**Práctica 8:** Procesador RISC

Laboratorio de Organización y Arquitectura de Computadoras

**Grupo:** 2

**Alumnos:**

* Guzmán Sánchez José Emmanuel
* Mejía Ortiz Aarón Enrique
* Sáenz Barragán Ricardo

Objetivo

Diseñar un microprocesador RISC en VHDL, específicamente la versión de pipeline del microprocesador 68HC11 de Motorola.

**Dispositivo**: MAX10 DE-Lite 10M50DAF484C7G

1. Desarrollo

Partiendo del diagrama del microprocesador 68HC11 de Motorola, algunos bloques del mismo fueron proporcionados, por lo que sólo fue necesario implementar el registro de estados y banderas (CCR) y el secuenciador.

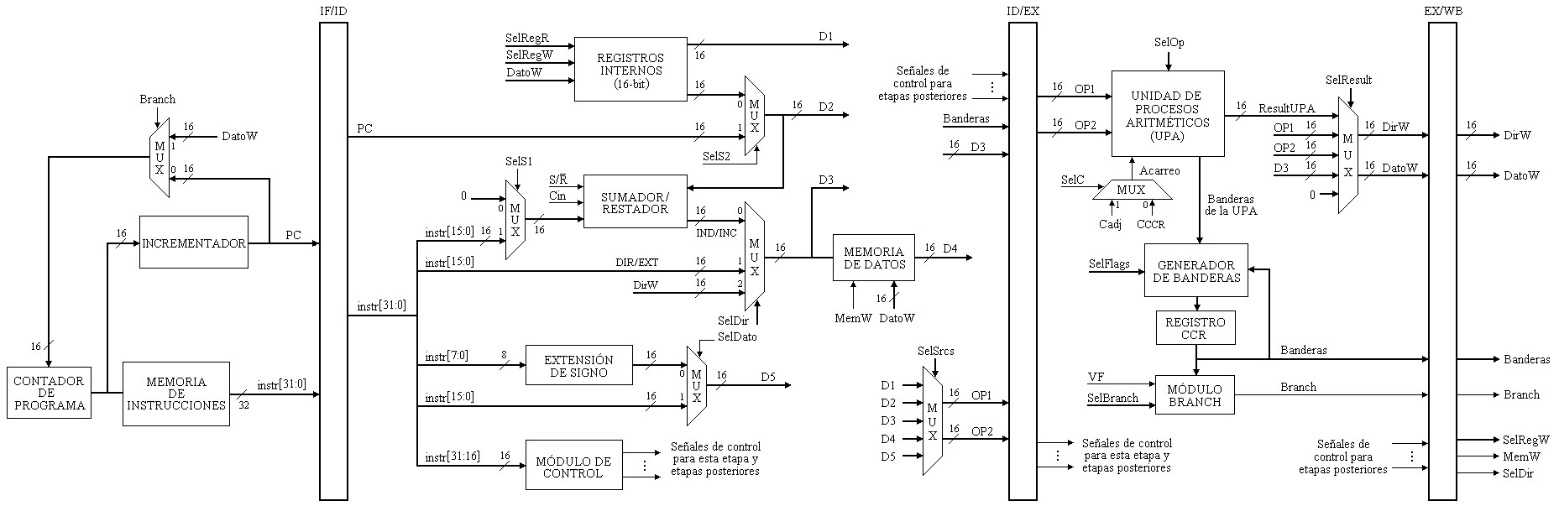


Figura 1. Diagrama del microprocesador 68HC11 de Motorola en su versión pipeline

lknkln

nkjnkj

Figura 2. Diagrama del registro CCR (de estados y banderas)

