

# Estructuras de Computadores – (34010)

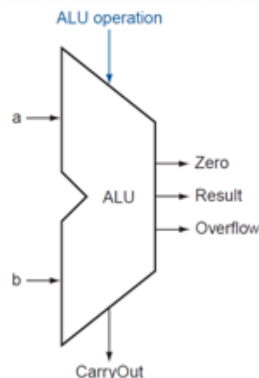
Examen (6 de Junio de 2014)

## Pregunta 1.

(2.5 puntos)

Diseñar la ALU del computador MIPS capaz de implementar las operaciones dadas en la tabla siguiente :

| Instr. | ALUOp | funct  | fun. ALU     | ControlALU |
|--------|-------|--------|--------------|------------|
| lw     | 00    | XXXXXX | suma         | 010        |
| sw     | 00    | XXXXXX | suma         | 010        |
| beq    | 01    | XXXXXX | resta        | 110        |
| add    | 10    | 100000 | suma         | 010        |
| sub    | 10    | 100010 | resta        | 110        |
| and    | 10    | 100100 | AND          | 000        |
| or     | 10    | 100101 | OR           | 001        |
| slt    | 10    | 101010 | activar si < | 111        |



Suponer que la señal “ALU operation” es de 3 bits y viene dada por la columna “ControlALU”. Utilizar sumadores completos de 1 bit, puertas lógicas y multiplexores. Calcular el tiempo de retardo máximo de la ALU suponiendo que el retardo de las puertas lógicas es de T.

## Pregunta 2.

(2.5 puntos)

Un computador formado por un microprocesador presenta las siguientes características: direcciona 64 Kbytes; dispone de un bus de datos de 8 bits. Tiene la memoria distribuida de la siguiente forma: las primeras 4K posiciones corresponden a una memoria ROM; luego hay un hueco de 4K posiciones; después hay 8K posiciones para RAM-VIDEO; a continuación, 32K posiciones para RAM convencional; y finalmente, el resto de posiciones es no direccionable.

Se pide:

- Mapa de memoria especificando los bits de dirección, las direcciones de inicio y fin de cada bloque en hexadecimal, su capacidad y tipo (ROM, RAM-VIDEO, RAM, HUECO, NO DIRECCIONABLE). (1 punto)
- Diagrama de conexiones del sistema de memoria y del procesador. (1.5 puntos)

Consideraciones a tener en cuenta:

- Se dispone de EPROM 2Kx4 y RAM de 8Kx8.
- Emplear el menor número de módulos de memoria.
- Facilidad en posibles ampliaciones futuras de memoria y de añadir más conmutadores y displays solamente.
- Las conexiones de los módulos de memoria son: EPROM (bus de direcciones, el bus de datos,  $\overline{CE}$ ,  $\overline{OE}$ ), RAM (bus de direcciones, el bus de datos,  $\overline{CS}$ ,  $\overline{OE}$ ,  $\overline{WE}$ ).
- Las conexiones del microprocesador son: el bus de direcciones, el bus de datos, MemRead y MemWrite
- Los decodificadores, codificadores, multiplexores, demultiplexores tienen una señal de selección  $\overline{CS}$ .



### Pregunta 3.

(2.5 puntos)

