

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

Tarea #5

1. Diagrama

Primeramente se presenta toda la información acerca cada compuerta de la biblioteca. Debido a que se realizó la descripción estructural mediante yosys, esta automáticamente seleccionó las compuertas NAND y NOR en comparación a las de la tarea anterior de AND y OR.

NAND: Se buscó el modelo SN74LVC1G00, el cual posee un costo de \$0.063 por unidad. En la figura 1 se muestran las características de temporización del componente, donde se selecciona una tensión de 1.8V, con temperatura que va desde -40 hasta 125°. En la figura 2 se observa el consumo de potencia.

7.8 Switching Characteristics, -40°C to 125°C

over recommended operating free-air temperature range, $C_L = 30$ pF or 50 pF (unless otherwise noted) (see Figure 4)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	RECOMMENDED -40°C to 125°C								UNIT
			V _{CC} = 1.8 V ± 0.15 V		V _{CC} = 2.5 V ± 0.2 V		V _{CC} = 3.3 V ± 0.3 V		V _{CC} = 5 V ± 0.5 V		
			MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
t _{pd}	A or B	Y	3.1	9.7	1.3	5.8	1	5	1	4.3	ns
t _{pd}	A	Y	2	6.4	1	4.2	0.7	3.3	0.7	3.1	ns

Figura 1: Características de temporización de la NAND

7.9 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	$f = 10\text{ MHz}$	22	22	23	25	pF

Figura 2: Consumo de potencia de la NAND

NOR: Se buscó el modelo 74LVC1G02, el cual posee un costo de \$1.653 por unidad. En la figura 3 se muestran las características de temporización del componente, donde se selecciona una tensión de 1.8V, a temperatura que va desde -40 hasta 125° y también se observa el consumo de potencia.

Table 8. Dynamic characteristics

Voltages are referenced to GND (ground = 0 V); for load circuit see Fig. 9.

Symbol	Parameter	Conditions	-40 °C to +85 °C			-40 °C to +125 °C		Unit
			Min	Typ[1]	Max	Min	Max	
t_{pd}	propagation delay	A, B to Y; see Fig. 8 [2]						
		$V_{CC} = 1.65 \text{ V to } 1.95 \text{ V}$	1.0	3.2	8.0	1.0	10.5	ns
		$V_{CC} = 2.3 \text{ V to } 2.7 \text{ V}$	0.5	2.2	5.5	0.5	7.0	ns
		$V_{CC} = 2.7 \text{ V}$	0.5	2.5	5.5	0.5	7.0	ns
		$V_{CC} = 3.0 \text{ V to } 3.6 \text{ V}$	0.5	2.1	4.5	0.5	6.0	ns
		$V_{CC} = 4.5 \text{ V to } 5.5 \text{ V}$	0.5	1.7	4.0	0.5	5.5	ns
C_{PD}	power dissipation capacitance	$V_I = \text{GND to } V_{CC};$ $V_{CC} = 3.3 \text{ V}$ [3]	-	14	-	-	-	pF

Figura 3: Características de temporización y consumo de potencia de la NOR

Inversor: Se buscó el modelo SN74LVC1G04, el cual posee un costo de \$0.063 por unidad. En la figura 4 se muestran las características de temporización del componente, donde se selecciona una tensión de 1.8V, con temperatura que va desde -40 hasta 85 grados y $C_L = 30pF$ o $C_l = 50pF$, adaptandose a las 2 compuertas anteriores. En la figura 5 se observa el consumo de potencia.

