

Trabajo Práctico N° 1

Electrónica III - 2019

Grupo 1:

Farall, Facundo

Gaytan, Joaquín

Kammann, Lucas

Maselli, Carlos

September 4, 2019

1 EJERCICIO 5

Se implementa en Verilog un multiplicador de 2 números de un dígito en código BCD que retorna el resultado como 2 dígitos en formato BCD. Incluye además, un bit de error que será 1 en caso de que haya un error y 0 en otro caso.

1.1 PROCESO DE DISEÑO

En un principio se decidió implementar este sistema a partir de realizar una tabla de verdad y un mapa de Karnaugh. Sin embargo, ese es un punto de vista que si bien parece ser simple, resulta muy engorroso de implementar. Esto se debe a la cantidad de entradas y salidas del sistema, 8 de cada una. Esto requiere resolver 8 mapas, con 8 variables de entrada. Se opta entonces, por implementarlo emulando la manera en que se resuelven multiplicaciones binarias a mano. Se puede observar en 1.1 un ejemplo de lo hablado.

$$\begin{array}{r} 1001 \\ \times 0111 \\ \hline 1001 \\ 1001. \\ 1001.. \\ 0000... \\ \hline 0111111 \end{array} \quad (1.1)$$

1.2 IMPLEMENTACIÓN

En la Figura 1.1 se puede observar la implementación de este proceso con Full-Adders y compuertas AND.

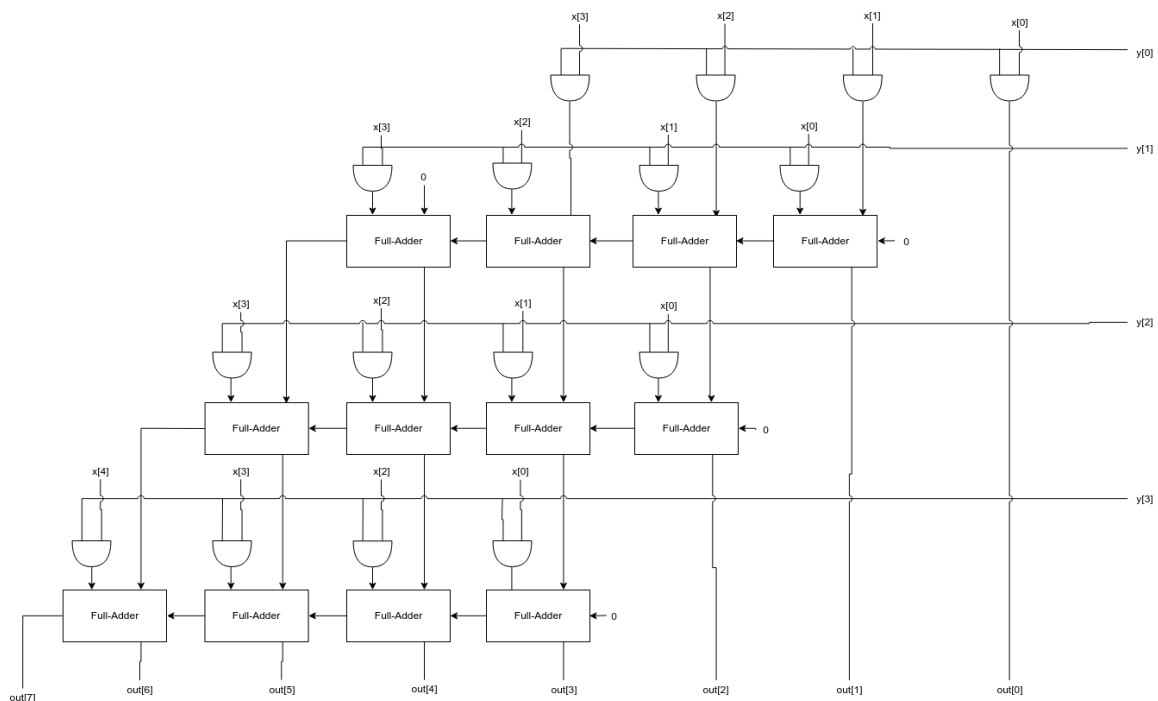


Figure 1.1: Circuito lógico que emula el caso descrito.

Este diagrama representa únicamente la sección de la multiplicación binaria descrita en 1.1, el código está escrito completamente con compuertas lógicas en Verilog. En cambio, tanto como para validar la

información recibida como para codificar los resultados binarios en formato BCD se opta por usar bloques *behavioral*. Se adjunta junto con este informe, el código de Verilog con su correspondiente Makefile y un archivo *run*.

1.3 TEST-BENCH

Es posible ejecutar un *Test-bench* para verificar que el módulo funciona correctamente, con algunos ejemplos de las posibles entradas al sistema. El mismo puede ser ejecutado por medio de los comandos:

```
user@computer: path/to/EJ_5/folder$ make
user@computer: path/to/EJ_5/folder$ ./run
```
