

**Tarea #9**  
(Entrega 27 de junio de 2019)  
Sumador con pipeline

**\*\*\*OJO\*\*\*** Al igual que en la **Tarea #1** tome el tiempo que demora en hacer cada una de las cosas solicitadas: búsqueda de información, diseño, elaboración de las pruebas, ejecución de las simulaciones, etc.

**Evaluación**

- |                               |     |
|-------------------------------|-----|
| 1. Funcionamiento del diseño: |     |
| a. Descripción conductual     | 35% |
| b. Descripción estructural    | 35% |
| c. Pruebas y verificador      | 15% |
| d. Makefile                   | 5%  |
| 2. Documentación              | 10% |

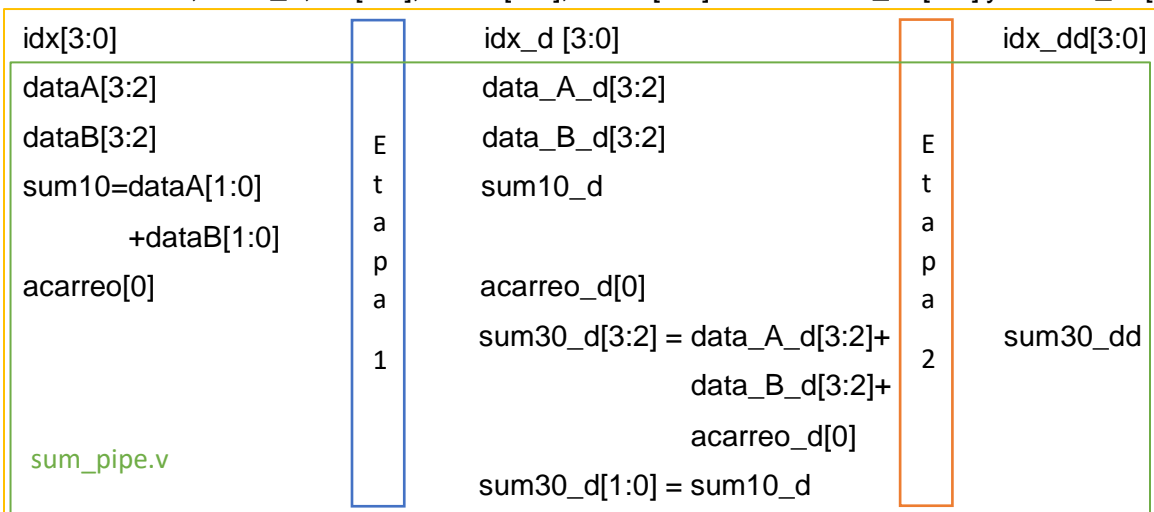
**Trabajo a realizar**

Cuando se desean realizar operaciones matemáticas entre dos buses muy grandes, muchas veces el cálculo no se puede realizar en un único ciclo de reloj y se necesita la implementación de un pipeline para poder realizar la operación.

Se va a diseñar un sumador que inicialmente suma las partes menos significativas de ambos buses y los almacene en un flop. Luego, suma las partes más significativas más el acarreo de la suma anterior y los almacene en otro flop. Este bloque deberá colocarse dentro de un módulo independiente de verilog, `sum_pipe.v`

En paralelo con esta lógica, se tendrá un identificador que se deberá retrasar la misma cantidad de ciclos que tarda la suma.

Entradas: `clk`, `reset_L`, `idx[3:0]`, `dataA[3:0]`, `dataB[3:0]`. Salidas: `idx_dd[3:0]` y `sum30_dd[3:0]`



`sumador.v`

Figura 1. Sumador con pipeline