

Banco de componentes estructurales

Gabriel Fernando Araya Mora B80525

Escuela de Ingeniería Eléctrica, Universidad de Costa Rica

Circuitos Digitales II (IE-0523)

1. Tiempo de ejecución de la tarea:

- **Buscar información:** Siempre antes de empezar cualquier trabajo es importante tomarse un tiempo prudente para buscar información e investigar acerca del tema. Esto es especialmente cierto cuando se trata de programar. Para la búsqueda de información, revisé la documentación de verilog acerca de multiplexores y flip-flops además de las descripciones estructurales de las compuertas solicitadas. Revisé el proyecto de verilog del curso anterior para ver si se podía reusar código de alguna u otra forma. Aproximadamente 1 hora me tomó la búsqueda de información.
- **Usando la información:** Como se pide la descripción estructural, se debe trabajar desde el nivel de lógica más bajo para armar los componentes mas complejos usando las compuertas después. La construcción de las compuertas fue algo relativamente sencillo de hacer; sin embargo, el checker y el contador fueron bastante complicados de implementar. Este proceso tomó aproximadamente 4 horas.
- **Elaboración del reporte:** La elaboración del reporte me tomó aproximadamente una hora.
- **Total:** De lo anterior se saca que el tiempo total para la realización de la tarea es de 6h.

2. Descripción arquitectónica o diagrama del circuito.

2.1. AND

Para la compuerta AND se toman:

- $C_L = 15pF$
- $T_A = -40^{\circ}C$ a $85^{\circ}C$
- $t_{PD} \rightarrow MIN = 1ns, Typ = 4.5ns, MAX = 8ns$
- $C_{pd} = 5pF$
- Precio: \$0.092 por unidad.

6.6 Switching Characteristics, $V_{CC} = 3.3 \text{ V} \pm 0.3 \text{ V}$

over recommended operating free-air temperature range, $V_{CC} = 3.3 \text{ V} \pm 0.3 \text{ V}$ (unless otherwise noted) (see Figure 2)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	OUTPUT CAPACITANCE	T_A	MIN	TYP	MAX	UNIT
t_{PD}	A or B	Y	$C_L = 15 \text{ pF}$	$T_A = 25^\circ\text{C}$		3.6	7	ns
				$T_A = -40^\circ\text{C to } +85^\circ\text{C}$	1		8	
				$T_A = -55^\circ\text{C to } +125^\circ\text{C}$	1		8.5	
	A or B	Y	$C_L = 50 \text{ pF}$	$T_A = 25^\circ\text{C}$		6.5	11	ns
				$T_A = -40^\circ\text{C to } +85^\circ\text{C}$	1.5		12	
				$T_A = -55^\circ\text{C to } +125^\circ\text{C}$	1.5		12.5	

Figura 1: Tiempos de retardo para la compuerta AND. (Creación Propia)

6.8 Operating Characteristics

$V_{CC} = 5 \text{ V}$, $T_A = 25^\circ\text{C}$

PARAMETER	TEST CONDITIONS	TYP	UNIT
C_{pd} Power dissipation capacitance	No load, $f = 1 \text{ MHz}$	5	pF

Figura 2: Consumo de la compuerta AND. (Creación Propia)

2.2. OR

Para la compuerta OR se toman:

- $C_L = 50 \text{ pF}$
- $T_A = 25^\circ\text{C}$
- $t_{PLH} = t_{PHL} \rightarrow MIN = NaN, Typ = 8 \text{ ns}, MAX = 11.4 \text{ ns}$
- $C_{pd} = 14 \text{ pF}$
- Precio: \$0.215 por unidad.

switching characteristics over recommended operating free-air temperature range, $V_{CC} = 3.3 \text{ V} \pm 0.3 \text{ V}$ (unless otherwise noted) (see Figure 1)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	LOAD CAPACITANCE	$T_A = 25^\circ\text{C}$			MIN	MAX	UNIT
				MIN	TYP	MAX			
t_{PLH}	A or B	Y	$C_L = 50 \text{ pF}$		8	11.4	1	13	ns
t_{PHL}					8	11.4	1	13	

