**Práctica 7:** Procesador CISC 68HC11

Laboratorio de Organización y Arquitectura de Computadoras

**Grupo:** 2

**Alumnos:**

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

Objetivo

Diseñar un procesador CISC de 8 bits en VHDL, específicamente un ‘clon’ 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.

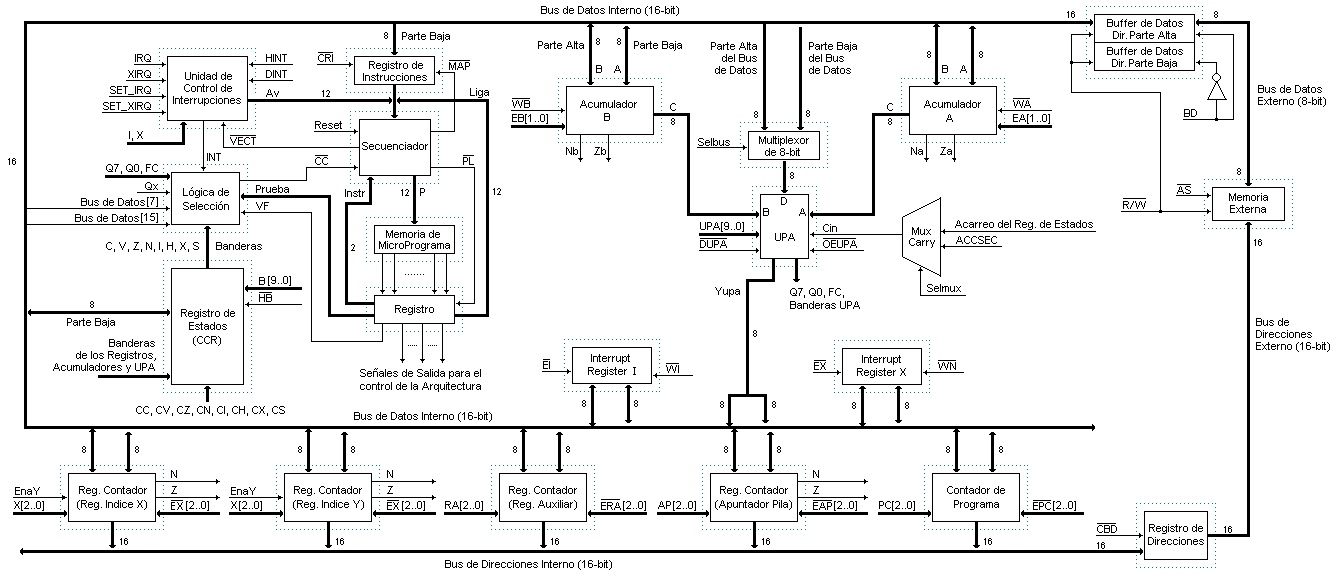


Figura 1. Diagrama del microprocesador 68HC11 de Motorola

Para el registro de estados fue necesario implementar los multiplexores que nos indican el origen de cada una de las banderas, ya sea de los acumuladores, los registros, la UPA o del bus de datos. Así mismo, cada una de las banderas tiene un registro que es donde se guardará su valor.

El secuenciador contiene los bloques de la lógica de selección, el registro de microinstrucción la memoria que contiene los códigos de las instrucciones y el registro del secuenciador, que es el que mandará las señales a los demás componentes del microprocesador.

Una vez construidos estos componentes, los conectamos con los que nos proporcionaron para poder armar el *clon* del procesador.

Aprovechamos la funcionalidad que tiene VHDL de poder conectar 2 líneas sólo asignándoles etiquetas con el mismo nombre, de este modo evitamos cruzar las conexiones a través de todo el procesador.

