

ARQUITECTURA DE COMPUTADORES

543.426

Tarea No. 3
7 de mayo de 2017

Esta tarea contiene problemas sobre procesadores unicycle y multiciclos. Entregar un informe escrito (en computador) con los problemas resueltos. El informe tiene que contener los esquemas de los procesadores modificados resaltando los cambios realizados. Se adjuntan diagramas y tablas de referencia para la resolución de los ejercicios. Plazo máximo de entrega: **Lunes 8 de Mayo hasta las 18 horas** en Secretaría de Electrónica. No se corregirán tareas atrasadas.

Se debe trabajar en grupos de 2 personas. Grupos de 1 persona son permitidos pero no recomendados. No se permitirán grupos de más de 2 personas. Se les recuerda que cualquier copia (entre tareas o de fuentes externas) resultarán en calificación 1 para todas las tareas involucradas.

Procesador Unicycle

Problema 1

Se desea agregar una nueva instrucción 'memim' cuya descripción RTL es:

```
1 | R[rs] <- Mem[R[rt]]  
2 | Mem[R[rt]] <- R[rt] - Sext(Imm16)
```

Esta instrucción lee una palabra de memoria desde la dirección dada por el registro \$rt y la guarda en el registro \$rs, además guarda en la misma dirección de memoria el valor de la diferencia entre el mismo registro y la constante inmediata de 16 bits, extendida en signo.

Indique modificaciones necesarias a la sección de datos y a la tabla de señales de control utilizando diagrama adjunto. Realice el mínimo de modificaciones necesarias para implementar esta instrucción.

Problema 2

Se desea agregar una nueva instrucción 'mem2reg' cuya descripción RTL es:

```
1 | R[rt+Zext(Imm16)] <- Mem[R[rt]+Zext(Imm16)]
```

Esta instrucción lee una palabra de memoria desde la dirección dada por el registro \$rt más el inmediato extendido en cero. Luego guarda el valor leído desde memoria en la dirección del registro dado por la suma de \$rt y el inmediato extendido en cero.

Indique modificaciones necesarias a la sección de datos y a la tabla de señales de control utilizando diagrama adjunto. Realice el mínimo de modificaciones necesarias para implementar esta instrucción.

Procesador Multiciclo

Problema 3

Deseamos agregar una nueva instrucción (mswap) al procesador multiciclo, la cual intercambia el contenido de dos posiciones de memoria adyacentes.

La instrucción utiliza un formato tipo-I. La descripción de transferencia de registros lógica es:

```
1 Mem[R[rs]+Sx(imm16)] <- Mem[R[rs]+Sx(imm16)+4]
2 Mem[R[rs]+Sx(imm16)+4] <- Mem[R[rs]+Sx(imm16)]
```

Implemente la instrucción mswap en un mínimo número de ciclos sin afectar el período de reloj del procesador original y sin agregar nuevas unidades funcionales (sumadores, ALUs, etc). Asuma que la ALU posee una operación que permite sumar 4: $ALUop = +4$.

Debe entregar: una descripción a nivel de transferencia de registros (RTL) física para la implementación, las modificaciones necesarias a la sección de datos para implementar la instrucción y una tabla con el valor asumido por las señales de control para cada ciclo de ejecución de la instrucción.

Problema 4

Se desea agregar una instrucción (vecsw) de guardado iterativo en memoria al procesador multiciclo MIPS. La instrucción utiliza el formato-R de la siguiente manera: *vecsw \$rd, \$rs, \$rt*

Esta instrucción guarda en memoria lo contenido en el registro \$rt en la dirección apuntada por \$rd y aumenta condicionalmente \$rd a partir de su valor respecto a \$rs

```
1 Mem [R[rd]] <- R[rt]
2 IF (R[rd] + 4 <= R[rs]) R[rd] <- R[rd]+4
```

Lo cual permitiría reemplazar un código en assembly con la siguiente iteración, donde el label CONT marca el código que sigue a continuación del loop:

```
1 LOOP: sw    $t2, 0($t0)
2         addi $t3, $t0, 4
3         bgt  $t3, $t1, CONT
4         move $t0, $t3
5         b    LOOP
6 CONT: ...
```

Simplemente por las instrucciones:

```
1 LOOP: vecsw $t0, $t1, $t2
2         bge  $t0, $t1, CONT
3         b    LOOP
4 CONT: ...
```

Agregue la instrucción, llamada 'vecsw' al procesador MIPS multiciclo. Minimice la cantidad de ciclos de ejecución de la instrucción. Puede agregar multiplexores y cables, pero no registros ni sumadores.

Debe entregar: una descripción a nivel de transferencia de registros (RTL) física para la implementación, las modificaciones necesarias a la sección de datos para implementar la instrucción y una tabla con el valor asumido por las señales de control para cada ciclo de ejecución de la instrucción.

Diagramas

Nota: Las tablas adjuntas son una referencia general para resolver los problemas. De ser necesario deberá agregar y/o quitar filas y/o columnas de las tablas para la correcta implementación de las señales de control.

