Universidad de Costa Rica

Sede Rodrigo Facio
Facultad de Ingeniería
Escuela de Ingeniería Eléctrica

IE-0523 Circuitos Digitales II

Tarea 4

Prof. Jorge Soto

Andrés Alvarado Velázquez

B30316

22 de abril
I Ciclo 2018.

1. Resumen

Se programa por comportamiento estructural la tarea 2. Para esto se tuvo que pasar la logica de programación de conductual a estructural. Es decir pasar de codigo .entendible"(if, else, case, etc) a pura logica de compuertas. Esta tarea se combinó con la tarea 3 la cual se escribió una biblioteca con comportamientos de compuertas logicas y retardo en sus conmutaciones.
Finalmente mediante el uso de la herramienta de gtkwave se comparó el output de ambas tareas, la conductural y la estructural, para confirmar que se haya hecho bien y poder observar la diferencia con los retardos.

2. Descripción Arquitectónica

Se pretende programar utilizando solo compuertas logicas, por lo que en su gran mayoría se utilizaron multiplexores de 2 entradas. Tambien se utilizaron nand, nor y not.

Con los multiplexores se pudieron hacer los casos para cada combinación de DIR y MODE y con logica combinacional se pudo resolver el comportamiento de Q.

3. Plan de pruebas

Pruebas de comportamiento

- Establecer A=0, B=0.
- Enviar flancos activos en CLK, para cargar el valor de C.
- Establecer A=0, B=1.
- Enviar flancos activos en CLK, para cargar el valor de C.
- Establecer A=1, B=0.
- Enviar flancos activos en CLK, para cargar el valor de C.
- Establecer A=1, B=1.
- Enviar flancos activos en CLK, para cargar el valor de C.

Conteo de actividad

- Establecer LE=0, Contador=0.
- Establecer cada Contador=0 haciendo uso de un for.
- Establecer un ciclo finito de 5000 veces.
- Establecer A[3]=0;
- Esperar un delay y establecer A[0]=0, B[0]=0.
- Esperar un delay y establecer A[0]=1, B[0]=1.
- Esperar un delay y establecer A[2]=0, B[2]=0.

- Esperar un delay y establecer A[2]=1, B[2]=1.
- Esperar un delay y establecer A[3]=1, con este último el conteo de la actividad será impar.

4. Instrucciones de utilización de la simulación

Por cuestiones de orden se posee dos archivos probadores, uno para el comportamiento lógico y temporal de los componentes y el otro para el conteo de la actividad de cambio de estado de estas. Para correr las respuestas del comportamiento debe ingresar los siguientes comandos:

```
make
make gtk
```

Los anteriores comandos imprimirán el comportamiento lógico en la terminal. Considerar que estas salidas presentan un sesgo al trabajarse con delays, se trató a prueba y error tratar de mostrar los cambios de una manera evidente y limpia.

5. Ejemplos de los resultados

La siguiente figura muestra las señales de salida con respecto a D y al Mode. Se puede observar como se comportan de la misma manera con los mismas entradas por lo que se puede afirmar su funcionalidad.

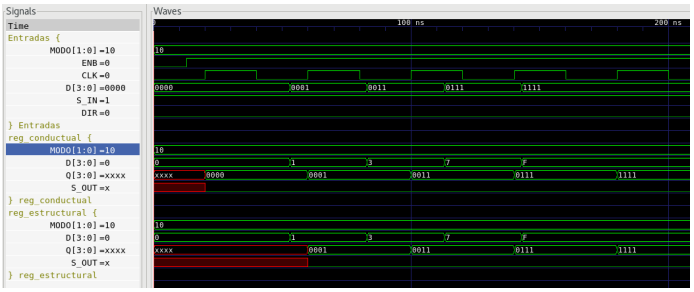

Figura 1: Comparación entre estructural y conductual

La imagen previa enseña como se ve cuando se usa el siguiente comando: *make gtk*.

6. Tiempo

Actividad	Tiempo	Descripción
Buscar información y estudiarla	1hora 20 minutos	Esto comprende entender la programación de comp
Ejecución	1 hora con 40 minutos	Programación Estructural
Ejecución	2 hora con 25 minutos	Confección de pruebas y verificación de las mismas
Reporte y presentación	2 horas con 35 minutos	

Cuadro 1: Tabla de tiempo.

7. Conclusiones y recomendaciones

- Se logro migrar un código de conductual a estructural con éxito
- Se vio la utilidad de tener una biblioteca de las compuertas propias.
- Recomendado para la próxima programación de forma conductual hacer cases en vez de tantos if's ya que simplifica la migración y entendimiento del código.