7.7 Switching Characteristics, $C_L = 30 \text{ pF}$ or 50 pF , -40°C to 85°C

over recommended operating free-air temperature range, $C_L = 30 \text{ pF}$ or 50 pF (unless otherwise noted)
(see Figure 4)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	-40°C to 85°C								UNIT
			V _{CC} = 1.8 V ± 0.15 V		V _{CC} = 2.5 V ± 0.2 V		V _{CC} = 3.3 V ± 0.3 V		V _{CC} = 5 V ± 0.5 V		
			MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
t _{pd}	A	Y	3	7.5	1.4	5.2	1	4.2	1	3.7	ns

Figura 4: Características de temporización del inversor

7.10 Operating Characteristics

over operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS	V _{CC} = 1.8 V	V _{CC} = 2.5 V	V _{CC} = 3.3 V	V _{CC} = 5.0 V	UNIT	
		TYP	TYP	TYP	TYP		
C _{od}	Power dissipation capacitance	f = 10 MHz	16	18	18	20	pF

Figura 5: Consumo de potencia del inversor

Flip Floop: Se buscó el modelo SN74AUP2G79, el cual posee un costo de \$0.273 por unidad. En la figura 6 se muestran las características de temporización del componente, donde se selecciona una tensión de 1.8V, con $C_L = 30pF$. En la figura 7 se observa el consumo de potencia.

SWITCHING CHARACTERISTICS

over recommended operating free-air temperature range, $C_L = 30 \text{ pF}$ (unless otherwise noted) (see [Figure 3](#) and [Figure 4](#))

PARAMETER	FROM (INPUT)	TO (OUTPUT)	V_{CC}	$T_A = 25^\circ\text{C}$			$T_A = -40^\circ\text{C}$ to 85°C		UNIT
				MIN	TYP	MAX	MIN	MAX	
f_{\max}			0.8 V		24		20		MHz
			$1.2 \text{ V} \pm 0.1 \text{ V}$		72		80		
			$1.5 \text{ V} \pm 0.1 \text{ V}$		100		100		
			$1.8 \text{ V} \pm 0.15 \text{ V}$		127		140		
			$2.5 \text{ V} \pm 0.2 \text{ V}$		185		210		
			$3.3 \text{ V} \pm 0.3 \text{ V}$		266		260		
t_{pd}	CLK	Q	0.8 V		27.2				ns
			$1.2 \text{ V} \pm 0.1 \text{ V}$	7	11.5	17.3	5.9	24	
			$1.5 \text{ V} \pm 0.1 \text{ V}$	5.7	8.3	13.3	4.6	15.9	
			$1.8 \text{ V} \pm 0.15 \text{ V}$	4.7	6.7	11.3	3.8	13	
			$2.5 \text{ V} \pm 0.2 \text{ V}$	3.7	4.9	7.8	2.9	9	
			$3.3 \text{ V} \pm 0.3 \text{ V}$	3.2	4.1	6.3	2.6	7.2	

Figura 6: Características de temporización del flip flop

OPERATING CHARACTERISTICS

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

PARAMETER		TEST CONDITIONS	V_{CC}	TYP	UNIT
C_{pd}	Power dissipation capacitance	$f = 10 \text{ MHz}$	0.8 V	2.5	pF
			$1.2 \text{ V} \pm 0.1 \text{ V}$	2.5	
			$1.5 \text{ V} \pm 0.1 \text{ V}$	2.5	
			$1.8 \text{ V} \pm 0.15 \text{ V}$	2.5	
			$2.5 \text{ V} \pm 0.2 \text{ V}$	3	
			$3.3 \text{ V} \pm 0.3 \text{ V}$	3	

Figura 7: Consumo de potencia del flip flop

Cabe mencionar que debido a que las hojas de fabricante no dan un valor típico, se saca la media entre el mínimo y el máximo para obtener este. En la figura 8 se muestra el diagrama del multiplexor estructural utilizado anteriormente en la tarea 2. El cual posee 4 entradas, las cuales son data_in0, data_in1, reset_L y clk, además de una salida data_out. EL diagrama utiliza 2 multiplexores de 2:1 con dos entradas y una salida de datos de 2 bits y un flip-flop tipo D.