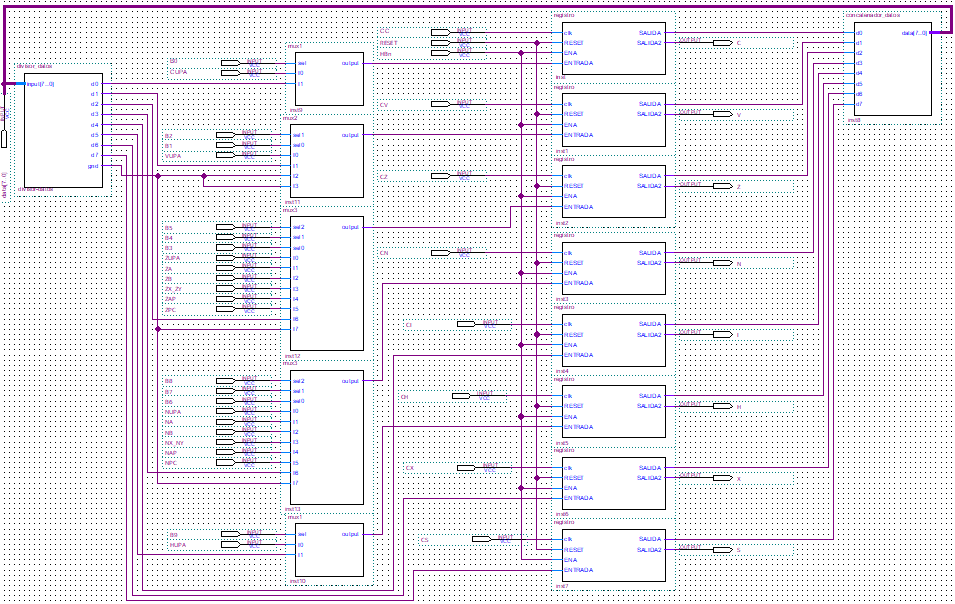


Figura 2. Diagrama del registro CCR (de estados y banderas)