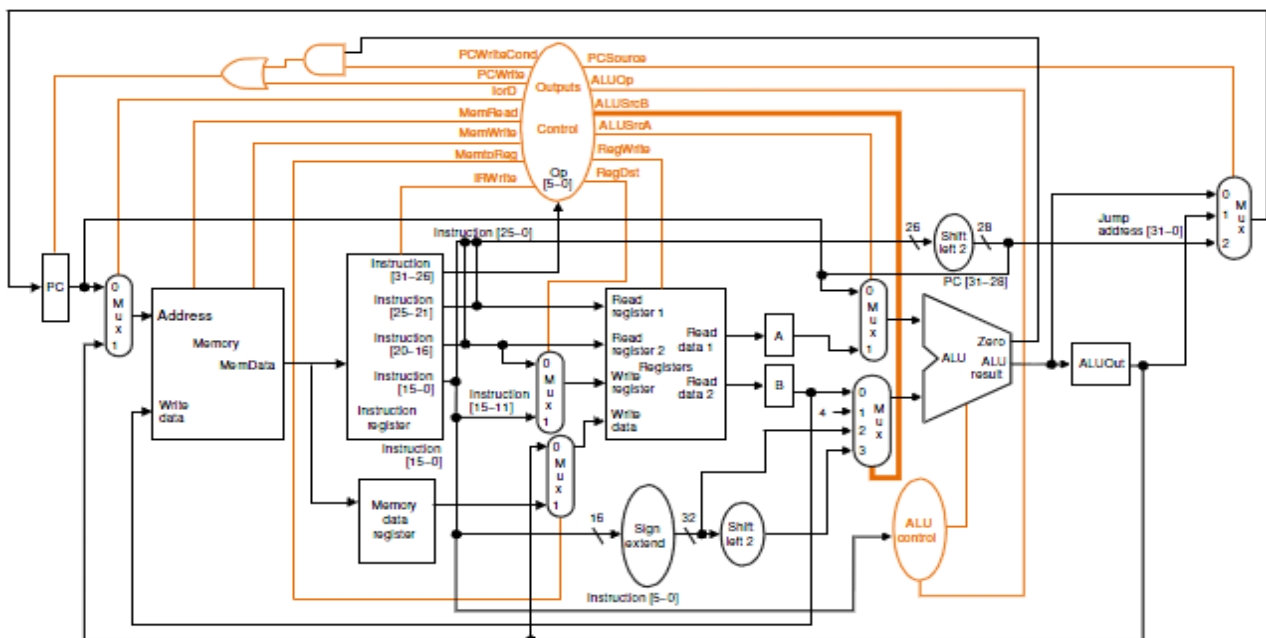
En el esquema de la máquina multiciclo cuya ruta de datos y señales de control es la mostrada en la figura, se quieren añadir las instrucciones siguientes:

- `mult rs, rt`. Multiplica el contenido de los registros especificados en los campos `rs` y `rt` y almacena el resultado de 64 bits en los registros HI y LO.
- `mflo rd`. Copia el contenido del registro LO en el registro indicado en el campo `rd` de la instrucción
- `mfhi rd`. Copia el contenido del registro HI en el registro indicado en el campo `rd` de la instrucción.

Se pide:

- Modificar en el esquema el camino de datos del MIPS para que puedan ejecutarse dichas instrucciones.
- Rellenar la siguiente tabla.

| INSTRUCCIÓN | Nº DE FASES | OPERACIONES EN LENGUAJE RTL | SEÑALES DE CONTROL |
|-------------|-------------|-----------------------------|--------------------|
| mult        |             |                             |                    |
| mflo        |             |                             |                    |
| mfhi        |             |                             |                    |



### Pregunta 4.

**(2.5 puntos)**

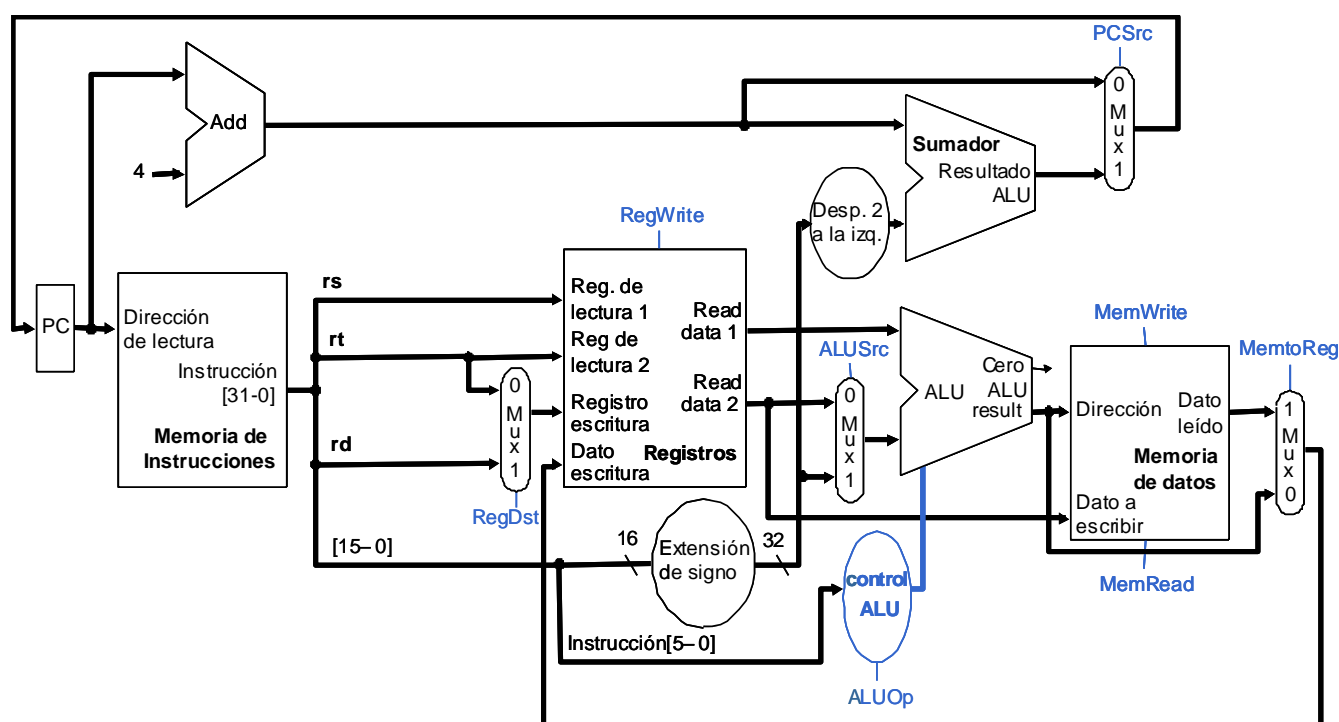
(1.2 punto) Se decide simplificar el repertorio de instrucciones del MIPS cambiando las instrucciones lw y sw por otras sin desplazamiento. Estas nuevas instrucciones tienen las siguiente formas generales:

