

## GUÍA MÓDULO 2: Circuitos contadores

La palabra “contar” parece de significado obvio, y a partir de allí los “contadores” surgen como si fueran también diseños donde es posible poca innovación. Pero, sin embargo, el significado de contar no es obvio, y la posible variedad de formas de contar, y de técnicas para hacerlo es enorme. De hecho, si bien un contador no es más que una máquina de estados con una simple secuencia de estados, el secreto es saber elegir esa secuencia.

Los contadores sirven para medir cantidad de eventos, lapsos de tiempo, relaciones entre frecuencias, generar secuencias de señales de control, y están asociados a señales de reloj y señales de control.

En este módulo se evalúan las ventajas y deficiencias de los contadores asincrónicos versus los sincrónicos. Y en el caso de estos últimos se presentan tres de las posibles topologías de conteo: binario natural, GRAY y LFSRs, y las comparaciones entre ellos.



### Actividad 2.1 (Entrega Obligatoria)

Busque información y describa cómo funcionan los siguientes dispositivos:

1. Para un mismo TCO evaluar la frecuencia máxima de operación de un contador asincrónico binario si se lo desea usar como divisor, o si importa el valor de conteo. ¿Cómo cambia este tiempo si el valor de reciclado no es potencia entera de dos?
2. Evaluar la complejidad circuital de implementar un contador sincrónico. En forma general indicar la ecuación lógica a implementar en una dada etapa “i”, para un contador UP/DOWN con ENABLE
3. Comparar la complejidad circuital de un contador sincrónico implementado en una FPGA con cadena de Carry (tipo Cyclone de ALTERA) versus una FPGA basada en macroceldas aisladas (tipo MAX3000A de Altera).

Buscar información de otros tipos de contadores (ring, Johnson, etc.) y en qué caso son de utilidad.



Utilice el recurso [Actividad 2.1](#) para enviar



### Actividad 2.2 (Entrega Obligatoria)

1. Implementar en VHDL una función que en base a los parámetros: cantidad de etapas, realimentación, valor inicial y tipo de realimentación (XOR/XNOR), devuelva el valor a esperar en la salida de un LFSR luego de N ciclos de reloj.
2. Sintetice usando QUARTUS un contador binario y un contador LFSR de 10 a 20 bits de ancho de palabra, y evalúe la FMAX obtenida en cada caso.
3. Elabore en VHDL y simule un contador UP de módulo 1.000.000, con entradas reset (activo en '1') y reloj (activo flanco +), que genere como salidas 10 señales activas en alto, libres de glitches, de un ciclo de reloj de duración, en las siguientes 10 ocasiones: 47, 2555, 7981, 91466, 18965, 215543, 327333, 510321, 777666, y 892345. Evaluar la solución con menor consumo de recursos usando una FPGA tipo Cyclone de Altera.



Utilice el recurso [Actividad 2.2](#) para enviar

### Consideraciones finales

El objetivo de este módulo es “abrir la cabeza” a comprender cómo el proceso de contar puede ser realizado de distintas maneras en función del objetivo de realizar esa cuenta, y que esta decisión no es obvia.

### Bibliografía sugerida:

- [http://en.wikipedia.org/wiki/Linear\\_feedback\\_shift\\_register](http://en.wikipedia.org/wiki/Linear_feedback_shift_register)
- [http://www.ee.ic.ac.uk/pcheung/teaching/ee4\\_asic/notes/EDN%20on%20LFSR.pdf](http://www.ee.ic.ac.uk/pcheung/teaching/ee4_asic/notes/EDN%20on%20LFSR.pdf)
- [http://www.xilinx.com/support/documentation/application\\_notes/xapp052.pdf](http://www.xilinx.com/support/documentation/application_notes/xapp052.pdf)
- <http://www.altera.com/support/examples/vhdl/vhd-gray-counter.html>
- [http://www.prevaling-technology.com/publications/counter\\_examples.pdf](http://www.prevaling-technology.com/publications/counter_examples.pdf)