Figura 3: Tiempos de retardo para la compuerta OR. (Creación Propia)

operating characteristics, $V_{CC} = 5\text{ V}$, $T_A = 25^\circ\text{C}$

PARAMETER		TEST CONDITIONS	TYP	UNIT
C_{pd}	Power dissipation capacitance	No load, $f = 1\text{ MHz}$	14	pF

Figura 4: Consumo de la compuerta OR. (Creación Propia)

2.3. NOT

Para la compuerta *NOT* se toman:

- $C_L = 15\text{ pF}$
- $T_A = 25^\circ\text{C}$
- $t_{PD} \rightarrow MIN = 1.5, Typ = 3.1\text{ ns}, MAX = 4.6\text{ ns}$
- $C_{pd} = 22\text{ pF}$
- Precio: \$0.187 por unidad.

6.6 Switching Characteristics, $C_L = 15\text{ pF}$

over recommended operating free-air temperature range (unless otherwise noted) (see Figure 3)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	$V_{CC} = 1.8\text{ V}$ $\pm 0.15\text{ V}$		$V_{CC} = 2.5\text{ V}$ $\pm 0.2\text{ V}$		$V_{CC} = 3.3\text{ V}$ $\pm 0.3\text{ V}$		$V_{CC} = 5\text{ V}$ $\pm 0.5\text{ V}$		UNIT
			MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
t_{pd}	A	Y	2.8	9.9	1.6	5.5	1.5	4.6	0.9	4.4	ns

Figura 5: Tiempos de retardo para la compuerta NOT. (Creación Propia)

6.8 Operating Characteristics

$T_A = 25^\circ\text{C}$

PARAMETER	TEST CONDITIONS	$V_{CC} = 1.8\text{ V}$	$V_{CC} = 2.5\text{ V}$	$V_{CC} = 3.3\text{ V}$	$V_{CC} = 5\text{ V}$	UNIT
		TYP	TYP	TYP	TYP	
C_{pd} Power dissipation capacitance	$f = 10\text{ MHz}$	20	21	22	25	pF

Figura 6: Consumo de la compuerta NOT. (Creación Propia)

2.4. FLIP-FLOP

Para el *Flip - Flop* se toman:

- $C_L = 50\text{ pF}$

- $T_A = 25^\circ C$
- $t_{PD} \rightarrow MIN = 8ns, Typ = 19ns, MAX = 42ns$
- $C_{pd} = 10pF$
- Precio: \$0.187 por unidad.

6.6 Switching Characteristics

$C_L = 50\text{ pF}$; over operating free-air temperature range; typical values measured at $T_A = 25^\circ C$ (unless otherwise noted). See [Parameter Measurement Information](#).

PARAMETER		FROM (INPUT)	TO (OUTPUT)	V_{CC}	MIN	TYP	MAX	UNIT
f_{max}	Max switching frequency			2 V	18	31		MHz
				4.5 V	45	95		
				6 V	65	105		
t_{pd}	Propagation delay	\overline{PRE} or \overline{CLR}	Q or \overline{Q}	2 V		19	42	ns
				4.5 V		8	19	
				6 V		7	15	
		CLK	Q or \overline{Q}	2 V		19	42	ns
				4.5 V		8	19	
				6 V		7	15	

Figura 7: Tiempos de retardo para la compuerta FLIP-FLOP. (Creación Propia)

C_i	Input capacitance		2 V to 6 V	5	pF
C_{pd}	Power dissipation capacitance per gate	No load	2 V to 6 V	10	pF

Figura 8: Consumo del FLIP-FLOP. (Creación Propia)

2.5. MUX

Como ya se tienen construidas las compuertas, para simplificar un poco el proceso, se construye el multiplexor estructural usando como bloques de construcción las compuertas. Resulta que el multiplexor es un circuito realmente útil, pero realmente simple, ya que consta de 2 compuertas *AND* las cuales están encargadas de tirar a la salida lo que venga en los datos si el selector así se lo permite, por último en la salida se tiene una compuerta *OR*, la cual se encarga de unir las salidas de las *AND*. Es importante destacar que el selector es una única señal; sin embargo, a una de las *AND* le entre el selector negado, esto para asegurar que solo pase una de las *AND*.

