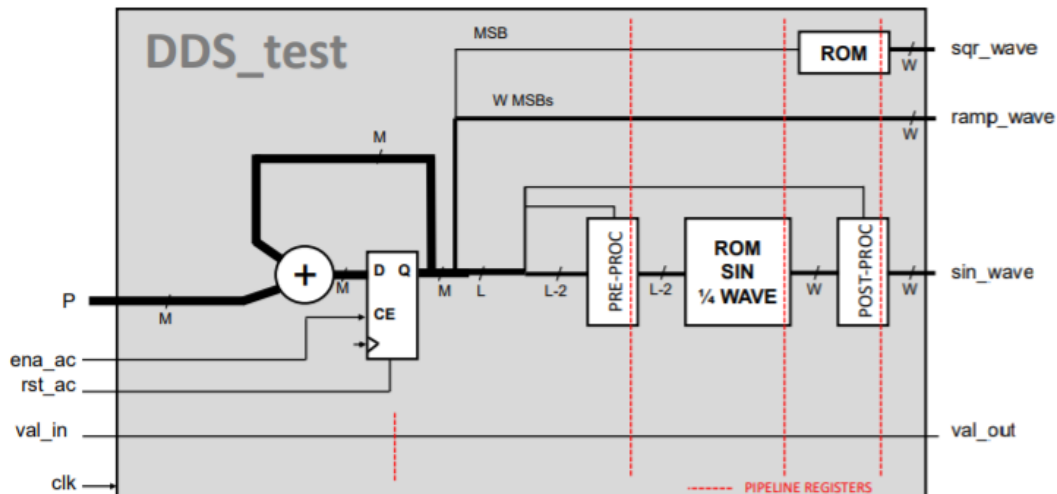


Ilustración 1: Esquema de implementación del DDS

## Sección:1 Descripción del módulo

El módulo desarrollado se ha seccionado en distintas partes según su función de generación de una onda cuadrada ( $sqr\_wave$ ), onda rampa ( $ramp\_wave$ ), onda senoidal ( $sin\_wave$ ) y salida de sincronización ( $val\_out$ ).

En primer lugar, todas las señales generadas parten de la salida de un acumulador, por lo que este es el primer bloque desarrollado. A la entrada del acumulador llega  $P$ , de formato  $U[M,M]$ , y su salida ( $acc\_out$ ) respeta dicho formato.

Para la generación de la onda cuadrada, se utiliza el MSB de la salida del acumulador ( $acc\_out$ ) y se registra con formato  $S[W,W-1]$ . Previo a su salida del bloque, se registra 3 veces adicionales dicha señal, respetando los pipelines, y no es necesario utilizar una memoria ROM tal y como se muestra en el esquema de la práctica.

Para la generación de la onda rampa, se utilizan  $W$  MSBs de la salida del acumulador ( $acc\_out$ ) y se registra con formato  $S[W,W-1]$ . Previo a su salida del bloque, se ha registrado 3 veces adicionales dicha señal, respetando los pipelines.

Para la generación de la onda senoidal se parte de la salida del acumulador con formato  $U[M,M]$ . En primer lugar, se truncan los  $L$  MSBs. Por un lado, del resultado del truncado ( $wire\_a$ ), los  $L-2$  bits menos significativos se llevan a un bloque de preproceso que a su salida obtiene una señal triangular ( $wire\_b$ ) que es la entrada del bus de direcciones de

la ROM. A la salida de esta, se obtiene una señal redondeada a partir de la triangular de la entrada. Por otro lado, los 2 MSBs se registran dos veces (respetando los pipelines) para llegar al bloque de postproceso junto a la salida de la ROM, donde se conforma finalmente la señal senoidal, de formato  $S[W, W-1]$ .

Finalmente, para la señal de sincronización ( $val\_out$ ), simplemente se registra la entrada ( $val\_in$ ) 4 veces (respetando los pipelines).

## Sección 2: Interfaz

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

Módulo DDS_test: PARÁMETROS	
Nombre	Descripción
<b>M</b>	Tamaño del acumulador
<b>L</b>	Número de bits usados para truncar la fase del acumulador
<b>W</b>	Tamaño de los datos de salida

Módulo DDS_test: INTERFAZ			
Nombre	Tipo	Formato	Descripción
<b>clk</b>	in	bit	Entrada de reloj
<b>rst_ac</b>	in	bit	Rset síncrono del acumulador, activo a nivel alto
<b>ena_ac</b>	in	bit	Clock enable del acumulador, activo a nivel alto
<b>val_in</b>	in	bit	Entrada de validación de la muestra de entrada
<b>P</b>	in	$U[M, M]$	Paso del acumulador
<b>sin_wave</b>	out	$S[W, W-1]$	Señal senoidal de frecuencia $f_o = P \cdot f_{clk} / 2^M$
<b>sqr_wave</b>	out	$S[W, W-1]$	Señal cuadrada de frecuencia $f_o = P \cdot f_{clk} / 2^M$
<b>ramp_wave</b>	out	$S[W, W-1]$	Señal triangular de frecuencia $f_o = P \cdot f_{clk} / 2^M$
<b>val_out</b>	out	bit	Señal de validación de la muestra de salida

