

**Universidad de Costa Rica
Escuela de Ingeniería Eléctrica
Circuitos Digitales II
Prof. Jorge Soto
II Ciclo 2020
IE-0523**

Tarea #8

1. Código Gray

El código Gray corresponde a un sistema de numeración, diferente al que se tiene por convención ordinaria. Lo que lo hace diferente es que en este sistema se busca que dos números consecutivos difieran solamente en uno de sus dígitos. Al poseer esta característica donde difiere de su precesor por un único bit, lo hace óptimo para una serie de aplicaciones. Se aplica directamente al área de circuitos digitales ya que se emplean como parte del algoritmo de diseño en los Mapas de Karnaugh, los cuales sirven como herramientas de diseño a la hora de implementar circuitos ya sea combinacionales o secuenciales. El código Gray es funcional en los circuitos digitales dado que siempre se busca que un diseño digital sea lo más eficientemente posible, para ello requerirá transiciones que sean más simples y a su vez rápidas entre sus estados lógicos, tal como lo brindan los diseños con esta numeración.¹

Además de circuitos digitales, el código Gray es útil en otros ámbitos dadas sus características. Un ejemplo de su aplicación es el de la robótica, ya que esta enumeración es aplicable en sistemas de posición lineal o angular. La robótica se aprovecha de esto y realiza la transferencia de datos de información acerca de la posición que se posee de un eje en particular a través de discos codificados en código Gray. Otra área donde posee aplicación este método es en el campo de sistemas de comunicación, ya que la transición de pasar de un código a otro cambiando únicamente un dígito hace que se presenten menos posibilidades de error y a su vez hace que se facilite la corrección de errores. Esto es importante, ya que dependiendo del medio en el que se transmite, como por ejemplo el aire, puede que este genere mucho ruido y es necesario prevenir y corregir la aparición de errores hasta donde sea posible.²

2. Diagrama y Diseño

Es necesario crear una descripción conductual del contador Gray en Verilog para 5 bits. La secuencia en el código Gray, donde se especifica la cantidad de bits, viene dada en la figura 1. Para seguir esta secuencia con 5 bits, se utilizó un contador en binario y posteriormente se hace la transición de Binario a Gray. Para realizar la transición de Binario a Gray se siguen una serie de pasos. Primeramente el bit más significativo es el mismo en binario y en Gray. Seguidamente comenzando de izquierda a derecha se van sumando cada par adyacente de los bits en código binario para obtener los bits en código Gray. Esta suma no toma en cuenta los acarreos.

Nº Decimal	CODIGO GRAY														
	1 Bit	2 Bit		3 Bit			4 Bit				5 Bit				
	G1	G2	G1	G3	G1		G4		G1		G5			G1	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	0	1	0	0	1	0	0	0	1	0	0	0	0	1
2		1	1	0	1	1	0	0	1	1	0	0	0	1	1
3	3	1	0	0	1	0	0	0	1	0	0	0	0	1	0
4				1	1	0	0	1	1	0	0	0	1	1	0
5				1	1	1	0	1	1	1	0	0	1	1	1
6				1	0	1	0	1	0	1	0	0	1	0	1
7				1	0	0	0	1	0	0	0	0	1	0	0
8							1	1	0	0	0	1	1	0	0
9							1	1	0	1	0	1	1	0	1
10							1	1	1	1	0	1	1	1	1
11							1	1	1	0	0	1	1	1	0
12							1	0	1	0	0	1	0	1	0
13							1	0	1	1	0	1	0	1	1
14							1	0	0	1	0	1	0	0	1
15						15	1	0	0	0	0	1	0	0	0
16											1	1	0	0	0
17											1	1	0	0	1
18											1	1	0	1	1
19											1	1	0	1	0
20											1	1	1	1	0
21											1	1	1	1	1
22											1	1	1	0	1
23											1	1	1	0	0
24											1	0	1	0	0
25											1	0	1	0	1
26											1	0	1	1	1
27											1	0	1	1	0
28											1	0	0	1	0
29											1	0	0	1	1
30											1	0	0	0	1
31										31	1	0	0	0	0

Figura 1: Secuencia de código Gray

3. Plan de Pruebas

Se cuenta con un código estructural brindado por el profesor, por ende se realiza un código conductual, un probador y un testbench. Se espera que el código conductual y estructural posean un comportamiento similar al principio y así corroborar que el diseño conductual funcione, por ende se espera que el diseño conductual posea un comportamiento como el que se muestra en la figura 2. Seguidamente se pretende aumentar el posedge clk mediante un repeat en el probador, y así ir comparando ambas señales mediante el uso de la herramienta gtkwave, hasta que difieran en alguna señal. De esta manera se pretende detectar el error, que se espera aparezca en un tiempo lejano.



Figura 2: Funcionamiento corrector del contador Gray

4. Instrucciones de utilización de la simulación

Para la automatización se crea el siguiente makefile:

```

1  all: iverilog gtkwave
2  iverilog:
3      iverilog -o prueba.vvp BancoPrueba.v    cmos_cells.v
4      vvp prueba.vvp
5
6  gtkwave:
7      gtkwave probador.vcd

```

Para utilizar el makefile primeramente hay que dirigirse en la terminal a la carpeta que contiene estos archivos. Utilizando el comando **make iverilog** se crea el archivo **probador.vcd**. Una vez creado el archivo probador.vcd correspondiente, se utiliza el comando **make gtk-wave** y se abre el programa gtkwave, se seleccionan las señales que se quieren observar y finalmente se presenta la simulación deseada. En caso de utilizar el comando **make all** se realizan juntos todos los comando necesarios.

