GUAYMAS MATIAS

PRÁCTICA 4

HOJA Nº 1

FECHA

1) a) DADD r1,r2,r0 : r1 = r2 + r0 → MOV R1, R2

b) DADDi r3,r0,5 : r3 = r0 + 5 → MOV R3, 5

c) DSUB r4,r4,r4 : r4 = r4 - r4 → MOV R4, 0

d) DADDi r5,r5,-1 : r5 = r5 + (-1) → DEC R5

e) XORi r6,r6,0Xffffffffffffffff : R6 XOR 0Xffffffffffffffff

2) a) Cuando forwarding no está habilitado la instrucción SD R2,A(r0) que en la etapa ID trata de leer el contenido de R2, pero dicho contenido no estará disponible hasta que la instrucción anterior LD r2, B(r0) llegue a la etapa de WB. Y debido a esto se genera un atasco en la etapa ID donde se procesa la instrucción SD R2,A(r0) retrasando la salida de esta etapa (con RAWS) a la espera del contenido del registro.

- Aparecen atascos de tipo RAW (Read After Write) causada por una dependencia de datos, en este caso se intenta leer un dato antes que esté guardado en el registro. Serán dos atascos, equivalentes a dos ciclos hasta que LD R2, B(r0) salga de la etapa WB.

-2.2 CPI

b) Con la opción forwarding habilitada el dato contenido en el registro R2 será cedido por la instrucción SD R2, A(r0) cuando la instrucción LD R2, B(r0) se encuentra finalizando la etapa MEM. La instrucción SD R2, A(r0) no tiene que esperar a que la instrucción LD R2, B(r0) salga de la etapa WB. De esta manera no hay atascos RAW

- El color verde en el registro R1 significa que el dato está disponible en etapa MEM para adelantamiento. Además, los registros pueden tener color rojo indicando que el resultado está disponible en EX y puede ser adelantado. Si el color es gris, el valor no está disponible en este ciclo para adelantamiento

-1,8 CPI

3) a)

- Se presentan atascos por dependencia de datos de tipo RAW causado por la instrucción BNEZ R2,loop al procesarse en la etapa ID. Esta instrucción necesita del contenido del registro R2 que está siendo utilizado por la instrucción DADDi R2, R2, -1 en la etapa EX sin salir aún de esta.

- Bis ocurre como consecuencia de la ejecución incorrecta de la instrucción siguiente a una instrucción condicional. Esto se debe a que la condición a evaluar tarda algunos ciclos en ser ejecutada, mientras que durante estos ciclos siguen entrando nuevas instrucciones al cauce. Luego de evaluar la condición si la instrucción posterior a esta que se ejecutó no es la que debía ser ejecutada, su ejecución se trunca y se ejecuta la que está en el lugar de memoria indicada por la etiqueta en la instrucción condicional.

- 21 ciclos, 12 instrucciones y 1,750 CPI

NOTA

b) - DSLL R1,R1,1: Trata de leer el contenido del registro R1, mientras que la instrucción LD R1, A(r0) todavía no copió el contenido de la dirección de memoria A en R1 y permanece aún en la etapa WB (RAW durante 1 ciclo).

- BNEZ R2, loop: Trata de leer el contenido del registro R2, mientras que DADDI R2,R2,-1 está buscando copiar el resultado de la operación en dicho registro, permaneciendo en la etapa MEM y posteriormente en la etapa WB (RAW durante 2 ciclos)

- Los atascos por este durando dos ciclos en cada vuelta del lazo loop, mientras que con forwarding habilitado se reducen a un ciclo por vuelta de lazo.

Esta diferencia tiene su causa en la instrucción condicional que es la que está generando los atascos RAW, entonces al disminuir la cantidad de RAWs provocados por esta, también disminuyen los ciclos de espera de la instrucción siguiente, que ahora se detendrá de ejecutar si la condicional así se le indica al procesador.

- 25 ciclos, 12 instrucciones y 2,083 CPI.

c) PROGRAMA EN PC.

d) PROGRAMA EN PC.

4) a) El programa busca en TABLA un elemento igual al contenido en la dirección de memoria NUM. En este caso dicha coincidencia se produce cuando el contenido del registro R4 es igual al contenido del registro R2 (R4=R2), razón por la cual luego se evaluará esta condición y de resultar verdadera se salta a la posición de memoria indicada por la etiqueta "listo". Cuando hay coincidencia la línea del programa en listo suma al registro R10 un 1, caso contrario el contenido de R10 queda en 0. Este es el resultado y queda almacenado en el registro F10. El registro R3 se utiliza como índice para recorrer la TABLA. El contenido del registro R3 se incrementa de a 8 porque cada elemento de Tabla es tamaño word (8 bytes).

b) Habilitando BTB logramos reducir los atascos Branch Taken Stalls a la mitad. Esta opción es útil cuando aumenta la cantidad de iteraciones de un lazo. Como vemos también esta opción no actúa sobre los atascos por dependencia de datos (RAW en este caso) que no se modifican.

c)

——	FORWARDING HAB.	BTB HAB.
CANT CICLOS	71	67
CPI	1,651	1,558
ATASCOS RAWS	16	16
ATASCOS BRANCH TAKEN	8	4

5) a) Tener habilitada la opción de Delay Slot implica que una instrucción se ejecute después de una instrucción de salto sin importar el resultado del salto. El salto retardado nos garantiza siempre 0 atascos. No siempre funciona y se requiere modificar el código para que siga funcionando.

b) Se incluye un NOP para no romper el funcionamiento del programa y como solución simple al Delay Slot. Si no estuviera el NOP el programa finalizaría a causa de HLT.

NOTA

GUAYMAS MATIAS

HOJA N° 2

FECHA

c) D)

———	Delay Slot con NOP	Delay Slot sin NOP y con Reordenamiento
CANT CICLOS	63	55
CANT INSTRUCCIONES	59	51
CPI	1,068	1,078

6) PROGRAMA EN PC

7) PROGRAMA EN PC

8) PROGRAMA EN PC

9) PROGRAMA EN PC

10) PROGRAMA EN PC