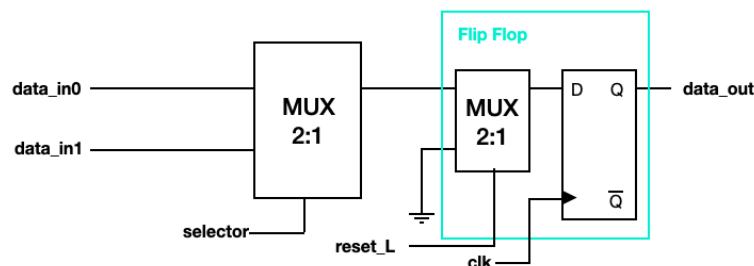


Figura 8: Diagrama mux estructural

2. Plan de Pruebas

Para realizar las pruebas se modificó el probador y el Banco de pruebas de la tarea 3 y 4. Esta modificación consistió en sustituir el archivo de descripción estructural realizado en la tarea 3 y 4, por el archivo sintetizado que se obtiene a partir de yosys. Las pruebas a realizar consisten en utilizar el checker creado en la tarea 3 y 4 para corroborar que el diseño sintetizado(nueva descripción estructural) y el diseño conductual sean iguales, además de confirmarlo de manera gráfica con gtkwave. Entre las verificaciones se deben asegurar que se cumplan las condiciones de que en reset sea 0, se posea una señal de salida de 00. De que en caso de que reset sea 1, dependa del selector(0=data_in0 y 1=data_in1) y además comprobar que todos los cambios se realicen únicamente cuando haya un flanco creciente en la señal del clock(reloj).

3. Instrucciones de utilización de la simulación

Para la automatización se crea el siguiente makefile:

```

1  yosys:
2      yosys -s script.ys
3
4  sed:
5      sed -i '5s/mux_desc_conductual/sintetizado/' sintetizado.v
6
7  iverilog:
8      iverilog -o tarea5 BancoPrueba_tarea5.v cmos_cells.v
9      vvp tarea5
10
11 gtkwave:
12     gtkwave mux_tarea5.vcd
13
14 all: yosys sed iverilog gtkwave

```

Para utilizar el makefile primeramente hay que dirigirse en la terminal a la carpeta que contiene estos archivos. Si se ejecuta el comando **make yosys**, este va a abrir el programa yosys y dentro de este va a escribir el script.ys y por ende va a crear el archivo sintetizado.v. Si se ejecuta el comando **make sed**, este modifica el nombre del modulo del archivo sintetizado, lo cambia de /mux_desc_conductual a sintetizado. Si se ejecuta el comando **make iverilog** este va a crear el archivo .vcd correspondiente. Si se ejecuta el comando **make gtkwave** se abre dicho programa y se puede ver de manera gráfica el comportamiento de las señales. Si se ejecuta el comando **make all** suceden todos los comandos anteriormente mencionados juntos.

4. Ejemplos de los resultados:

Como medio de comprobación de resultados, se muestran las diversas simulaciones en caso de la síntesis con componentes internos, con los componentes de la biblioteca cmos_cells, además de la prueba modificando los componentes de la biblioteca cmos_cells agregando los retardos y modificando su frecuencia por encima de la máxima.

En la figura 9 se muestra la simulación con síntesis con componentes internos. Por medio del comando de iverilog, donde se comprueba que el modelo conductual y el archivo sintetizado poseen el mismo comportamiento debido a que el checker no señala ninguna diferencia entre ambas señales. En la figura 10 se puede comprobar de manera gráfica mediante el gtkwave.

```
alberto@alberto-VirtualBox:~/Descargas/Digitales2/tarea5$ make iverilog
iverilog -o tarea5 BancoPrueba_tarea5.v
vvp tarea5
VCD info: dumpfile mux_Tarea5.vcd opened for output.
```

	clk,	out,	reset,	selector,	in0,	in1
0	0	xx		xx	0	000
13	1	00		00	0	000
27	0	00		00	0	000
40	1	00		00	0	112
53	0	00		00	0	112
67	1	11		11	1	010
80	0	11		11	1	010
93	1	00		00	1	002
106	0	00		00	1	002
120	1	10		10	0	113
133	0	10		10	0	113
146	1	11		11	1	001
160	0	11		11	1	001
173	1	01		01	0	100
186	0	01		01	0	100
200	1	10		10	1	100
213	0	10		10	1	100
226	1	00		00	1	100