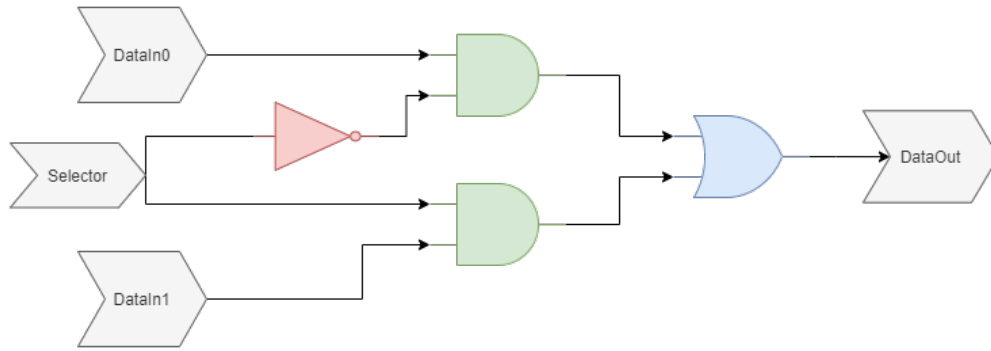


Figura 9: Esquemático de Diseño para el mux de 1 bit. (Creación Propia)

2.6. MUX 2 Bits

Para la construcción del mux de dos bits, se colocan dos muxes de un bit en paralelo. La lógica detrás de esto es que un bus de dos bits es literalmente que son dos cables que se tratan como uno, y por ende se pueden separar en « hilos » los cuales se tratan como unitarios y se pueden comparar en cada multiplexor, y al final unirlos en un nuevos bus de 2 bits.

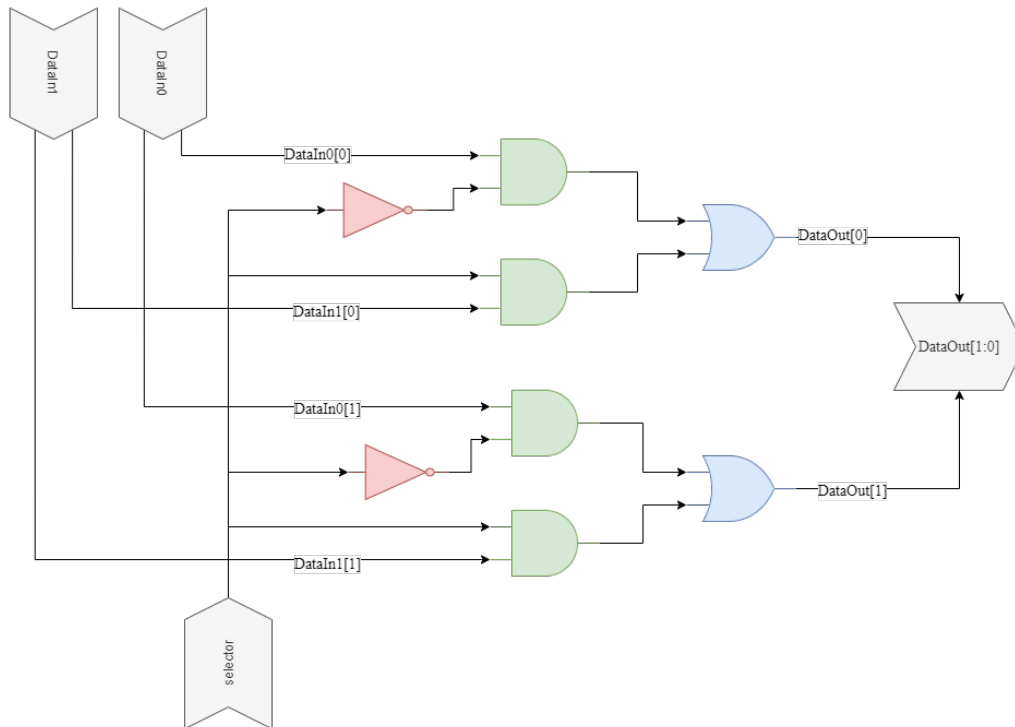


Figura 10: Esquemático de Diseño para el mux de 2 bit. (Creación Propia)