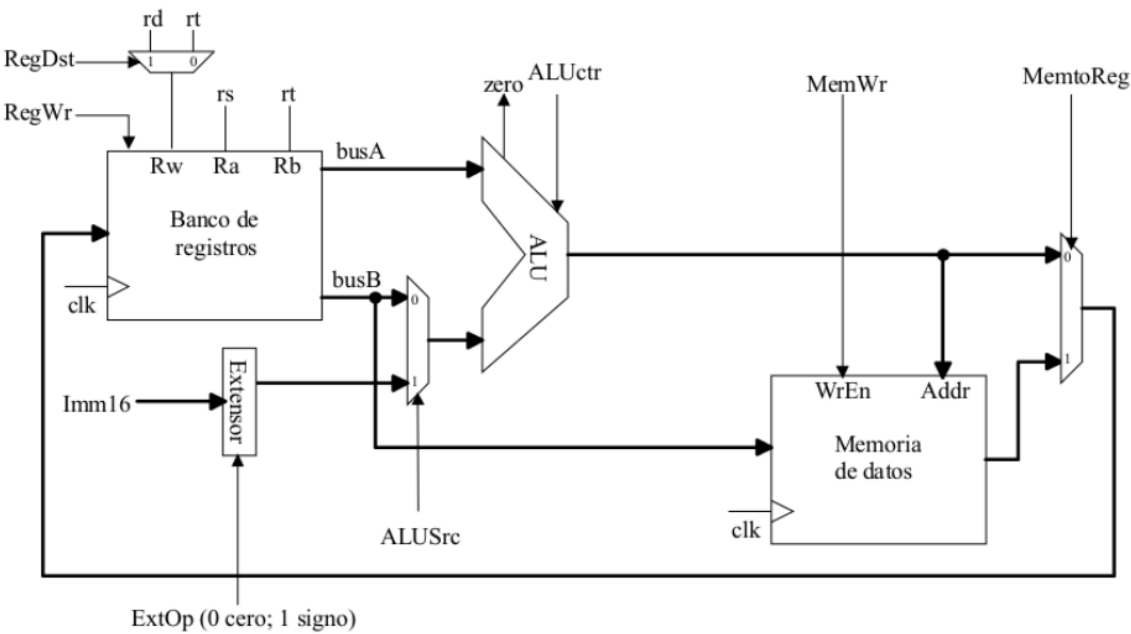


Figura 1: Procesador Uniciclo

	R-type	ori	andi	lw	sw	beq		
RegDst	1	0	0	0	x	x		
ALUSrc	0	1	1	1	1	0		
MemtoReg	0	0	0	1	x	x		
RegWr	1	1	1	1	0	0		
MemWr	0	0	0	0	1	0		
nPC_sel	0	0	0	0	0	1		
ExtOp	x	0	0	1	1	x		
ALUOp	func	or	and	add	add	sub		

Cuadro 1: Señales de control proc. uniciclo

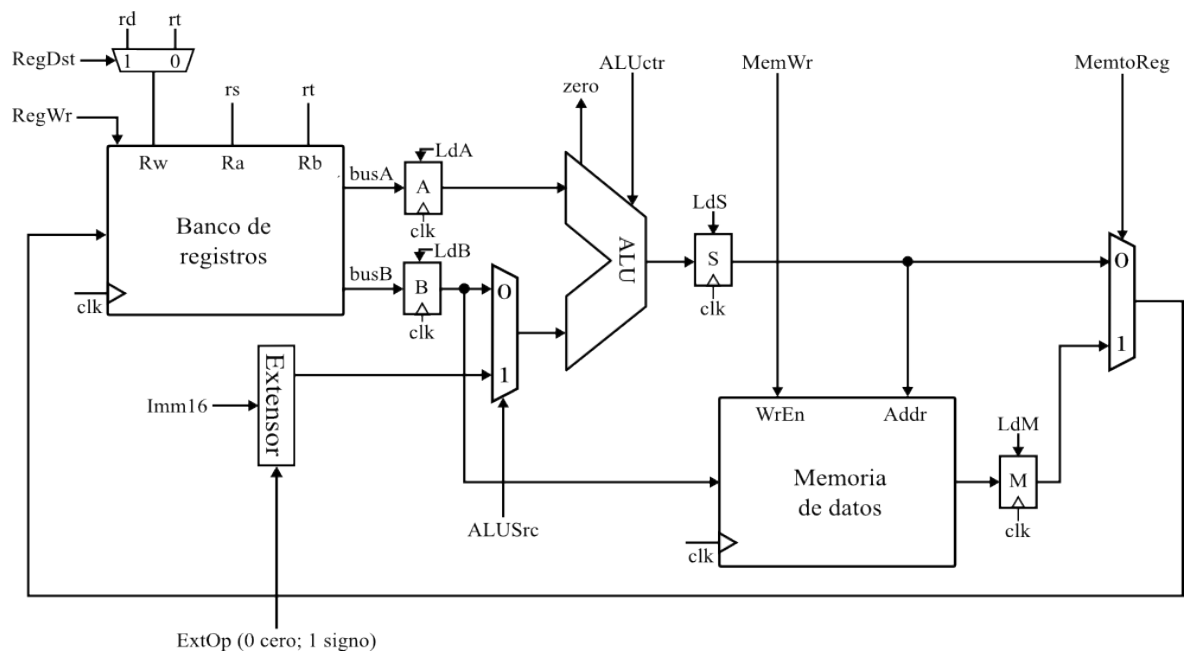


Figura 2: Procesador Multiciclo

	(1)	(2)	(3)	(4)	(5)
LdIR					
RegDst					
RegWr					
LdA					
LdB					
ExtOp					
ALUSrc					
ALUctr					
LdS					
MemWr					
LdM					
MemtoReg					

Cuadro 2: Señales de control por ciclo de instrucción