Figura 9: Ejemplo iverilog con síntesis con componentes internos

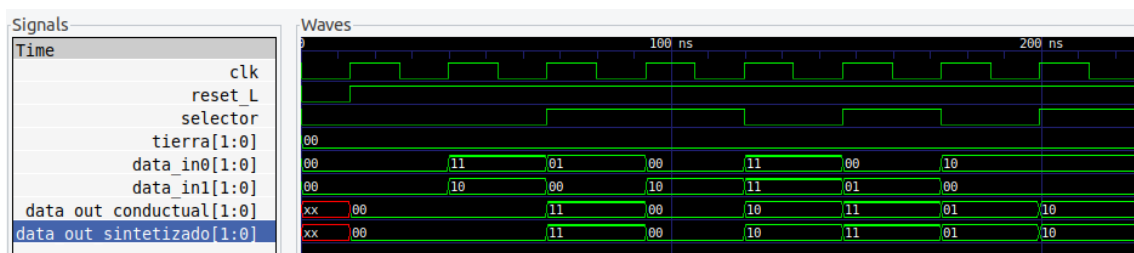


Figura 10: Ejemplo gtkwave con síntesis con componentes internos

En la figura 11 se muestra la simulación con síntesis con componentes de la biblioteca cmos_cells. Por medio del comando de iverilog, donde se comprueba que el modelo conductual y el archivo sintetizado poseen el mismo comportamiento debido a que el checker no señala ninguna diferencia entre ambas señales. En la figura 12 se puede comprobar de manera gráfica mediante el gtkwave.

```

alberto@alberto-VirtualBox:~/Descargas/Digitales2/tarea5$ make all
iverilog -o tarea5 BancoPrueba_tarea5.v cmos_cells.v
vvp tarea5
VCD info: dumpfile mux_tarea5.vcd opened for output.
      clk,  out,  reset,  selector,  in0,  in1
      0    0    xx      xx      0    0    000
     13    1    00      00      0    1    000
     27    0    00      00      0    1    000
     40    1    00      00      0    1    112
     53    0    00      00      0    1    112
     67    1    11      11      1    1    010
     80    0    11      11      1    1    010
     93    1    00      00      1    1    002
    106    0    00      00      1    1    002
    120    1    10      10      0    1    113
    133    0    10      10      0    1    113
    146    1    11      11      1    1    001
    160    0    11      11      1    1    001
    173    1    01      01      0    1    100
    186    0    01      01      0    1    100
    200    1    10      10      1    1    100
    213    0    10      10      1    1    100
    226    1    00      00      1    1    100
gtkwave mux_tarea5.vcd

```

Figura 11: Ejemplo iverilog con síntesis con componentes de la biblioteca cmos_cells

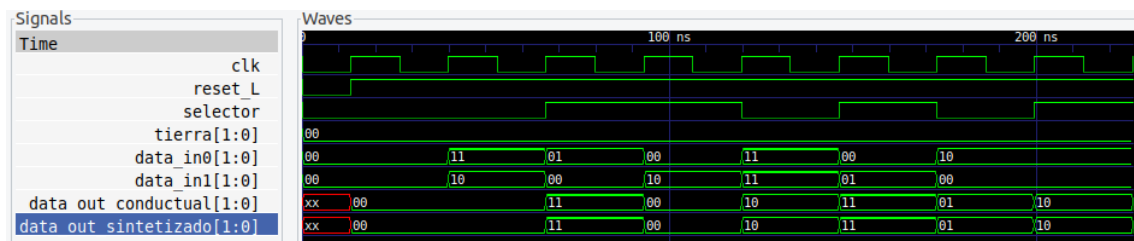


Figura 12: Ejemplo gtkwave con síntesis con componentes de la biblioteca cmos_cells

