



CI-0120 Arquitectura de computadores

Grupo 1

Francisco Arroyo

Enunciado del proyecto

Fecha de entrega: Viernes 2021/Jul/23 al final del día

Modalidad: grupos de dos personas

Construir un circuito en “logisim” para emular una CPU MIPS

Descripción

El proyecto involucra el desarrollo de una CPU utilizando el emulador de circuitos “logisim”. Esta CPU debe ser capaz de ejecutar instrucciones MIPS que estarán almacenadas en su memoria interna. Para los diferentes elementos intentaremos desarrollar un documento de diseño que establecerá lo que tiene **que** hacer ese elemento, luego desarrollaremos el circuito que especifica **como** lo hicimos.

Para el diseño determinaremos las entradas y las salidas del componente involucrado, los elementos necesarios para establecer su funcionamiento, clasificaremos cuáles de esos elementos corresponden a “datapath” y cuáles a “control”. Este diseño será presentado como un documento aparte.

Componentes de la CPU

Listamos algunos de los componentes que debe desarrollar para completar su CPU. Algunos de ellos los construiremos como tareas o exámenes cortos durante el desarrollo del curso. Algunos componentes los podremos utilizar directamente de los provistos por “logisim”, sin embargo, para los otros tendremos que elaborarlos más detalladamente. Conforme avancemos con el curso, tendremos descripciones más detalladas de los componentes a desarrollar.

- Memoria de instrucciones y registro para contador del programa (PC)
- Memoria de datos
- “Register File”
- ALU
- Elementos para la ejecución en “pipeline” MIPS
 - Etapas: IF, ID, EXE, MEM, WB
 - Registros para almacenamiento del contexto de cada instrucción ejecutada
 - Elementos de control para cada etapa del “pipeline”



- Unidad de control
- Unidad de “hazard” y “forwarding”

El “*Register File*” debe ser capaz de manejar los 32 registros de enteros de 32 bits cada uno. El registro cero siempre contendrá el valor cero y no será posible modificarlo, aunque si es válido en el ensamblador elegir este registro como destino. Será posible leer los datos de cualquier registro y de escribir en ellos en un solo pulso de reloj. Es posible que necesitemos que los registros puedan actualizarse en la “primera mitad” de un ciclo de reloj, para que el resultado de una modificación esté disponible al finalizar ese ciclo.

Nuestra CPU debe tener elementos de memoria para almacenar las *instrucciones* y los *datos* necesarios para completar las tareas solicitadas.