Lw rt, rs    #rt← M[rs]

$$\text{Sw rt, rs} \quad \#M[\text{rs}] \leftarrow \text{rt}$$

Estas instrucciones son de tipo I pero no se utiliza el campo desplazamiento.

- a) Muestra los cambios que se deben hacer en la ruta de datos monociclo siguiente para evitar que estas nuevas instrucciones lw y sw pasen a través de la ALU. (intenta dibujar los cambios en la figura de la forma más clara posible) (0.4 puntos)



- b) Rellena la tabla con el valor de las señales de control para que se ejecuten correctamente las nuevas instrucciones lw y sw. (utilizar X cuando sea posible) (0.4 puntos).

|    | RegDst | RegWrite | AluSrc | ALUOp | MemWrite | MemRead | MemToReg | PCSrc |
|----|--------|----------|--------|-------|----------|---------|----------|-------|
| Lw |        |          |        |       |          |         |          |       |
| sw |        |          |        |       |          |         |          |       |

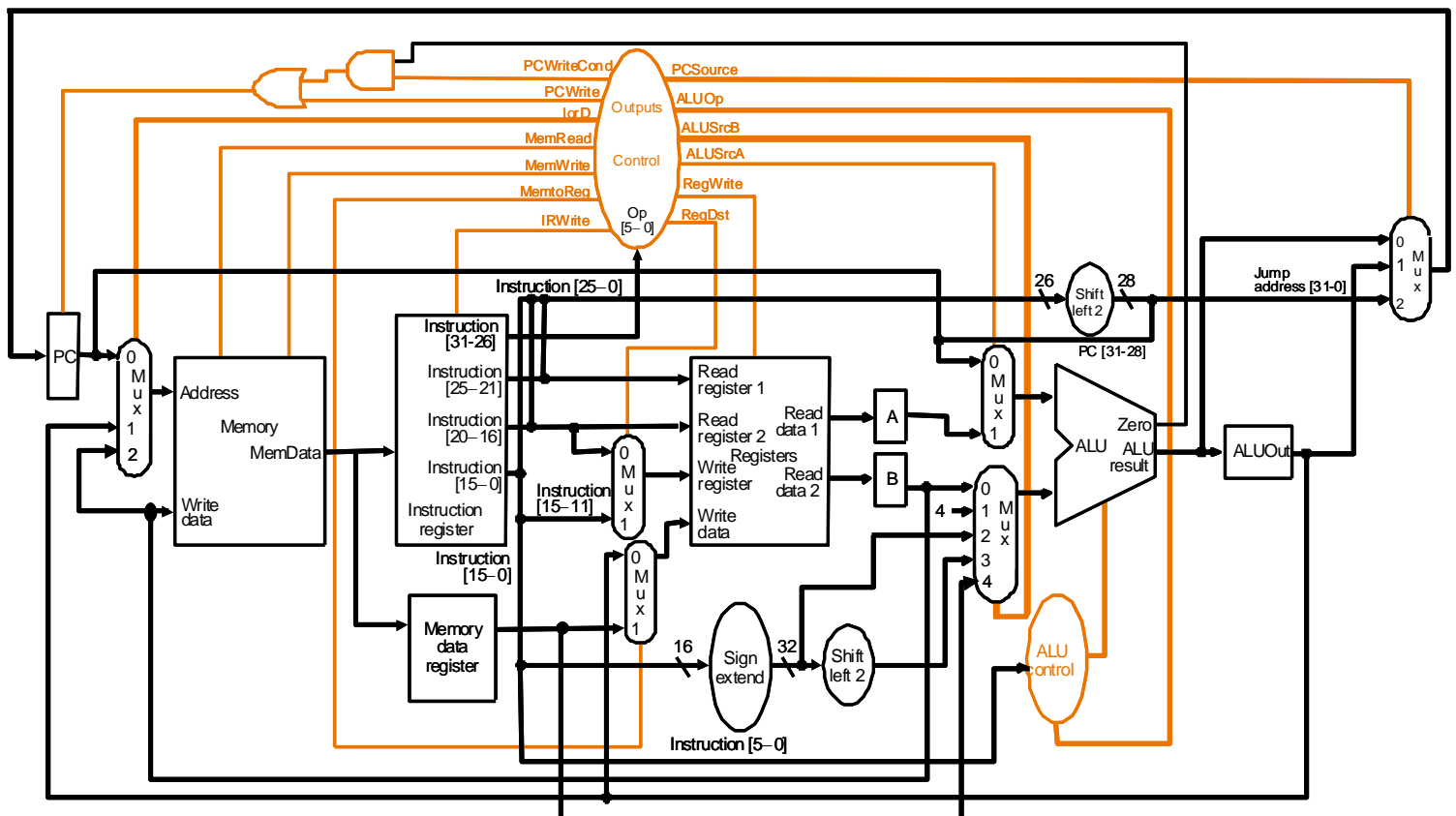
- c) Supón que las memorias y la ALU tienen un retardo de 2ns y el Banco de Registros un retardo de 1ns. Calcula la duración del ciclo de reloj para nueva la ruta de datos que has modificado. (0.4 puntos)



(1.3 puntos) En la ruta de datos multiciclo del MIPS de la figura se han realizado unas pequeñas modificaciones respecto a la ruta de datos estudiada en clase. Esta nueva ruta de datos permite ejecutar también una instrucción suma del tipo registro-memoria:

Addm rd, rs, rt     $\#rd \leftarrow rs + M[rt]$

Es decir, el registro rt contiene la dirección de memoria de uno de los operando de la ALU para la suma. El formato de la instrucción es de tipo R (los campos shamt y función no se utilizan).