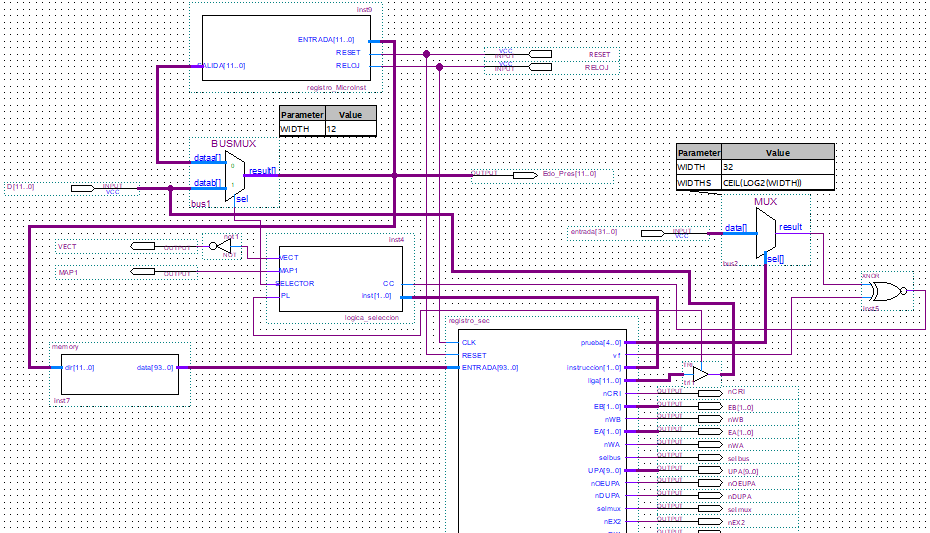


Figura 3. Diagrama del secuenciador

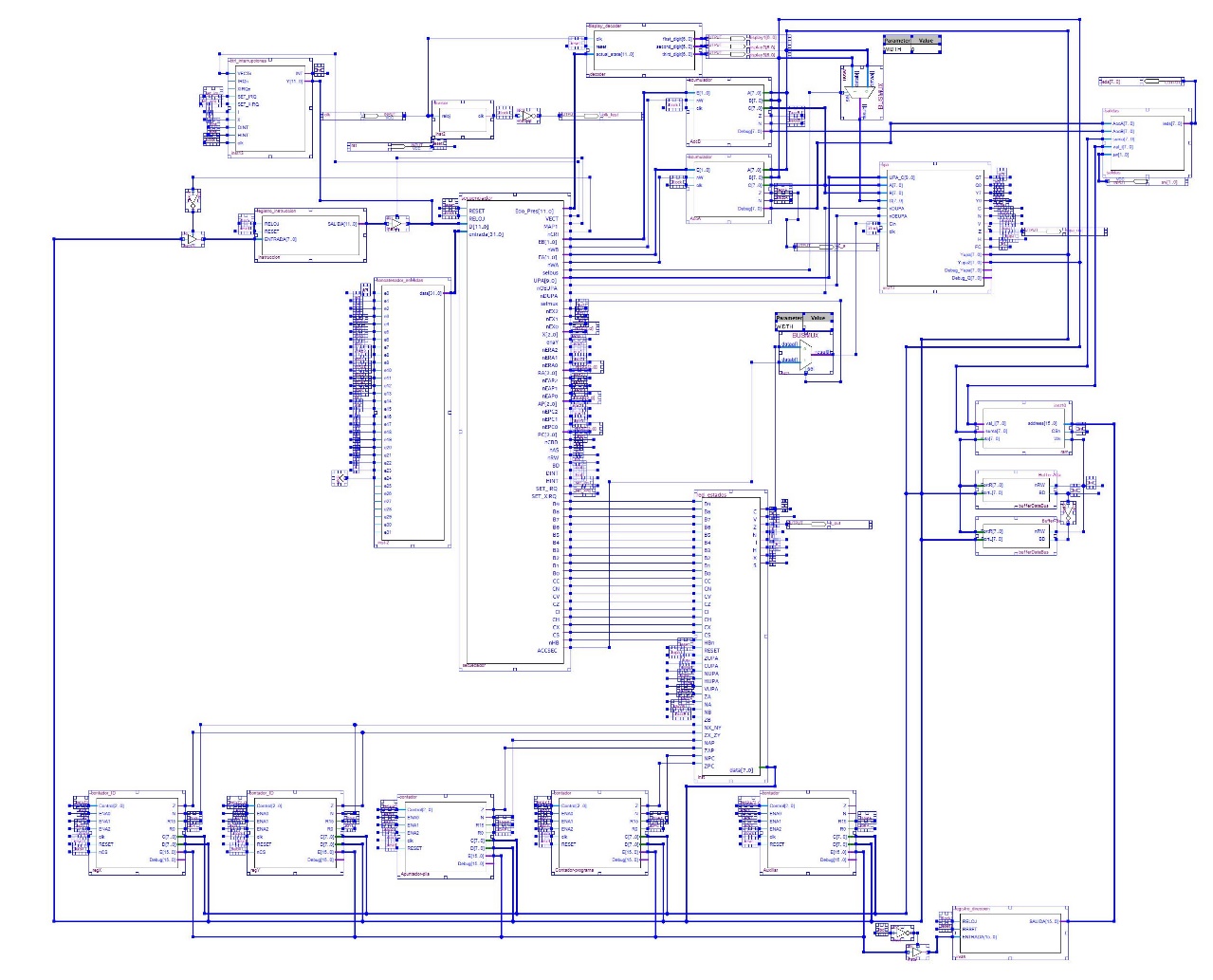


Figura 4. Diagrama de bloques del microprocesador.

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.

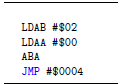
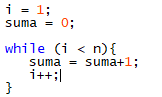
 