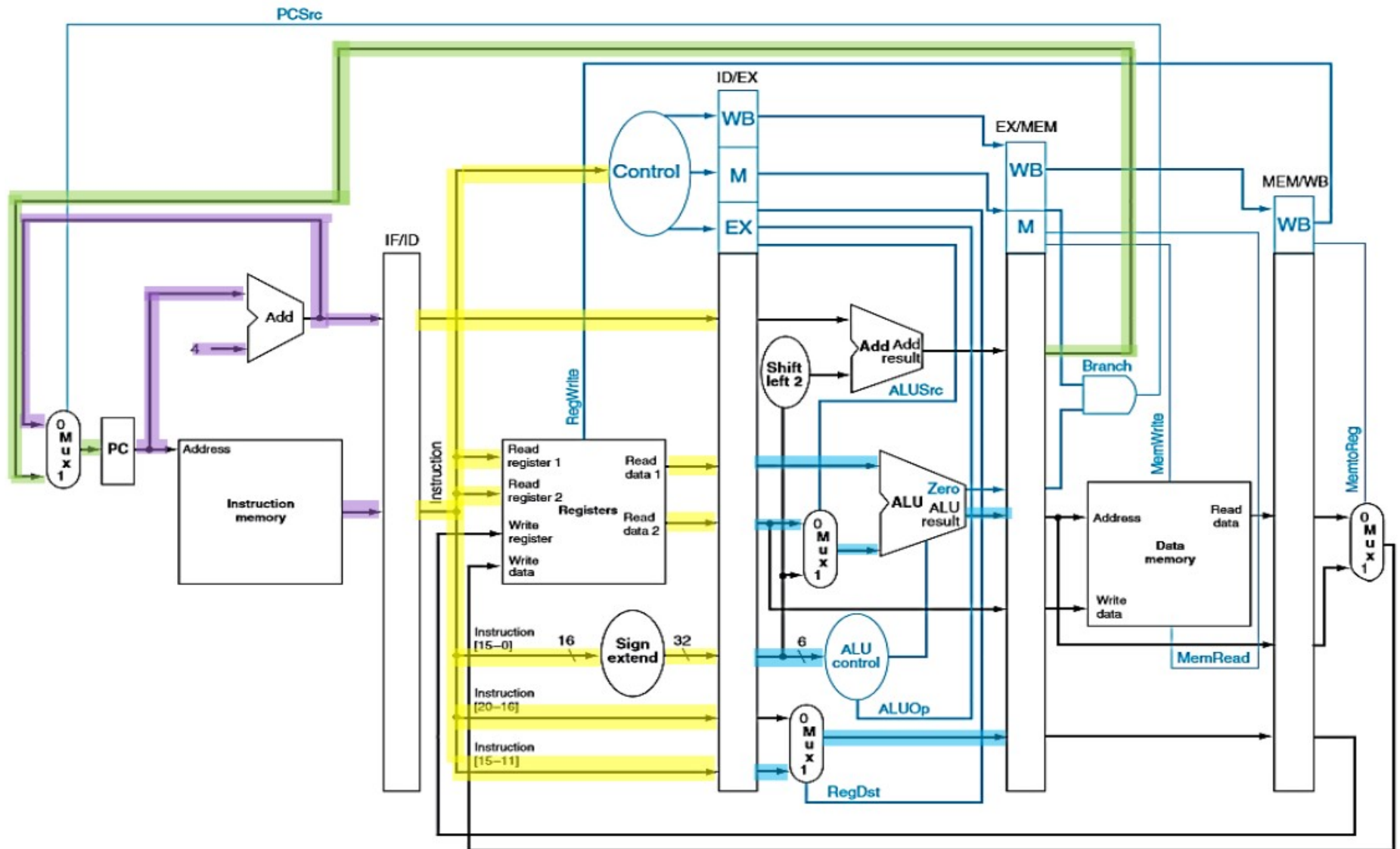
Para el *ALU*, tendremos un documento de especificación detallado con las operaciones necesarias para nuestra CPU. Este ALU tendrá dos entradas con datos de 32 bits que pueden provenir de distintas fuentes, tales como registros o valores incrustados en las instrucciones a ejecutar y una entrada de control que especifica la operación a completar. Entre sus salidas contamos con el resultado de la operación que realiza de 32 bits y banderas que indican el valor del resultado, por ejemplo, si fue igual a cero, mayor que cero, etc.

Elementos de “*pipeline*”, incluye todos los componentes para poder ejecutar partes de las instrucciones de manera simultánea. Unidades adicionales para evitar conflictos estructurales, registros para almacenamiento del contexto, bandera de control para los elementos en cada una de las etapas. Utilizaremos el procesador MIPS para establecer las etapas necesarias para completar las instrucciones, en total cinco: IF, ID, EXE, MEM, WB.

La *unidad de control* es la encargada de dictar cómo ocurren las cosas en nuestro procesador, debe atender a todos los elementos de control de nuestra CPU. Especifica como fluyen los datos a través de la CPU operando compuertas (multiplexores), además dirige a nuestra ALU para que realice la operación adecuada de acuerdo con la instrucción que queremos ejecutar.

La *unidad de “hazard”* nos ayudará a resolver distintos conflictos que puede enfrentar nuestra CPU al ejecutar instrucciones que presentan dependencias de datos.

CPU de ejemplo





Referencias

- Single cycle (filminas)
- Multi cycle (filminas)
- Pipeline (filminas)
- Pipelining: datapath and control (filminas)
- Pipelining hazards (filminas)
- Computer architecture de Hennessy