2.7. Tarea #2 Estructural:

Este caso es muy sencillo ya que se tienen el resto de compuertas, basta con hacer las instancias respectivas y conectarlas de forma correcta. El Flip-Flop implementado es de dos bit y para el cual se siguió la misma lógica que en el multiplexor, por ende se colocaron dos flops de un bit en paralelo. Este módulo de tarea #2 estructural consta de dos instancias de mux de 2 bits, donde la primera escoge entre las entradas de datos según el selector, y el segundo escoge entre un cero de reset o el dato del mux anterior según lo indique el reset. Por último se tiene la entrada del flop de dos bits.

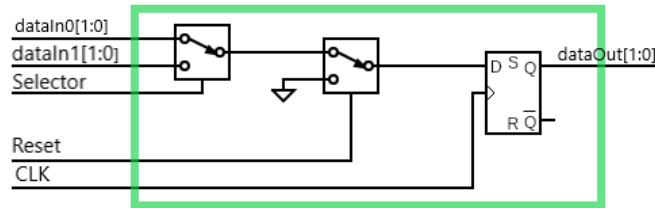


Figura 11: Esquemático de Diseño para la tarea #2 ahora de forma estructural. (Creación Propia)

3. Plan de pruebas

Todos los cambios en las señales de entrada se hicieron con base en el archivo mostrado en la tarea #2 dada por el profesor, esto debido a que hace muchos casos posibles y combinaciones de los mismos, lo cual asegura que las descripciones funcionen de manera correcta.

1. En el probador 1, se hacen las pruebas respectivas a para cada una de las compuertas individuales, el flop y el multiplexor. Es importante notar que las entradas marcadas como *dataIn0* y *dataIn1* son las que se usan como entradas para el multiplexor, mientras que las entradas rotuladas como *dataInA* y *dataInB* son utilizadas para probar las compuertas lógicas.
2. Para el probador 2, se hacen las pruebas de los componentes complejos de 2 bits, como el mux y el flop. De igual forma las entradas rotuladas como *dataIn0* y *dataIn1* son usadas para probar el multiplexor, mientras que para probar el flop se usa *dataIn1*.
3. Por último el probador3 es utilizado para comparar el módulo diseñado bajo una descripción conductual y el estructural. Además aquí se muestran los contadores tanto del estructural y el conductual, además se ve el correcto funcionamiento del checker.

4. Instrucciones para ejecutar el programa.

Para ejecutar el programa, se tienen dos opciones validas. Estando en el directorio **src** y abriendo una terminal en esta dirección basta con escribir **make**; sin embargo, esto lo hará de forma secuencial, es decir primero se abrirá el probador1, al cerrar el Gtk wave se abre el siguiente probador y así hasta terminar con los tres.

La segunda opción trata de decirle al make file cual probador se quiere ejecutar por lo tanto se hace: **make probador1** para el primer probador y así con los otros. Es necesario recalcar que independientemente de el método seleccionado para correr la tarea, se debe estar dentro del directorio **src**

```
build: probador1 probador2 probador3
      echo fin

probador1:
      iverilog bancoPrueba1.v -T typ
      ./a.out
      rm a.out
      gtkwave probador1.vcd

probador2:
      iverilog bancoPrueba2.v -T typ
      ./a.out
      rm a.out
      gtkwave probador2.vcd

probador3:
      iverilog bancoPrueba3.v -T typ
      ./a.out
      rm a.out
      gtkwave probador3.vcd
```

Después de haber usado el comando bash anteriormente mencionado se abrirá el programa con gtkwave abriendo el archivo .vcd con la simulación de la tarea. Lo único que restaría hacer es agregar las señales al gtkwave.

