

# Taller de Lógica Digital - Parte 2

Organización del Computador 1

Primer Cuatrimestre 2023

## Ejercicios

### 1. Componentes de 3 estados

a) Completar la siguiente tabla:

A	A <sub>en</sub>	B	B <sub>en</sub>	C	C <sub>en</sub>	Estimado	Obtenido
0	0	0	0	0	0	No Conectado	No Conectado
0	1	1	1	0	0	Error	Error
1	0	1	0	1	0	No conectado	No conectado
1	1	0	0	0	1	Error	Error
0	1	0	1	0	1	0	0
0	1	1	1	1	1	Error	Error
1	0	1	1	1	0	1	1

b) Completar la siguiente tabla:

Color	Interpretación
Gris	Cables sin ningún conexión.
Verde claro	La señal es 1.
Verde oscuro	La señal es 0.
Azul	La señal es indefinida o desconocida.
Rojo	Están pasando dos valores distintos por el mismo cable.

c) Enunciar la regla:

La condición o regla sería la siguiente:

Si dos o más componentes de 3 estados están dejando pasar la señal entonces esas señales deben tener el mismo valor.

d) Explicar cuáles son y por qué:

Las combinaciones basura son aquellas que tienen más de un componente de 3 estados activo al mismo tiempo. Esta se debe a que son lógicamente inútiles (y la misma lógica podría ser representado con menos compuertas de forma más simple).

### 2. Transferencia entre registros

a) Detallar entradas y salidas:

Entradas:
<b>ckl:</b> control que determina cuando se guarda el dato de entrada Force_input
<b>Force_input:</b> data a almacenar en los registros
<b>en_Force_input:</b> control para almacenar un valor arbitrario en el circuito
Salidas:
<b>R0, R1, R2:</b> de monitoreo de dato almacenado en los registros

b) Secuencia de señales:

- I. Habilito en\_Force\_input en 1,
- II. Habilito la escritura(W) de R1 en 1,
- III. Habilito Force\_input en 1,
- IV. Hago un clk (franco ascendente).

c) Secuencia de señales:

- I. en\_out R0 en 1,
- II. w de R2 en 1,
- III. clk(franco ascendente),
- IV. desactivo w de R1
- V. y desactivo en\_out R0,
- VI. ---- en\_out de R2 en 1,
- VII. w de R0 en 1,
- VIII. clk(franco ascendente),
- IX. desactivo w de R0,
- X. desactivo en\_out de R2
- XI. --- activo w de R2,
- XII. activo en\_out de R1 ,
- XIII. clk(franco ascendente).

### 3. Máquina de 4 registros con suma y resta.

a) Detallar entradas y salidas:

Entradas Control:	
<b>OP</b> :	señal de control de operaciones de la ALU
<b>clk</b> :	señal de control para guardar en los registros
<b>De Reg0_Write ...Reg3_Write</b> :	señal de control que habilita la escritura
<b>De Reg0_EnableOut...Reg3_EnableOut</b> :	señal de control que habilita la salida de los registros 00 a 03
<b>ALU_A_Write y ALU_B_Write</b> :	señal de control que habilita a la escritura de los registros de la ALU A y B
<b>ALU_EnableOut</b> :	señal de control que habilita al calculo almacenado en la ALU_regs
<b>en_Force_input</b> :	señal de control que habilita la entrada de un valor arbitrario a los registros de 00 a 03
Entrada:	
<b>Force_input</b> :	ingresa un valor arbitrario en los registros 00 al 03
Salidas:	
FLAGS de la ALU:	
<b>N</b> :	Indica si el valor almacenado es negativo
<b>Z</b> :	Indica si el valor almacenado es 0
<b>V</b> :	Indicar si hubo overflow
<b>C</b> :	en el caso de la suma indica si hubo carry -out y en el caso de la resta indica si hubo borrow.

b) Detallar el contenido de cada display:

Displays	Funcionalidad
Displays de registros 00 hasta 03	Todos representan el valor almacenado en su registro de 4 bits correspondiente.
Displays conectados A y B debug	representan el valor almacenado en los registros A y B de la ALU_regs.
Display S-debug	representa el valor almacenado resultante de la operación realizada por la ALU_regs.

**Como observación final todos los displays interpretan la información almacenada como sin signo y en base hexadecimal.**

c) Secuencia de señales:

- I. Activar Reg2\_Write de R2,
- II. Activar en\_Force\_input,
- III. En el Force\_input ingreso 0100 (4),
- IV. Realizar un clk(franco ascendente),
- V. Volver a clk (0),
- VI. ---- Desactivar Reg2\_Write de R2,
- VII. Activar Reg3\_Write R3,
- VIII. Ingresar en el Force\_input 1101 (-3),
- IX. clk (franco ascendente).

**Los números están interpretados en complemento a 2.**

d) Completar la siguiente tabla:

Los resultados interpretados en sin signo y en complemento a 2.

Valor inicial	Resultado operación 1	Flags	Resultado operación 2	Flags
(4,0)	C2: 4 Sin-signo: 4	N:0,Z:0,V:0,C:0	C2: 4 Sin-signo: 4	N:0,Z:0,V:0,C:0
(7, -1)	C2: -8 Sin-signo: 8	N:1,Z:0,V:1,C:1	C2: 7 Sin-signo: 7	N:0,Z:0,V:0,C:0
(-8, -2)	C2: 6 Sin-signo: 6	N:0,Z:0,V:1,C:1	C2: -6 Sin-signo: 10	N:1,Z:0,V:1,C:1
(8, -9)	C2: Sin-signo:		C2: Sin-signo:	

Secuencia de señales de primer caso:

Almacenamos los datos en el R1 y R0:

- I. Activar en\_Force\_input,
- II. Cargar en Force\_input el numero 0100(4),
- III. Habilitar Reg0\_Write de R0 ,
- IV. clk(franco ascendente),
- V. Desactivar Reg0\_Write de R0,en\_Force\_input ,
- VI. Activar Reg1\_Write de R1,
- VII. Desactivar en\_Force\_input,
- VIII. clk(franco ascendente).
- IX. -- Habilitar la salida de R0,
- X. Activar w de Alu\_A\_WRITE,
- XI. clk(franco ascendente),
- XII. Desactivar w de Alu\_A\_Write,
- XIII. Desactivar la salida de R0,
- XIV. Activar la salida de R1,
- XV. Activar W ALU\_B\_WRITE,
- XVI. clk(franco ascendente).
- Ahora realizar la operaciones:
- XVII. Operación: "OR" en op(11),
- XVIII. clk(franco ascendente),
- XIX. Activar la salida de alu\_regs,activo Reg2\_Write de R2,

- XX. clk(franco ascendente),
- XXI. desactivo la Reg2\_Write R2,
- XXII. Operación: "sub" en op(01),
- XXIII. clk(franco ascendente),
- XXIV. Habilitar Reg3\_Write de R3,
- XXV. clk(franco ascendente).

e) Explicar

## Corrección

Integrantes:

Nombre y Apellido:

LU:

Nombre y Apellido:

LU:

Para uso de los docentes:

1	2	3