

ESCUELA COLOMBIANA DE INGENIERÍA  
ARQUITECTURA DEL COMPUTADOR  
Proyecto

---

Objetivos

- Implementar en el simulador de hardware la Memoria y la Unidad Aritmética de un computador sencillo
- 

1. **Circuitos:** Implementar en el simulador de hardware los siguientes circuitos:

- **Memoria:** Implementar un circuito en HDL para una memoria de 32 posiciones y 48 bits en cada posicion. Use para la direccion un vector de 5 bits (a), use para la entrada tres vectores de 16 bits (in2, in1 e in0), use para la salida tres vectores de 16 bits (out2, out1 y out0) y use un bit (w) para indicar si se escribe a la memoria o no. Utilice el circuito Bit de la carpeta BuiltIn. El archivo del circuito debe llamarse **memory.hdl**
- **Complemento a 2:** Implementar un circuito en HDL que realice el complemento a 2 de un número de 48 bits. Utilice en la entrada tres vectores de 16 bits (a2, a1 y a0) y en la salida tres vectores de 16 bits (b2, b1 y b0). El archivo del circuito debe llamarse **complement2.hdl**
- **Incremento de 48 bits:** Implementar un circuito en HDL que realice el incremento en uno de un número de 48 bits. Utilice en la entrada tres vectores de 16 bits (a2, a1 y a0) y en la salida tres vectores de 16 bits (b2, b1 y b0). El archivo del circuito debe llamarse **increment.hdl**
- **Suma de 48 bits:** Implementar un circuito en HDL que realice la suma de dos numeros de 48 bits usando complemento a 2. Utilice en la entrada seis vectores de 16 bits (a2, a1, a0, b2, b1 y b0) y en la salida tres vectores de 16 bits (c2, c1 y c0). El archivo del circuito debe llamarse **add.hdl**
- **Resta de 48 bits:** Implementar un circuito en HDL que realice la resta de dos numeros de 48 bits usando complemento a 2. Utilice en la entrada seis vectores de 16 bits (a2, a1, a0, b2, b1 y b0) y en la salida tres vectores de 16 bits (c2, c1 y c0). El archivo del circuito debe llamarse **subtract.hdl**
- **Unidad Aritmética:** Implementar un circuito en HDL para la unidad aritmetica. Utilice nueve entradas de 16 bits (CIn2, CIn1, CIn0, Ain2, Ain1, Ain0, SLin2, SLin1 y SLin0) y una entrada de 3 bits (inst). Utilice nueve salidas de 16 bits (CIout2, CIout1, CIout0, Aout2, Aout1, Aout0, SLout2, SLout1 y SLout0). Las salidas que no se utilicen en la instrucción a ejecutar deben tomar el valor cero en todos los bits. El archivo del circuito debe llamarse **AU.hdl**

Entrada	Acción
000	$CIout \leftarrow SLin$
001	$CIout \leftarrow CIn + SLin$
010	$Aout \leftarrow -SLin$
011	$SLout \leftarrow Ain$
100	$Aout \leftarrow Ain - SLin$
101	$Aout \leftarrow Ain - SLin$
110	Si $Ain < 0$ entonces $CIout \leftarrow CIn + 1$
111	Ninguna