

# Taller de Lógica Digital - Parte 2

Organización del Computador 1

Primer Cuatrimestre 2022

## 1. Enunciado

El Taller consistirá en analizar y desarrollar los componentes avanzados necesarios para la construcción de una computadora moderna.

## 2. El simulador (Logisim)

Para más información ver la primer parte del taller.

## 3. Ejercicios

Durante el presente taller sólo podrán utilizarse compuertas lógicas básicas de dos entradas (AND, OR, XOR) y la compuerta NOT, el buffer de 3 estados (*Controlled Buffer*) de un *bit* y el flip-flop D de un *bit*. Éste último componente ya está implementado en el simulador y puede encontrarse en “Memoria/Biestable D”.

Para la realización de este taller deberá utilizar el archivo `TallerLogica-Secuenciales.circ` que puede descargar de la página de la materia.

---

Inicio checkpoint 1

1. **Componentes de 3 estados.** El componente llamado `3estados` tiene 3 entradas de datos (A, B y C), 3 de control (A\_en, B\_en y C\_en) y una salida (Res).

- Realizar un análisis exhaustivo y describir el comportamiento del circuito para los distintos valores que pueden tomar las entradas.
- Explicar qué significan los distintos colores que toman los cables.
- Proponer alguna regla para que en la salida el cable nunca tome el color rojo.
- Por motivos didácticos en la materia se interpretan como basura ciertas combinaciones que en el simulador no lo son. Indicar cuáles son y por qué.

---

Fin checkpoint 1

---

Inicio checkpoint 2

2. **Transferencia entre registros.** El componente llamado `ej-2` presenta la composición de varios componentes.

Tiene internamente 3 ejemplares del componente llamado `registro-salida-restringida`, que se encuentra incompleto. Este componente tiene la capacidad de almacenar un valor interno. Completarlo de manera que su comportamiento sea:

- Cuando la señal de entrada `clk` pasa a su flanco alto, si la señal de entrada `W` vale 1, entonces se debe almacenar internamente el valor de la entrada `in`. En caso de que no se cumpla alguna condición de las antes nombradas el valor interno no se modifica.
- Cuando la señal de entrada `en_out` vale 1, en la salida `out` se debe poder leer el valor interno almacenado. Caso contrario esta salida debe estar en alta impedancia (**Hi-Z**).

- Por motivos didácticos se pide también una salida de debug para poder observar el valor interno almacenado (salida `out_debug`) todo el tiempo.

Una vez completado el componente anterior, analizar el componente **ej-2** y responder:

- ¿Cuáles son y qué respresenta cada entrada y cada salida del componente? ¿Cuáles entradas deben ser consideradas como de control?
- Las entradas **Force\_input** y **en\_force\_input** sirven para poder introducir en el circuito un valor arbitrario. Escribir una secuencia de activación y desactivación de entradas para que el registro R1 pase a tener el valor 1.
- Dar una secuencia de activaciones que inicialmente ponga un valor arbitrario en R0 (suponer para un valor y luego generalizarlo), luego que este valor se transfiera a R1, luego que el valor de R2 pase a R0 y finalmente el valor de R1 a R2.

---

Fin checkpoint 2

---



---

Inicio checkpoint 3

---

- Máquina de 4 registros con ALU.** El componente **ej-3** representa una máquina con 4 registros *de propósito general* y una ALU. Todo lo antes nombrado es de 4 *bits* y las operaciones se realizan en complemento a 2.

La ALU se encuentra incompleta. La misma debe completarse en el circuito **ALU\_con\_Reg** agregando y conectando correctamente la ALU realizada en el taller previo. Para ello en el menú *proyecto*, utilizar la opción *Cargar circuito → Biblioteca Logisim...* y seleccionar el archivo del taller pasado resuelto. Luego agregar el componente ALU.

Todos los registros usados son ejemplares de **registro-4b-salida-restringida**. El mismo debe extender el registro de un *bit* del ejercicio anterior (**registro-salida-restringida**) para que almacene cuatro *bits*. El comportamiento es el mismo, la entrada **in\_4** se debe separar en 4 *bits* independientes y ser guardados, y al momento de devolverlos (tanto por **out** como por **out4\_d**) deben ser reagrupados conservando el orden.

Una vez completado el componente anterior, analizar el componente **ej-3** y responder:

- ¿Cuáles son y qué respresenta cada entrada y cada salida del componente? ¿Cuáles entradas deben ser consideradas como de control?
- ¿Qué información muestra cada display y cómo es interpretada esta información?
- Escribir la secuencia de activación y desactivación de señales para cargar el valor 4 en el registro R2 y el -3 en el registro R3
- Cargar los siguientes pares de valores en R0 y R1 respectivamente, con las operaciones indicadas:

- 1) (4, 0), OR, SUB
- 2) (7, -1), SUB, AND
- 3) (-8, -2), ADD, SUB
- 4) (8, -9), OR, AND

Realizar las dos operaciones indicadas y guardar los resultados en R2 y R3 respectivamente. Para el primer caso, escribir la secuencia completa de activación de señales.

- ¿Por qué se niega la señal *clk* en la ALU antes de pasarle este valor al registro de salida?

---

Fin checkpoint 3

---