## Sección 3: Recursos Hardware

Recursos totales utilizados (con wrap):

Analysis & Synthesis Summary	
<<Filter>>	
Analysis & Synthesis Status	Successful - Thu Mar 03 17:23:32 2022
Quartus Prime Version	17.1.0 Build 590 10/25/2017 SJ Lite Edition
Revision Name	E1
Top-level Entity Name	DDS_wrap_test
Family	Cyclone IV E
Total logic elements	182
Total registers	181
Total pins	74
Total virtual pins	0
Total memory bits	114,688
Embedded Multiplier 9-bit elements	0
Total PLLs	0

*Ilustración 2: Sumario de síntesis del código Verilog con fichero Wrap*

Les: 182

Mults: 0

Analysis & Synthesis RAM Summary								
<<Filter>>								
	Name	Type	Mode	Port A Depth	Port A Width	Port B Depth	Port B Width	Size
1	DDS_test:DDS1 rom_mem:m1 altsyncram:rom_tsyncram_ie61:auto_generated ALTSYNCRAM	AUTO	ROM	8192	14	--	--	114688

M9K blocks: 14

Recursos totales utilizados (sin wrap):

Analysis & Synthesis Summary	
<<Filter>>	
Analysis & Synthesis Status	Successful - Thu Mar 03 18:11:43 2022
Quartus Prime Version	17.1.0 Build 590 10/25/2017 SJ Lite Edition
Revision Name	E1
Top-level Entity Name	DDS_test
Family	Cyclone IV E
Total logic elements	121
Total registers	120
Total pins	74
Total virtual pins	0
Total memory bits	114,688
Embedded Multiplier 9-bit elements	0
Total PLLs	0

Ilustración 3: Sumario de síntesis del código Verilog sin fichero Wrap

LEs: 121

Mults: 0

Analysis & Synthesis RAM Summary									
<<Filter>>									
	Name	Type	Mode	Port A Depth	Port A Width	Port B Depth	Port B Width	Size	
1	DDS_test:DDS1 rom_mem:m1 altsyncram:rom...tsyncram_le61:auto_generated ALTSYNCRAM	AUTO	ROM	8192	14	--	--	114688	

M9K blocks: 14

Netlist viewer del bloque implementado con el wrap:

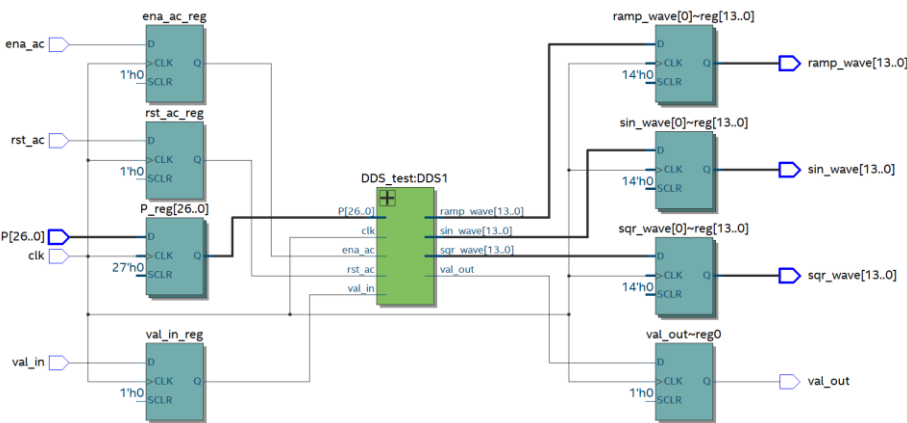


Ilustración 4: Estructura RTL del bloque implementado

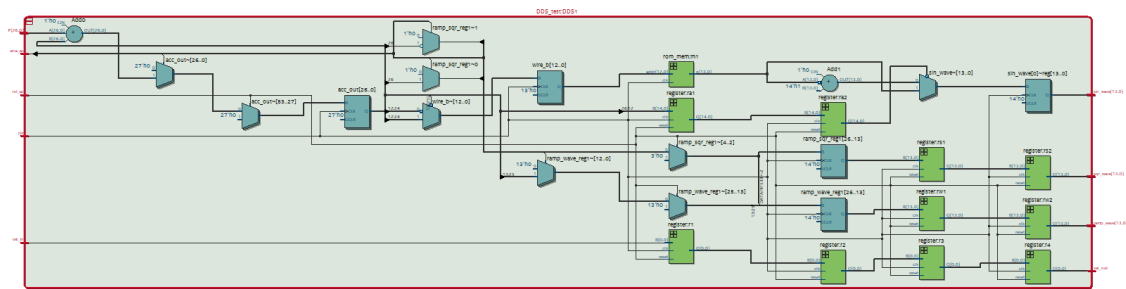


Ilustración 5: Estructura RTL del módulo DDS\_test

## Sección 4: frecuencia de operación

La frecuencia de operación obtenida es la siguiente:

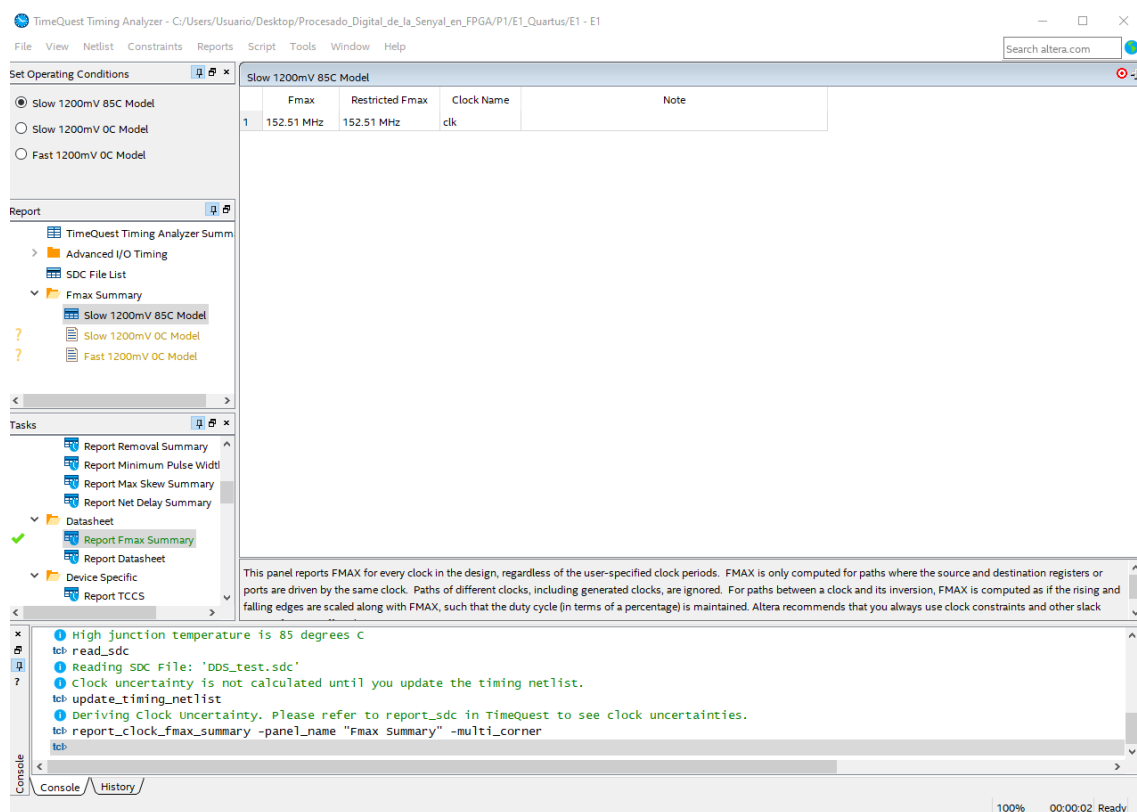


Ilustración 6: Frecuencia máxima de operación en el dispositivo

Se observa se cumple con el criterio de frecuencia de operación  $\geq 125$  MHz.

El camino crítico es el que llega a la salida de la onda senoidal (sin\_wave) a través de la memoria ROM, puesto que la lectura de dicha ROM se estima que tiene el mayor retardo del bloque.

# Sección 5: Verificación

Ondas de salida con parámetros M=16, L=6, W=16:

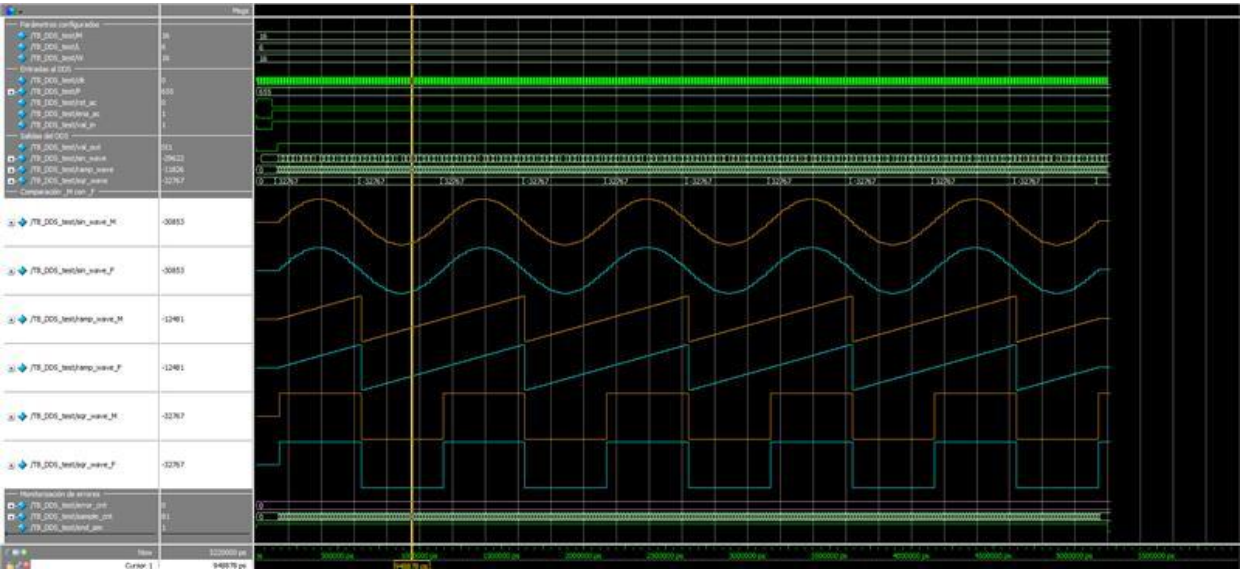


Ilustración 7: Ejemplo ondas de salida 1

Ondas de salida con parámetros M=27, L=15, W=14:

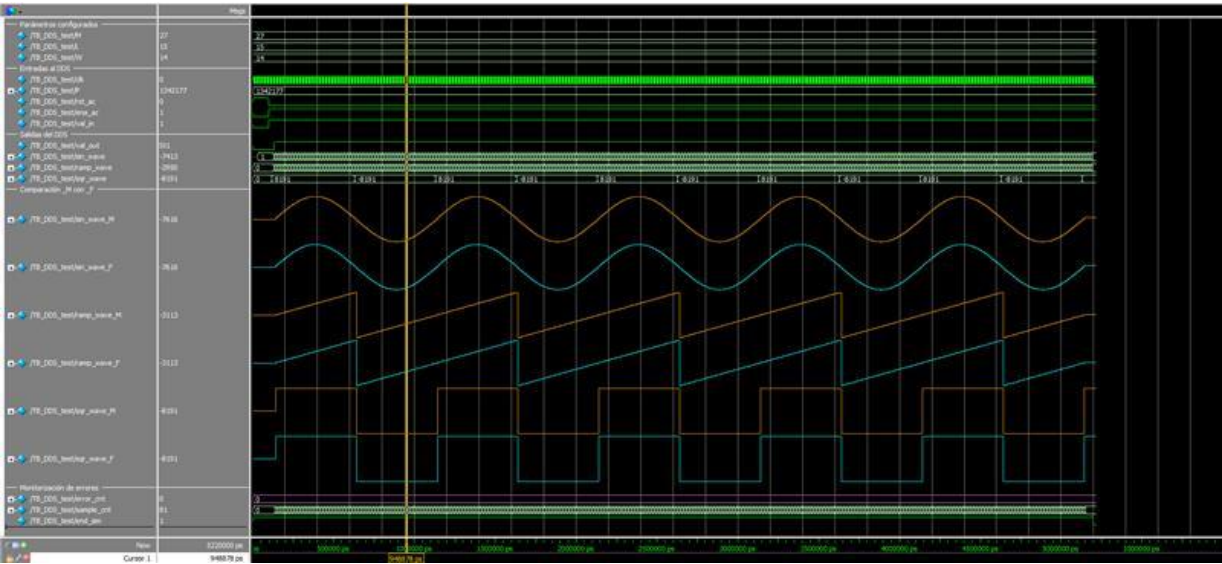


Ilustración 8: Ejemplo ondas de salida 2

En ambos casos puede observarse completa coincidencia con las formas de onda generadas a partir del banco de pruebas proporcionado.



## Sección 6: Resolución de problemas encontrados

En cuanto a los retardos producidos por el pipeline, se ha hecho uso de registros específicos para sincronizar las señales a lo largo del bloque. El número de registros correspondiente a cada bloque coincide con las líneas verticales rojas que se muestran en el esquema de implementación del DDS.

Por otro lado, una vez representadas las ondas en el testbench para corregir los errores en las no coincidencias entre gráficas, se han analizado las mismas para encontrar más fácilmente la causa de estas discrepancias y se han corregido en el código Verilog.