En la figura 13 se muestra la simulación con síntesis modificando los componentes de la biblioteca cmos_cells para que estos posean retardo. Por medio del comando de iverilog, donde se comprueba que el modelo conductual y el archivo sintetizado poseen el mismo comportamiento debido a que el checker no señala ninguna diferencia entre ambas señales. En la figura 14 se puede comprobar de manera gráfica mediante el gtkwave.

```

MacBook-Air-de-Alberto:tarea5 Beto$ make iverilog
iverilog -o tarea5 BancoPrueba_tarea5.v cmos_cells.v
cmos_cells.v:11: warning: choosing typ expression.
cmos_cells.v:11: warning: choosing typ expression.
cmos_cells.v:17: warning: choosing typ expression.
cmos_cells.v:17: warning: choosing typ expression.
cmos_cells.v:23: warning: choosing typ expression.
cmos_cells.v:23: warning: choosing typ expression.
vvp tarea5
VCD info: dumpfile mux_tarea5.vcd opened for output.

```

	clk,	out,	reset,	selector,	in0,	in1
0	0	xx	xx	0	0	000
13	1	00	00	0	1	000
26	0	00	00	0	1	000
39	1	00	00	0	1	112
52	0	00	00	0	1	112
65	1	11	11	1	1	010
78	0	11	11	1	1	010
91	1	00	00	1	1	002
104	0	00	00	1	1	002
117	1	10	10	0	1	113
130	0	10	10	0	1	113
143	1	11	11	1	1	001
156	0	11	11	1	1	001
169	1	01	01	0	1	100
182	0	01	01	0	1	100
195	1	10	10	1	1	100
208	0	10	10	1	1	100
221	1	00	00	1	1	100

Figura 13: Ejemplo iverilog con síntesis con componentes de la biblioteca cmos_cells con retraso

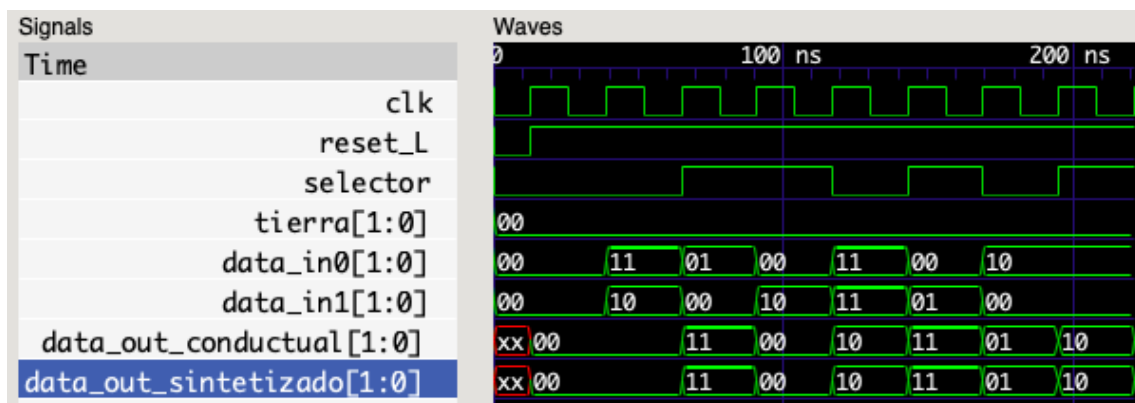


Figura 14: Ejemplo gtkwave con síntesis con componentes de la biblioteca cmos_cells en retardo

En la figura 15 se muestra la simulación con síntesis modificando los componentes de la biblioteca cmos_cells para que estos posean retardo, además que se aumenta la frecuencia por encima de su máximo, ya que se sabe que si no se cumple con la frecuencia máxima de operación el comportamiento de la salida será erróneo. Por medio del comando de iverilog, donde se comprueba que el modelo conductual y el archivo sintetizado poseen el diferente

comportamiento debido a que el checker señala algunas diferencias entre ambas señales. En la figura 16 se puede comprobar de manera gráfica mediante el gtkwave.

```
MacBook-Air-de-Alberto:tarea5 Beto$ make iverilog
iverilog -o tarea5 BancoPrueba_tarea5.v cmos_cells.v
cmos_cells.v:11: warning: choosing typ expression.
cmos_cells.v:11: warning: choosing typ expression.
cmos_cells.v:17: warning: choosing typ expression.
cmos_cells.v:17: warning: choosing typ expression.
cmos_cells.v:23: warning: choosing typ expression.
cmos_cells.v:23: warning: choosing typ expression.
vvp tarea5
VCD info: dumpfile mux_tarea5.vcd opened for output.
```