5. Ejemplos de los Resultados.

Primero se muestran los resultados para las compuertas individuales. En la siguiente imagen se puede apreciar el correcto comportamiento para cada una de las compuertas solicitadas en la biblioteca; sin embargo, los cambios en las salidas no se dan exactamente en el flanco positivo del pulso, esto debido a que hay un retardo que está dado por los valores anteriormente mencionados.

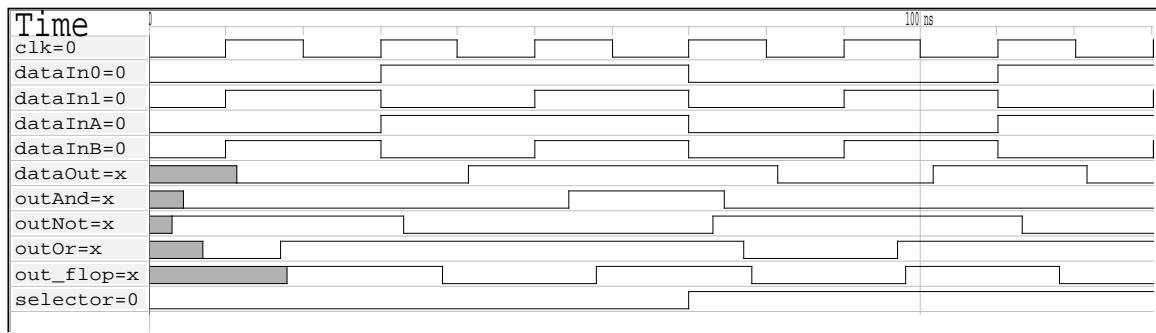


Figura 12: Probador 1 (Creación Propia)

Como los componentes más complejos están contruidos con las compuertas anteriores los retardos se heredan y se suman cuando sea necesario, este comportamiento se puede o observar en la siguiente figura, en donde los retardos en la señal de salida del mux y el flop son mayores que en las compuertas individuales.

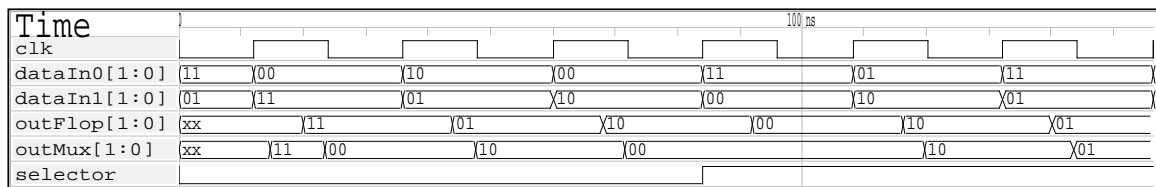


Figura 13: Probador 2 (Creación Propia)

Por último se tiene el módulo con doble mux y el flop. Se puede ver claramente que tanto la señal de la descripción conductual como la descripción estructural se comportan igual, solo que la estructural al tener retardos, esto se muestra como un desfase entre ambas señales. para comprobar que ambas señales son equivalentes de flanco a flanco, se espera que el checker devuelva un 1 si ambas son equivalentes, pero si la señal de la descripción estructural llega después del siguiente flanco, el checker devuelve cero, indicando que no son equivalentes. Para el contador de transiciones positivas, se cuentan usando un posedge y se cuentan los bits individuales para fanalmente sumarlos.

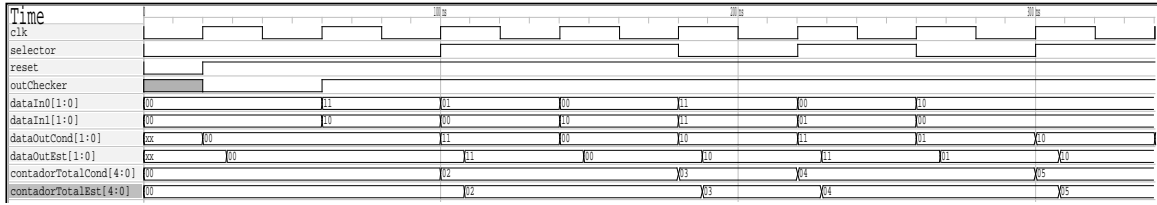


Figura 14: Probador 3 (Creación Propia)