5. Ejemplos de los resultados:

Primeramente como se observa en la figura 3, se muestra un comportamiento idéntico entre el modelo conductual y estructural como era de esperar, siendo `salida_gray` el modulo sintetizado y `salida_gray_no_error` el modulo conductual, desde el segundo 0 al 77. Seguidamente, recorriendo las gráficas, en el segundo 145 se detecta una diferencia entre ambos módulos, lo que hace referencia a que se encontró el desperfecto en el modelo sintetizado.

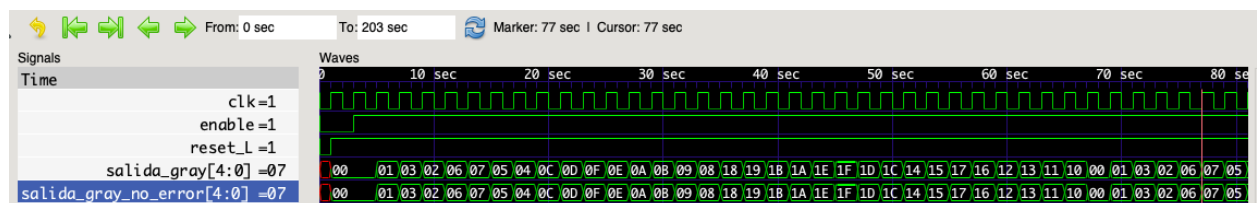


Figura 3: Comportamiento idéntico entre modulo conductual y estructural

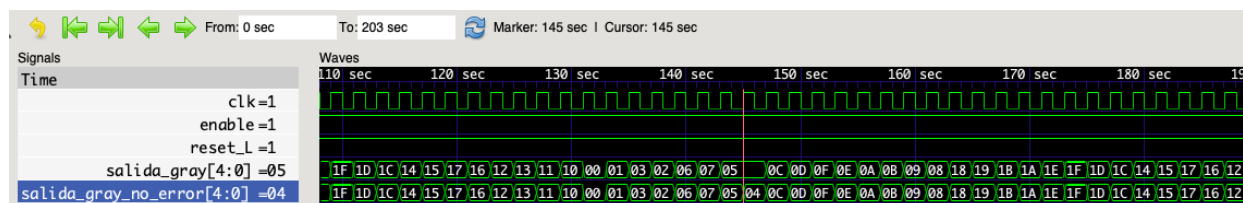


Figura 4: Diferencia entre modulo conductual y estructural

6. Análisis y conclusiones

Analizando la figura 3 se puede afirmar que el diseño conductual funciona correctamente, ya que se comporta idénticamente al diseño estructural, donde se sabe que al principio el diseño estructural funcionaba bien. Cabe mencionar que en el probador se utilizaron las mismas señales de la figura 2, donde el programa funciona de acuerdo a lo esperado. Primeramente se posee un reset en bajo, el cual debería sacar a la salida un 0, como está sucediendo en la figura 3. Seguidamente cuando el reset sube, debido a que el enable queda abajo sigue sacando a la salida de los contadores de Grey 0. Una vez que estén en alto ambos ya se empieza a transmitir la secuencia del contador de Gray, por lo que se puede afirmar que el modulo conductual sirve de manera esperada de acuerdo a las instrucciones solicitadas. Una vez comprobado que el diseño conductual funciona, es posible buscar el error, para ello se tuvo que aumentar el tiempo, ya que al principio ambas gráficas daban resultados idénticos. De la figura 4 se tiene que en el segundo 145 ocurre un error en el modelo estructural, debido a que posee consecutivamente 05 dos veces, saltándose el número 04 que debería ir después de 05.

En conclusión se puede decir que se obtuvo un buen diseño del modelo conductual, ya que a partir de ese diseño fue posible encontrar el desperfecto del modelo sintetizado. También se puede concluir que para los procesos de verificación se debe tener bastante cuidado y hay que ponerle atención a los detalles, debido a que en este caso se presentaba un error que a simple vista no se percibía, sino que se tuvo que examinar por una gran cantidad de tiempo e ir comparando resultado por resultado hasta encontrar alguna diferencia, que puede pasar desapercibida debido que después de pasar por el error, el contador vuelve a funcionar de la manera esperada. Este a pesar de ser un pequeño error, que después de ahí no afecta la secuencia, es incorrecto desde las características del código Gray que cada numero debe diferir de su anterior en un bit, donde el error provoca que el resultado de 05 y el posterior den exactamente igual en el segundo 145. Dependiendo de la aplicación que se le de a este puede ser un grave error, por lo que es necesario aplicar un buen proceso de verificación para darse cuenta que el código conductual no está funcionando correctamente.

7. Distribución del tiempo invertido en la tarea

En total se duró 3 horas con realizando la tarea

- Buscar información: 30 minutos
- Estudiando la información tomó 10 minutos

- Ejecutando lo que decidió hacer y probándolo se tomó 1 hora con 20 minutos.
- Realizar el reporte tomó 1 hora.

Referencias

- [1] Electronica Radical. Código gray. <https://electronicaradical.blogspot.com/2014/08/codigo-gray.html>, 08 2014.
- [2] David Naranjo. ¿qué es el código gray? <https://unicrom.com/codigo-gray/>, 01 2010.