

Objetivos

- Implementar en el simulador de hardware la Memoria y la Unidad Aritmética de un computador sencillo complemento a 2
-

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 dirección un vector de 5 bits (a), use para la entrada cuatro vectores de 12 bits (in3, in2, in1 e in0), use para la salida cuatro vectores de 12 bits (out3, out2, out1 y out0) y use un bit (w) para indicar si se escribe a la memoria o no. El archivo del circuito debe llamarse **memory.hdl**
- **Unidad Aritmética:** Implementar un circuito en HDL para la unidad aritmética. Utilice como entradas doce vectores de 12 bits (CIin3, CIin2, CIin1, CIin0, Ain3, Ain2, Ain1, Ain0, SLin3, SLin2, SLin1 y SLin0) y un vector de 3 bits (inst). Utilice como salidas doce vectores de 12 bits (CIout3, CIout2, CIout1, CIout0, Aout3, Aout2, Aout1, Aout0, SLout3, 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 inst	Acción
000	CIout \leftarrow SLin
001	CIout \leftarrow CIin + 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 CIin + 1
111	Ninguna

- Solo utilice los circuitos Bit, And, Or, Xor, Not, HalfAdder, FullAdder, DMux8Way de la carpeta builtInCircuits y circuitos propios.