6. Análisis y conclusiones:

En las siguientes 3 figuras se muestra como al variar la frecuencia de reloj la salida que arroja la descripción estructural, se atrasa cada vez hasta que no le da tiempo de llegar antes del siguiente flanco positivo de reloj y por ende no serviría de manera deseada. Para calcular la frecuencia teórica máxima de operación:

$$T = 4.5ns + 8ns + 3.1ns + 8ns = 24ns$$

$$f_{max} = \frac{1}{T} = \frac{1}{24 \times 10^{-9}} = 42Mhz$$

En la siguiente figura se ve como a una frecuencia más baja de la nominal máxima, la descripción estructural funciona bien.

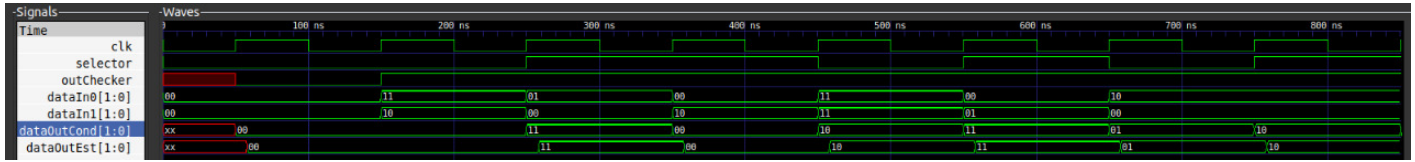


Figura 15: Frecuencia de 10MHz (Creación Propia)

En la siguiente figura se ve como a una frecuencia más alta de la nominal máxima, la descripción estructural funciona mal y esto se ve ya que se pierde información, además de que el checker dice que hay señales que no llegan a tiempo al flanco positivo del reloj.

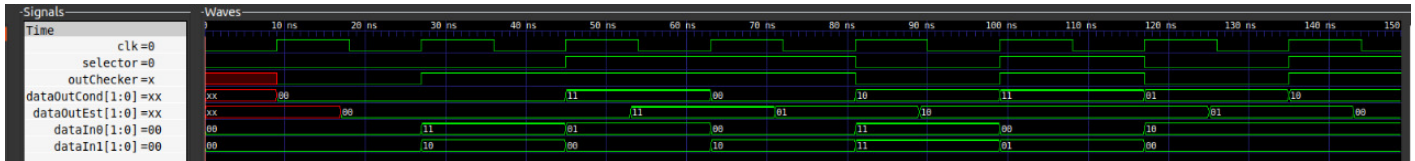


Figura 16: Frecuencia de 56MHz(Creación Propia)

Por último en la siguiente figura se muestra cómo entre más alta sea la frecuencia de reloj, más se atrasa la salida de la descripción estructural, el checker aquí muestra que ambas descripciones nunca son iguales.

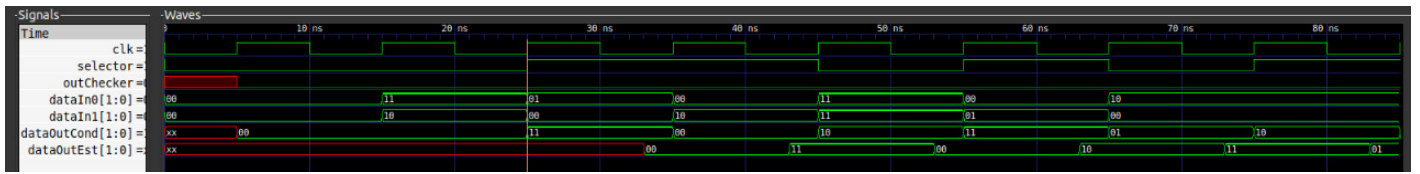


Figura 17: frecuencia de 100MHz (Creación Propia)