	clk,	out,	reset,	selector,	in0,	in1
	0	xx	xx	0	0	000
	10	1	00	xx	0	000
	20	0	00	xx	0	000
	30	1	00	00	0	112
	40	0	00	00	0	112
	50	1	11	11	1	010
	60	0	11	11	1	010
	70	1	00	10	1	002
	80	0	00	10	1	002
Diferencia modulo conductual y estructural						
	90	1	10	00	0	113
	100	0	10	00	0	113
Diferencia modulo conductual y estructural						
	110	1	11	11	1	001
	120	0	11	11	1	001
	130	1	01	11	0	100
	140	0	01	11	0	100
Diferencia modulo conductual y estructural						
	150	1	10	10	1	100
	160	0	10	10	1	100
	170	1	00	00	1	100

Figura 15: Ejemplo iverilog con síntesis con componentes de la biblioteca cmos_cells con retraso

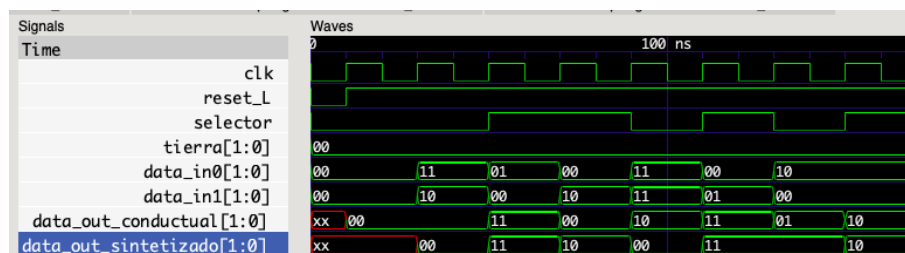


Figura 16: Ejemplo gtkwave con síntesis con componentes de la biblioteca cmos_cells en retardo

5. Análisis y conclusiones

Analizando los ejemplos de la sección anterior, se puede notar que los distintos casos cumplen con lo esperado. Ya que como se ve con los componentes internos, con los de la biblioteca cmos_cells y con los de la biblioteca cmos_cells con retardo, para cada uno de estos casos se obtuvo el mismo comportamiento a la salida del archivo sintetizado con respecto a la descripción conductual. En cuanto a la frecuencia máxima, modificando el

toggle en severas ocaciones, se encontró que la frecuencia máxima se presenta cuando hay un periodo de 23.6ns, por ende dado la ecuación 1 se tiene una frecuencia máxima es de 42.37Mhz.

$$F_{max} = \frac{1}{T_{max}} = \frac{1}{23,6ns} = 42,37Mhz \quad (1)$$

El diseño estructural brindado por la síntesis automática de yosys se presenta en la figura 17. En la figura 18 se muestran los componentes y su cantidad necesaria para realizar el circuito, a pesar que no sale el floop se sabe por el diagrama que se ocupan 2. Debido a que se necesitan 4 NAND, 4 NOR, 2 NOT y 2 Floop, el costo de los componente es de \$7.54 como se ve en la ecuación 2. Se trabajaron 3 horas, entonces cobrando \$15 la hora se tendría un costo total de \$52.536 por el circuito, como se ve en la ecuación 3.