Figura 3. Diagrama del secuenciador

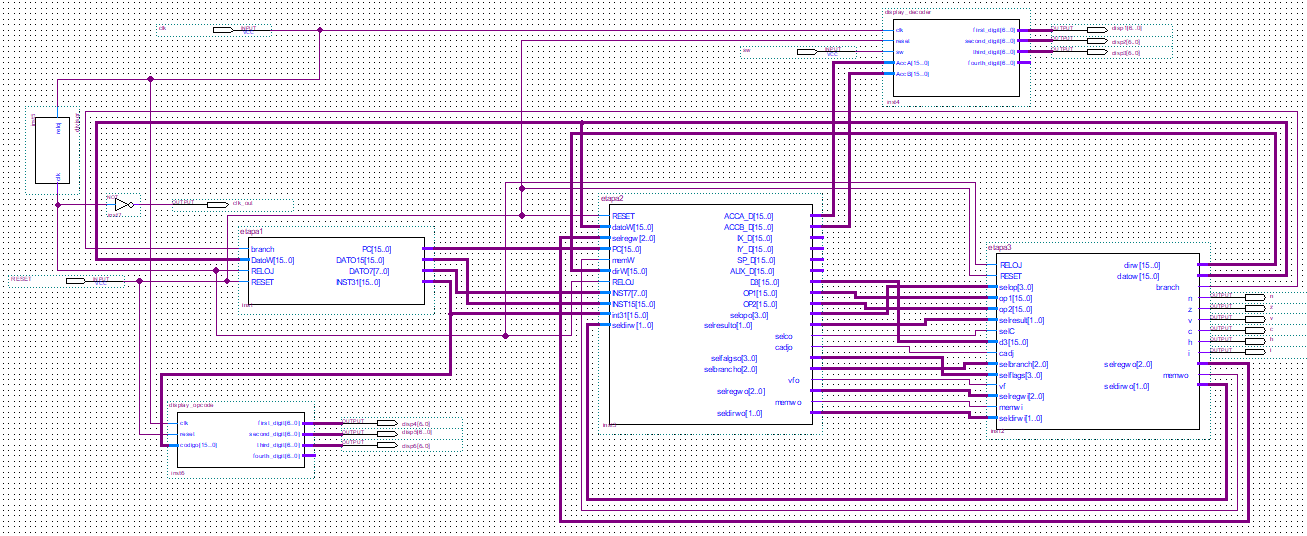


Figura 4. Diagrama de bloques del microprocesador.

Para poder probar la arquitectura es necesario implementar instrucciones que se ejecuten en la misma. Como prueba se propusieron las instrucciones que se muestran en la Fig. 4. Una vez comprobado el funcionamiento de la arquitectura, se pidió ejecutar el programa mostrado en la Fig. 5.

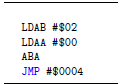
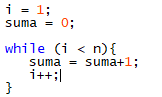
 

Figura 5. Instrucciones a ejecutar Figura 6. Programa a ejecutar

Para ejecutar las instrucciones anteriores es necesario definir las cartas ASM de cada una, es decir, cada uno de los estados por los que necesitan transitar el procesador para lograr ejecutarla. De igual manera, fue necesario definir sus códigos de operación, en este caso se ocuparon los mismos que el set de instrucciones del procesador.

Para ejecutar el programa, primero fue necesario definir las instrucciones que nos permitirían realizarlo, una vez definidas, construimos las cartas ASM de cada una de las instrucciones necesarias para la ejecución de nuestro programa

Finalmente, colocamos dentro de la memoria externa las instrucciones a ejecutar con sus respectivos operadores

Figura 7. Contenido de la memoria para las instrucciones de prueba y el programa de prueba

Las cartas ASM de cada uno de las instrucciones que necesitaremos se muestran a continuación

Figura 8. Cartas ASM de las instrucciones

Con lo anterior, podemos construir el contenido de la memoria del secuenciador para que ejecute cada instrucción. Esto se logra activando cada una de las banderas necesarias.

Figura 9. Cartas ASM de las instrucciones necesarias para el ciclo while

Una vez que hemos puesto en la memoria externa las instrucciones a ejecutar con sus respectivos operadores, si es el caso, y en la memoria del secuenciador las cartas ASM de cada una de las instrucciones la arquitectura está lista.

1. Simulaciones

Después de 6 ciclos, en el acumulador B se carga el dato que obtiene de la memoria por acceso inmediato, en este caso un 2. Para el acumulador A, en los primeros 6 ciclos se carga el dato en B, los siguientes 6 ciclos se carga el dato 0 en el acumulador A y para los siguientes 5 ciclos se suma A y B y se guarda en el acumulador A.

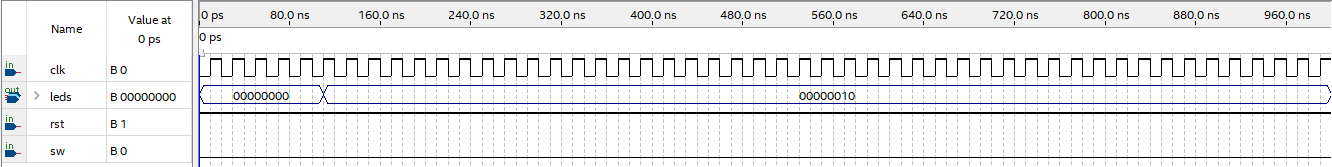


Figura 10. Contenido del acumulador B

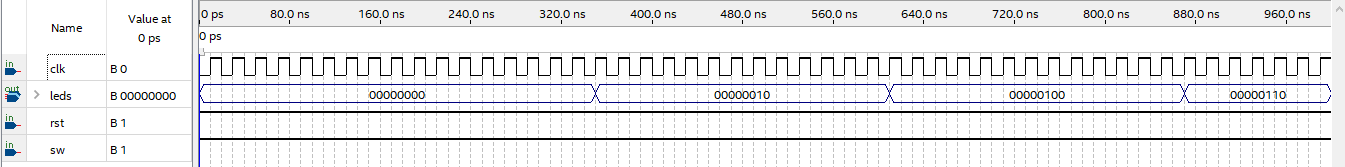


Figura 11. Contenido de la

1. Conclusiones

Guzmán Sánchez José Emmanuel

.

Mejia Ortiz Aarón Enrique

.

Sáenz Barragán Ricardo

.