- Obtén las acciones a realizar en cada ciclo de reloj mediante lenguaje de transferencia de registros (por ejemplo:  $PC \leftarrow PC + 4$ ) (0.6 puntos).
- Rellena la tabla de salida siguiente donde se muestren el valor de las señales de control en cada ciclo de reloj. (rellena tantas columnas como ciclos de reloj necesita la instrucción para ejecutarse) (0.7 puntos).



| Señales Control | Ciclo de reloj |   |   |   |   |   |   |   |   |    |
|-----------------|----------------|---|---|---|---|---|---|---|---|----|
|                 | 1              | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| PCWrite         |                |   |   |   |   |   |   |   |   |    |
| PCWriteCond     |                |   |   |   |   |   |   |   |   |    |
| IorD            |                |   |   |   |   |   |   |   |   |    |
| MemRead         |                |   |   |   |   |   |   |   |   |    |
| MemWrite        |                |   |   |   |   |   |   |   |   |    |
| IRWrite         |                |   |   |   |   |   |   |   |   |    |
| MemtoReg        |                |   |   |   |   |   |   |   |   |    |
| PCSource        |                |   |   |   |   |   |   |   |   |    |
| ALUOp           |                |   |   |   |   |   |   |   |   |    |
| ALUSrcB         |                |   |   |   |   |   |   |   |   |    |
| ALUSrcA         |                |   |   |   |   |   |   |   |   |    |
| RegWrite        |                |   |   |   |   |   |   |   |   |    |
| RegDst          |                |   |   |   |   |   |   |   |   |    |