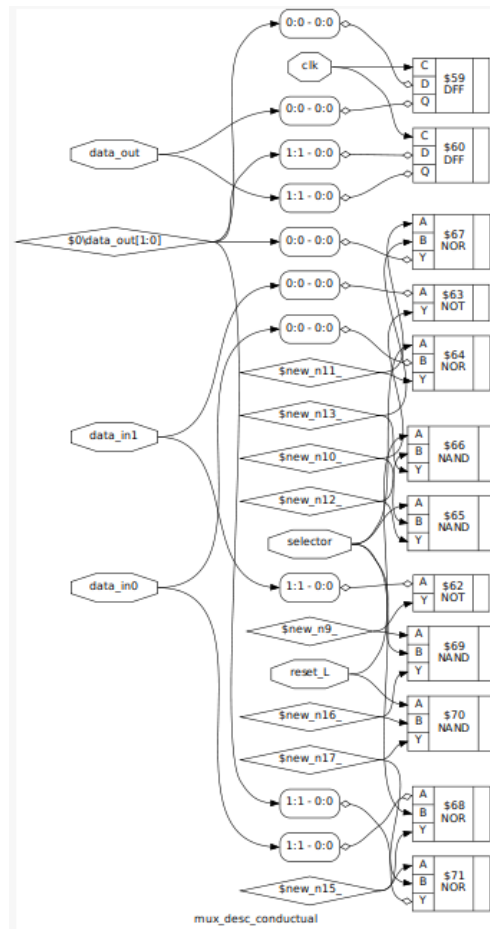


Figura 17: Diagrama del circuito

12.1.2. Re-integrating ABC results.		
ABC RESULTS:	NAND cells:	4
ABC RESULTS:	NOR cells:	4
ABC RESULTS:	NOT cells:	2

Figura 18: Componentes necesarios

$$Costo_{componentes} = 4(NAND + NOR) + (2 * NOT) + (2 * Floop) \quad (2)$$

$$Costo_{componentes} = 4(0,063 + 1,653) + 2(0,063) + 2(0,273) = \$7,54$$

$$Costo_{total} = 4(NAND + NOR) + (2 * NOT) + (2 * Floop) + 3(Costo1hora) \quad (3)$$

$$Costo_{total} = \$6,99 + \$45 = \$52,536$$

Comparando el diseño estructural manual y el sintetizador automático se pueden obtener beneficios de alguna u otra dependiendo del uso que se le quiera dar. Con respecto al tiempo invertido, realizar el diseño estructural mediante el sintetizador automático acelera bastante el proceso. Empezando porque no hay que diseñar cada compuerta de manera manual, sino que la biblioteca `cmos_cells` ya las incluye, además que de esta manera prevenimos errores o pulgas que podrían aparecer a la hora de diseñarlo uno mismo. Además que `yosys` se encarga de las conexiones de las compuertas, por lo que uno simplemente se debe preocupar por las lógica que posee el componente. Sin embargo al utilizar el sintetizador automático hay que tener profundo cuidado a la hora de utilizar frecuencias altas, ya que si se sobrepasa de la frecuencia máxima, este va a presentar una salida errónea. No como el diseño estructural manual, que en caso de sobrepasar la frecuencia máxima sufría un retraso en la señal y estas pasaban después, sin embargo si presentaba los valores correctos con respecto a la descripción conductual

Se puede concluir que realizar el diseño estructural por medio de la síntesis automática es mucho más sencillo que hacerlo de manera manual, aparte de que a uno le ahorra tiempo y mediante la importación de bibliotecas uno se asegura que esas están libres de errores. Se comprobó que para todos los casos se presentó el valor esperado a la hora de comparar la descripción conductual con la estructural. Se obtuvo una frecuencia máxima de 42.37Mhz y se comprobó la importancia de no sobrepasar esta frecuencia, ya que de lo contrario se obtendrán valores erróneos. En el costo de horas de trabajo se le resto 1 hora al total del tiempo invertido en la tarea, debido a que esa fue realizando el reporte, no propiamente algo con el circuito. Cabe mencionar que para esta tarea se utilizó código de las tareas anteriores, y al igual que en la tarea 3 y 4 se omite el retraso en el flip flop debido a que no interfiere en toda la lógica que tiene por detrás.

6. Distribución del tiempo invertido en la tarea

En total se duró 4 horas con 20 minutos realizando la tarea

- Buscar información: 10 minutos, debido a que la mayoría se vió en clase
- Estudiando la información tomó 1 hora y 10 minutos debido a que tuve que repasar y analizar la clase anterior
- Ejecutando lo que decidió hacer y probándolo se tomó 2 horas
- Realizar el reporte tomó 1 hora.