Figura 5. Instrucciones a ejecutar Figura 6. Programa a ejecutar

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

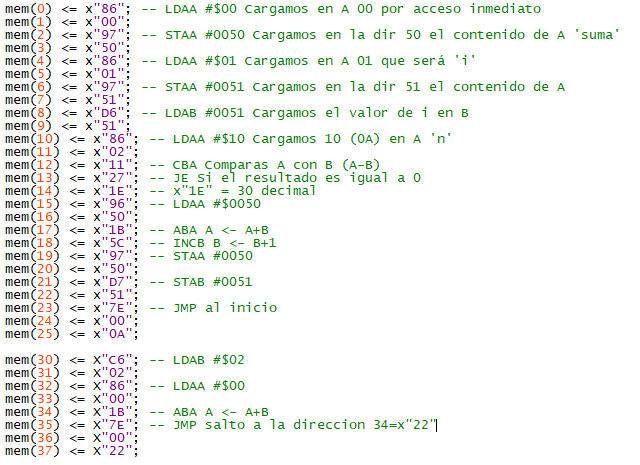
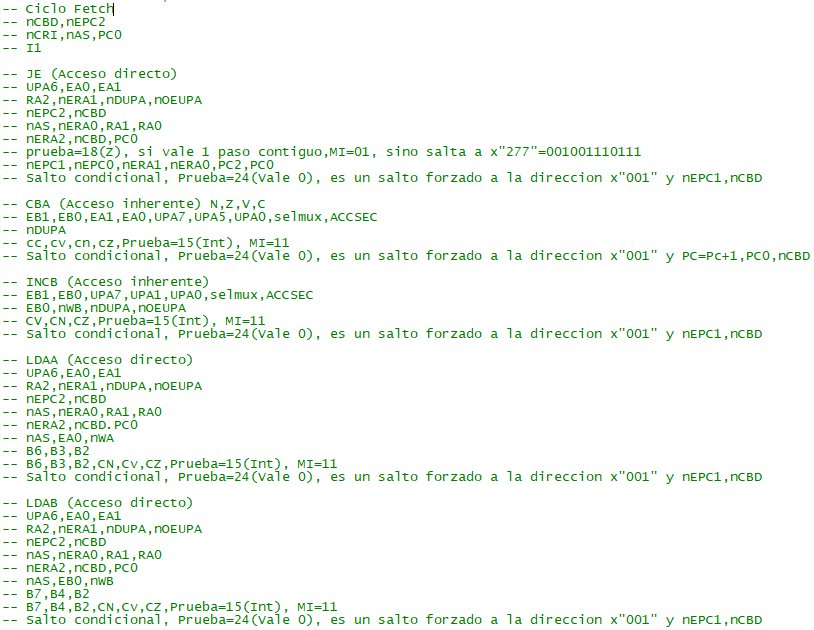


Figura 7. Contenido de la memoria para las instrucciones de prueba y el programa de prueba

