

Auxiliar 9

Profesor: Pablo Guerrero.
Auxiliar: Ian Yon
Viernes 17 de octubre de 2014



Problema 1: instrucciones M32

A continuación se indican 8 conjuntos de transferencias entre registros para el diseño actual de M32. Considerando cada uno de estos conjuntos en forma independiente, señale cuáles de ellos se pueden llevar a cabo en un solo ciclo del reloj y cuáles no. Si un conjunto de transferencias es válido indique las señales de control necesarias para realizarlas. Cuando un conjunto no es válido explique porque (no necesita dar las señales de control).

1. $AR = PC ; PC = PC+4$
2. $PC = 4 ; AR = 4$
3. $R1 = PC$ cuando IR contiene la instrucción ADD R2, R3, R1 ($R1 = R2+R3$)
4. $PC = R1 + R3$ cuando IR contiene ADD R2, R3, R1
5. $R1 = PC + R2$ cuando IR contiene ADD R2, R3, R1
6. $IR = R2 + R3$ cuando IR contiene ADD R2, R3, R1
7. $PC = Mem[AR]$
8. $R1 = R2 + R3; IR = Mem[AR]$ cuando IR contiene ADD R2, R3, R1

Problema 2: extensión M32

Se desea agregar la instrucción LDRPC a M32 (load relative to pc). Esta instrucción lee una palabra de 32 bits en la memoria suministrando un desplazamiento relativo al registro PC, es decir:

Assembler	Instrucción	Formato instrucción		
		31-24	23-19	18-0
LDRPC imm, r_i	$r_i \leftarrow Mem[PC + imm]$	Instrucción	r_i	imm

No es posible implementar LDRPC con el diseño físico actual, puesto que no se puede llevar el desplazamiento de 19 bits a la ALU.

1. Modifique e implemente el módulo Y-SEL para que esta instrucción sea implementable. Su modificación debe ser tal que las actuales instrucciones de M32 continúen siendo implementables.
2. Indique ciclo por ciclo las transferencias entre registros y las señales de control necesarias para ejecutar LDRPC. No incluya la fase de fetch ni la fase de decodificación.

Problema 3: Entrada - Salida

Ud. dispone del teclado de la Figura 1, que está organizado como una matriz de 16 columnas de 6 teclas cada una. Para leer el estado de una columna completa (6 teclas), se suministra

como entrada en C3 – C0 el número de la columna. El teclado entrega en K5 – K0 el estado de las 6 teclas pertenecientes a la columna. No es posible leer dos columnas al mismo tiempo.

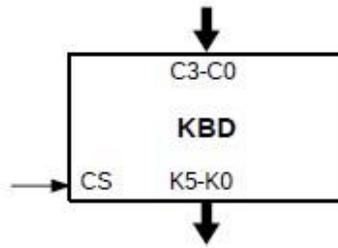


Figura 1: Circuito KBD

1. Construya una interfaz para este teclado considerando un computador con un bus de direcciones de 16 bits y un bus de datos de 8 bits. Su interfaz debe poseer 16 puertos de entrada en las direcciones 0xFF00 a 0xFF0F (en hexadecimal). Cada uno de estos puertos sirve para leer una columna del teclado. No implemente KBD.
2. Programe la rutina *int consultar tecla(int fila, int columna)*, que entrega verdadero si la tecla de coordenadas (fila, col) está presionada.

Problema 4: entrada y salida con interrupciones

En la figura se muestra el dispositivo TIMER, que se programa para que produzca una interrupción al cabo de t ciclos del reloj. Para programarlo se coloca simultáneamente un 1 en la entrada CS, un 1 en WR y la cantidad de ciclos t en D31 – D0.

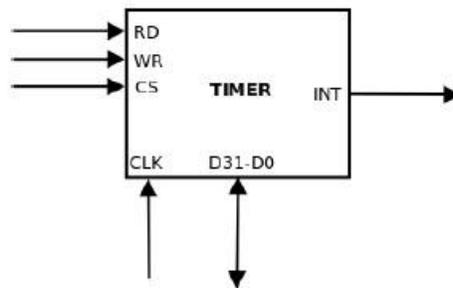


Figure 2: Módulo TIMER

Como se muestra en el diagrama de tiempo, el timer activará la línea INT después de t ciclos del reloj, la que permanecerá en 1 hasta que se desactive el timer. El timer se desactiva colocando un 1 en CS, un 1 en WR y un 0 en D31 – D0. También se puede recuperar la cantidad de ciclos que restan para que se active la interrupción, lo que se logra colocando un 1 en CS y un 1 en RD. La cantidad de ciclos restantes t_0 aparece por D31 - D0 (0 si el timer está desactivado).

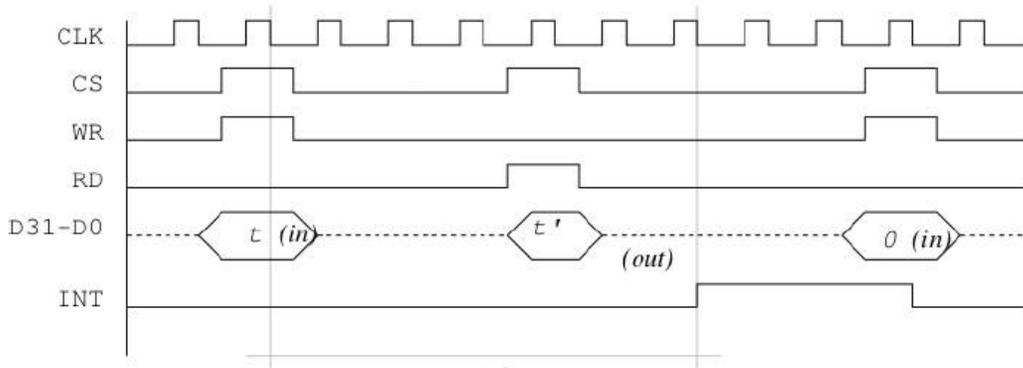


Figure 3: Diagrama de tiempo TIMER

1. Implemente una interfaz que conecte el timer de la Figura con un procesador M32. Haga que cada vez que se escribe un dato t en la dirección $0xffff0000$ se programa el timer para que interrumpa en t ciclos del reloj, si es que t es mayor que 0, o se desactiva el timer, si t es cero. Además haga que cada vez que se lea esa misma dirección se obtiene la cantidad de ciclos que restan para la interrupción.
2. Programe en C los siguientes procedimientos:
 - a. `progTimer(int t, void (* f)())`: programa el timer para que produzca una interrupción en t ciclos del reloj. Además registra el procedimiento f para que se invoque cuando ocurra la interrupción. Si t es 0, se desactiva el timer.
 - b. `handleTimer()`: rutina de atención de la interrupción que desactiva el timer e invoca el procedimiento f .