

# El language de descripción de hardware Verilog

March 19, 2024

## Herramientas

EDApplayground: <https://www.edaplayground.com/>

**Nota:** En todos los ejercicios se deberá implementar un módulo de simulación (testbench) y simular la resolución del mismo.

**Ejercicio 1** - Implementar las compuertas: AND, OR, XOR, AOI, OAI.

**Ejercicio 2** - Implementar un multiplexor de dos líneas de selección y 4 entradas de datos.

**Ejercicio 3** - Implementar un multiplexor de 4 líneas de selección y 16 entradas de datos. Cada entrada de datos debe ser de un ancho de  $N$  bits.

**Ejercicio 4** - Implementar un sumador con/sin signo genérico de  $N$  bits.

**Ejercicio 5** - Implementar un restador con/sin signo genérico de  $N$  bits.

**Ejercicio 6** - Implementar un multiplicador sin signo genérico de  $N$  bits.

**Ejercicio 7** - Implementar un multiplicador en complemento a dos genérico de  $N$  bits.

**Ejercicio 8** - Implementar cuatro diferentes flip flops D con las siguientes entradas en cada caso:

- Reset asincrónico
- Reset sincrónico
- Set asincrónico
- Set sincrónico

**Ejercicio 9** - Implementar cuatro diferentes registros de  $N$  bits con las siguientes entradas en cada caso:

- Reset asincrónico
- Reset sincrónico
- Set asincrónico
- Set sincrónico

**Ejercicio 10** - Implementar un contador genérico de  $N$  bits.

**Ejercicio 11** - Implementar un contador genérico de  $N$  bits y señal de carga. Cuando dicha señal de carga sea '1', el valor de la cuenta pasará a ser el valor de una señal de entrada de  $N$  bits.

**Ejercicio 12** - Implementar un contador genérico de  $N$  bits y señal de *enable*. Cuando dicha señal de *enable* sea '0', el valor de la cuenta no deberá ser modificado.

**Ejercicio 13** - Implementar un contador genérico de  $N$  bits con señal de *enable* y señal de carga. Cuando la señal de *enable* sea '0', el valor de la cuenta no deberá ser modificado. Cuando la señal de carga sea '1', el valor de la cuenta pasará a ser el valor de una señal de entrada de  $N$  bits.

**Ejercicio 14** - Implementar un registro de desplazamiento de  $N$  bits de largo.

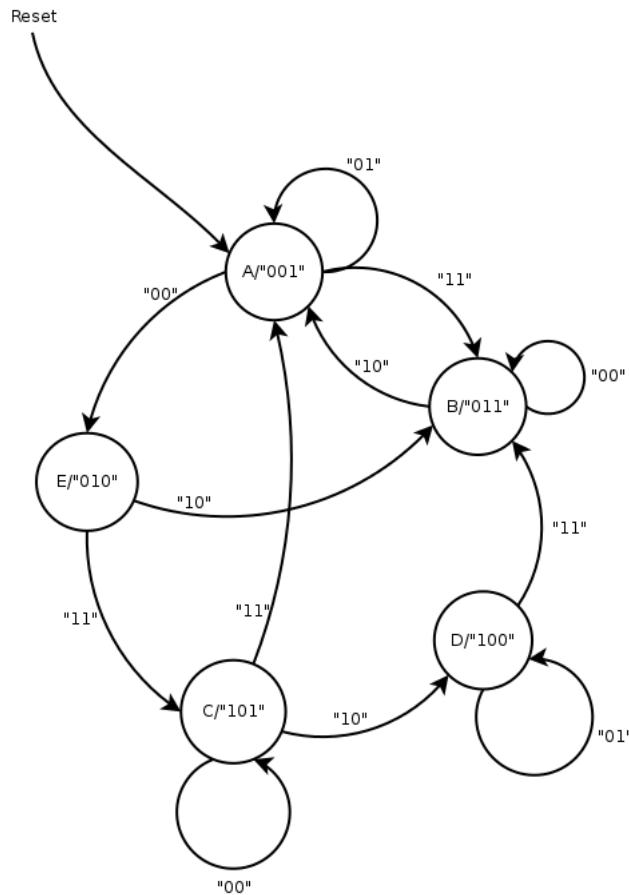
**Ejercicio 15** - Implementar un registro de desplazamiento genérico de  $N$  bits y señal de carga. Cuando dicha señal de carga sea '1', el contenido del registro pasará a ser el valor de una señal de entrada de  $N$  bits.

**Ejercicio 16** - Implementar un registro de desplazamiento genérico de  $N$  bits y señal de *enable*. Cuando dicha señal de *enable* sea '0', el contenido del registro no deberá ser modificado. Implementar también señal de carga.

**Ejercicio 17** - Implementar un registro de desplazamiento genérico de  $N$  etapas y de un ancho de palabra de  $M$  bits. El mismo deberá tener señal de carga y señal de *enable*.

**Ejercicio 18** - Implementar un circuito que detecte la secuencia "0010110" de una señal de entrada. Deberán realizarse tres implementaciones diferentes: mediante una FSM, mediante una FM on codificación one-hot y mediante un registro de desplazamiento y la lógica de detección adecuada.

**Ejercicio 19** - Implementar el circuito de Moore correspondiente al siguiente diagrama de estados:



**Ejercicio 20** - Implementar un detector de flancos. Este circuito tendrá dos salidas denominadas *rising* y *falling* las cuales serán '1' cuando en la señal de entrada se produzca un flanco de subida o de bajada respectivamente.

**Ejercicio para entregar** - Implementar en Verilog el ejercicio de la unidad 1.