

Problema 8:

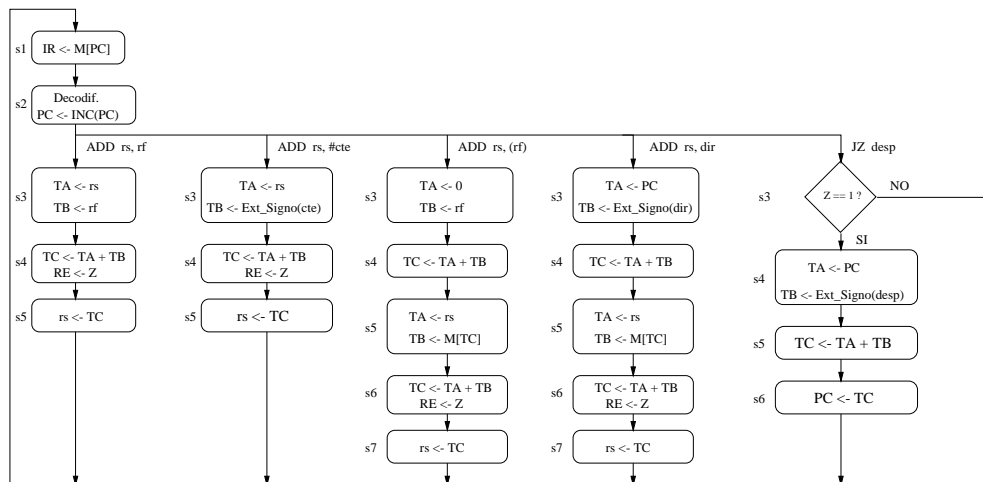
En la siguiente tabla se recoge el formato de algunas de las instrucciones de una máquina hipotética:

Instrucción		Formato de la Instrucción
Mnemónico	Significado	
ADD rs, rf	$rs \leftarrow SUMA(rs, rf)$	00 rs rf XXXX 00
ADD $rs, \# cte$	$rs \leftarrow SUMA(rs, cte)$	00 rs cte 01
ADD $rs, (rf)$	$rs \leftarrow SUMA(rs, M[rf])$	00 rs rf XXXX 10
ADD rs, dir	$rs \leftarrow SUMA(rs, M[PC + dir])$	00 rs dir 11
JZ $desp$	Si $Z == 1$ $PC \leftarrow SUMA(PC, desp)$	10 $desp$

(XX - no usado)

Las instrucciones ocupan siempre una palabra (de 16 bits) en memoria principal. Obsérvese sin embargo, que el número de campos del formato de instrucción varía. En cualquier caso, los dos primeros bits del formato de instrucción representan siempre el código de operación. Los campos rs y rf representan a alguno de los 16 registros del banco de registros. El campo cte representa una constante entera codificada en complemento a 2. Los campos dir y $desp$ representan parte de una dirección efectiva de memoria y también están codificados en complemento a 2. Esa dirección efectiva de memoria se calcula sumando al PC el campo dir o el campo $desp$ (dependiendo de la instrucción, tal y cómo se muestra en la tabla anterior). Además, hemos de hacer notar que en la instrucción ADD los dos últimos bits del formato de instrucción representan el modo de direccionamiento del segundo operando de la suma.

El ciclo de instrucción, para las instrucciones propuestas, se muestra en la siguiente figura:



1. Rellena la siguiente tabla indicando los modos de direccionamiento de los operandos para cada una de las instrucciones.

Modos de Direccionamiento		
Instrucción	Operando fuente1	Operando fuente2
ADD <i>rs, rf</i>		
ADD <i>rs, #cte</i>		
ADD <i>rs, (rf)</i>		
ADD <i>rs, dir</i>		

2. Construir un *datapath* basado en las unidades funcionales que aparecen en la siguiente figura, y que se describen a continuación:
 - Memoria principal de 64Kx16 bits. Se direcciona a través de la entrada *Dirección*. Los datos en lectura/escritura los proporciona/toma por el bus bidireccional *Dato*.
 - Banco de 16 registros de 16 bits. Se pueden direccionar dos registros para lectura direccionando en las entradas *DA* y *DB*, obteniéndose los datos por las salidas *A* y *B*, respectivamente. Simultáneamente se puede escribir el registro direccionado en *DC*, con el valor presente en la entrada *C*.
 - SUM es un sumador de 16 bits. Permite realizar la operación SUMA de los valores presentes en sus entradas. Genera el flag Z.
 - Registros: IR, PC, TA, TB, TC, todos de 16 bits y registro de estado RE, que almacena el flag Z, de 1 bit. La señal de reloj es la mostrada en la figura como CK

Para realizar el diseño, se pueden incluir los buses y la circuitería combinacional (puertas lógicas, multiplexores, etc.) que creas necesaria. Debes especificar claramente, la función de los puntos de control y el tamaño de los buses, que hayas tenido que añadir.