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



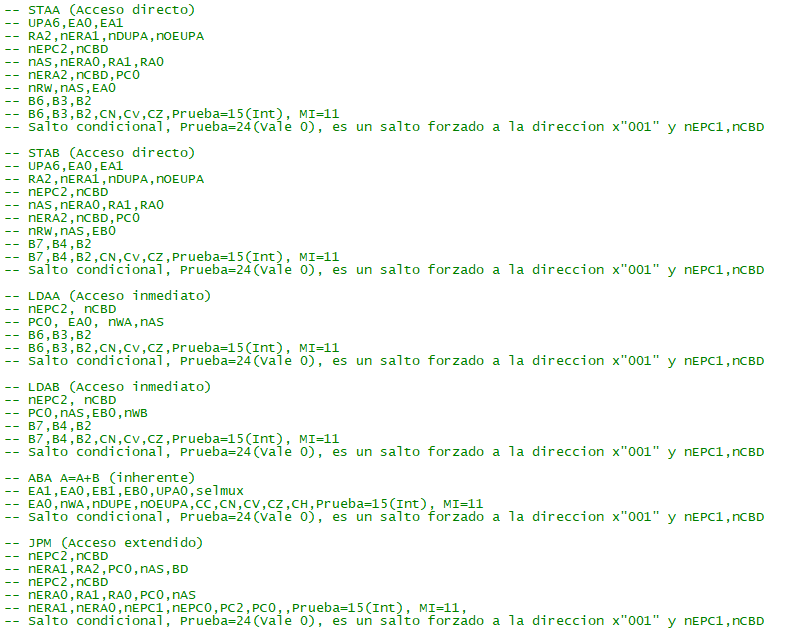
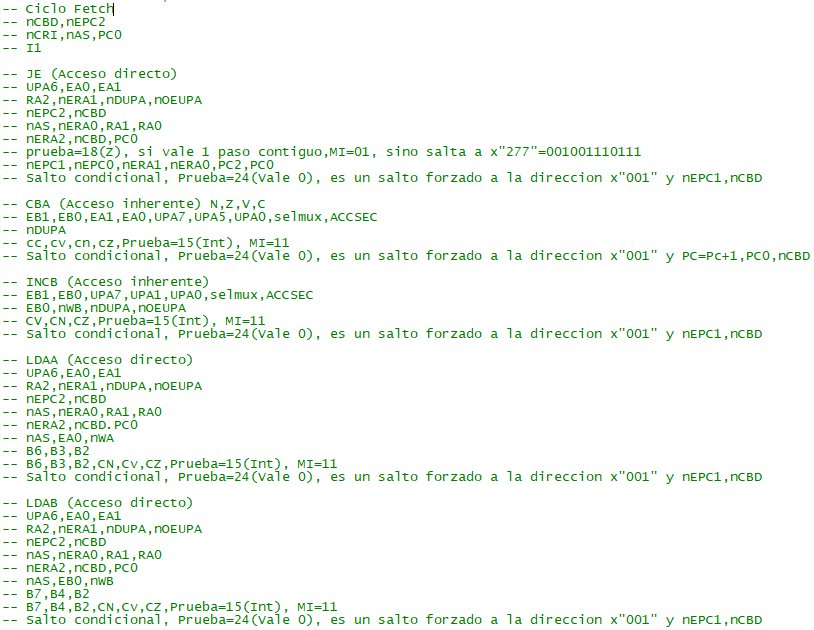
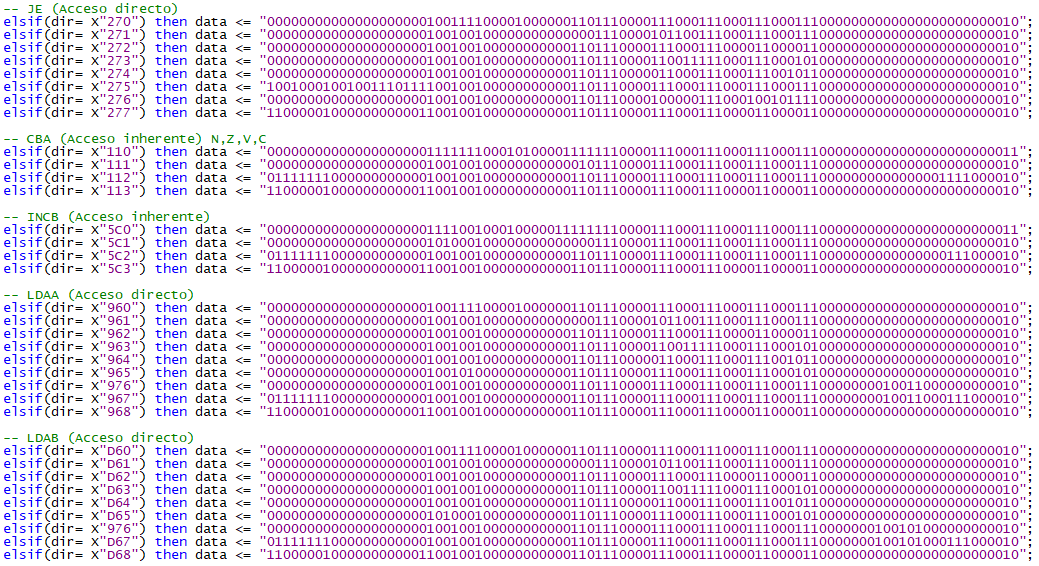


Figura 8. Cartas ASM de las instrucciones

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.





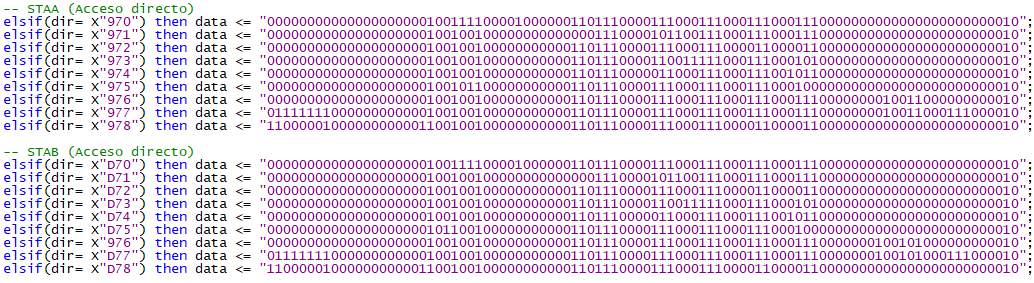


Figura 9. Cartas ASM de las instrucciones necesarias para el ciclo while

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.

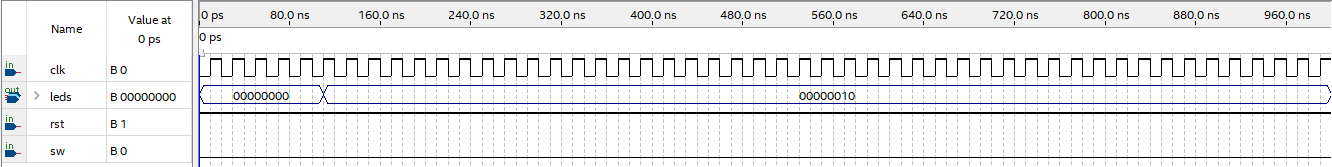


Figura 10. Contenido del acumulador B

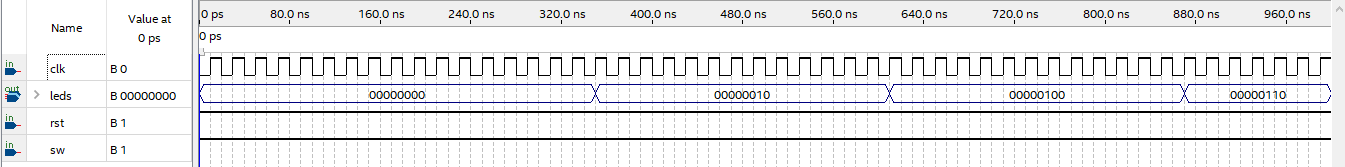


Figura 11. Contenido de la

1. Conclusiones

Guzmán Sánchez José Emmanuel

Para la realización de la práctica fue importante comprender adecuadamente el funcionamiento del secuenciador, ya que de esto dependía toda la arquitectura del procesador y de las instrucciones que se ejecutan en el mismo. De igual manera, un parte fundamental es entender las cartas ASM de las instrucciones, ya que éstas nos ayudarán para poder saber qué señales de control son necesarias activar en cada estado, además, los modos de acceso fueron una parte muy importante, ya que de esto dependía el flujo de la información.

Mejia Ortiz Aarón Enrique

El secuenciador es el elemento más importante de un procesador, ya que es el que controla el resto de componentes que habilitan la ejecución de problemas e interacción con datos. Fue muy importante entender el funcionamiento para poder conectar todos los componentes, y sobre todo, para poder elaborar las cartas ASM y llenar el contenido de memoria del secuenciador. Construir el procesador es un proceso sumamente complicado, no por el contenido en sí, si no por el manejo de todas las conexiones, incluso tomando en cuenta que sí contábamos con toda la lógica de cada componente.

Sáenz Barragán Ricardo

Creando el procesador comprendimos la importancia del secuenciador, con el logramos controlar los demás componentes y nos permitió la interacción con los datos. Al crear un procesador tipo CISC aprendimos como se ejecutan las instrucciones y que elementos de hardware trabajan para crear lo que conocemos como software. Nos dimos cuenta de las ventajas que puede llevar la arquitectura CISC, ya que las instrucciones no interfieren con otras, pero al solo poder ejecutar una instrucción a la vez puede llegar a ser algo lento, pero al mismo sabemos que las instrucciones se van a ejecutar de forma confiable. Realizar las cartas ASM nos dio una facilidad para comprender que hace el procesador y que elementos se comunican con otros para realizar las instrucciones.