

AIC Practica 2

Pregunta 1:

Calcul de cicles de parada **SENSE DEPENDENCIES**

-La primera instruccio necessita 5 cicles per plenar la ruta.

-4 instruccions despres de la primera abans del bucle

-El bucle te 10 instruccions e itera 16 vegdes. 160

-3 instruccions fora del bucle. 3

$5+4+160+3=172$ cicles

(per a 168 instruccions) $\leftarrow 172 - 4$ de plenar la ruta = 168

Solucionar les dependències: INSERINT CICLES DE PARADA

-2 antes del bucle

BUCLE

-2load r12

-2 load r14

-2 sd r14

-2 seq r5

-3 riesgo control beqz

$2+ 16*(2+2+2+2+3)=178$ parada

$172+178=350$ cicles execucio total

start:

dadd r1, \$gp, x

dadd r4, r1, #128 ; $16*8$

dadd r2, \$gp, y

dadd r3, \$gp, z

ld r10, a(\$gp)

loop:

ld r12, 0(r1)

dadd r12, r10, r12

ld r14, 0(r2)

dadd r14, r12, r14

sd r14, 0(r3)

dadd r1, r1, #8

dadd r2, r2, #8

dadd r3, r3, #8

seq r5, r4, r1

beqz r5, loop

nop

nop

trap #0

; Fin de programa

Pregunta 2:

Partim del calcul de cicles de parada **sense dependències**, que igual que abans sera 172.

Utilitzar predict-not-taken

-2 antes del bucle

BUCLE

-2load r12

-2 load r14

-2 sd r14

-2 seq r5

-3 riesgo control beqz

$2+ 16*(2+2+2+2+3)=178$ parada

Restar 3 per el predict not taken de la ultima iteracio

$172+178 -3 = 347$ cicles en total de execucio

Pregunta 3:

Partim del calcul de cicles de parada **sense dependències**, que igual que abans sera 172.

Solucionar les dependències: EMPRANT CURTSCIRCUITS + Predict not taken 3

-0 antes del bucle

BUCLE

-1load r12 (utilitzar un registre despres de un load requereix 1 cicle parada)

-1 load r14 (utilitzar un registre despres de un load requereix 1 cicle parada)

-0 sd r14

-0 seq r5

-3 riesgo control beqz

$16 \cdot (2+3) = 80$ cicles parada

Restar 3 per el predict not taken de la ultima iteracio

$172 + 80 - 3 = 249$

MODIFICACIONES DEL CODI axpy-p3

Pregunta 1:

Partim del calcul de cicles de parada **sense dependències**, que igual que abans sera 172.

Solucionar les dependències: EMPRANT CURTSCIRCUITS + Predict not taken 3

-0 antes del bucle

BUCLE

-0load r12

-0 load r14

-0 sd r14

-0 seq r5

-3 riesgo control beqz

$16 \cdot (3) = 48$ cicles parada

Restar 3 per el predict not taken de la ultima iteracio

$172 + 48 - 3 = 217$

```
18 start:
19     dadd r1,$gp,x
20     dadd r4,r1,#128
21     dadd r2,$gp,y
22     dadd r3,$gp,z
23     ld r10,a($gp)
24
25 loop:
26     ld r12,0(r1)
27     ld r14,0(r2)
28     dadd r12,r10,r12
29     dadd r14,r12,r14
30     sd r14,0(r3)
31     dadd r1,r1,#8
32     dadd r2,r2,#8
33     dadd r3,r3,#8
34     seq r5,r4,r1
35     beqz r5,loop
36     nop
37     nop
38     trap #0 ; Fin de programa
39
```

Pregunta 2:

Partim del calcul de cicles de parada **sense dependències**, que igual que abans sera 172.

Solucionar les dependències: EMPRANT CURTSCIRCUITS + Predict not taken 1

-0 antes del bucle

BUCLE

-0load r12

-0 load r14

-0 sd r14

-0 seq r5

```
18 start:
19     dadd r1,$gp,x
20     dadd r4,r1,#128
21     dadd r2,$gp,y
22     dadd r3,$gp,z
23     ld r10,a($gp)
24
25 loop:
26     ld r12,0(r1)
27     ld r14,0(r2)
28     dadd r12,r10,r12
29     dadd r14,r12,r14
30     sd r14,0(r3)
31     dadd r1,r1,#8
32     dadd r2,r2,#8
33     dadd r3,r3,#8
34     seq r5,r4,r1
35     beqz r5,loop
36     nop
37     nop
38     trap #0 ; Fin de programa
39
```

-2 riesgo control beqz
16*(2)=32 cicles parada

Restar 1 per el predict not taken de la ultima iteracio
172+32-1=203

Pregunta 3:

Partim del calcul de cicles de parada **sense dependències**, que igual que abans sera 172.

Solucionar les dependències: EMPRANT CURTSCIRCUITS + Delay Slot 1

-0 antes del bucle

BUCLE

-0load r12

-0 load r14

-0 sd r14

-0 seq r5

-0 riesgo control beqz

16*(1)=16 cicles parada

172+16 =188

```
25  loop:
26      ld r12,0(r1)
27      ld r14,0(r2)
28      dadd r12,r10,r12
29      dadd r14,r12,r14
30      sd r14,0(r3)
31      dadd r1,r1,#8
32      dadd r2,r2,#8
33
34      seq r5,r4,r1
35      beqz r5,loop
36      dadd r3,r3,#8
37      nop
38      nop
39      trap #0 ; Fin de programa
40
```

dadd r3 r3 #8 se pot desplaçar ahi (entre altres) porque el delay slot se pot passar ahi una instruccio i com el punter se te que manejar si o si, dona igual ferho abans del seq que despres del beqz.

Desarrollo de un nuevo programa

El codi esta en el arxiu search.s en la pagina següent.

INICIO	FINAL	Estado		
Resultados				
Ciclos	Instrucciones	CPI	Op. CF	Op. CF/Ciclo
77	60	1.28	0	0.00
Configuración				
Parámetro	Valor			
Programa	search.s			
Riesgos de datos	Forwarding			
Riesgos de control	Predict-not-taken (Lat=1)			
Registros	32			
Lat. L/S	2			
Lat. FP ADD	4			
Lat. FP CMP	4			
Lat. FP MUL	7			

Registros int		Registros fp		Memoria de Datos. Region 2					
Registro	Valor	Registro	Hi	Lo	Dirección	+0	+1	+2	+3
R0 (zero)	0	F0		0.0	a				
R1 (at)	8272	F1		0.0	8196		9		
R2 (v0)	0	F2		0.0	8200				
R3 (v1)	0	F3		0.0	8204		8		
R4 (a0)	0	F4		0.0	8208				
R5 (a1)	0	F5		0.0	8212		0		
R6 (a2)	0	F6		0.0	8216				
R7 (a3)	0	F7		0.0	8220		1		
R8 (t0)	0	F8		0.0	8224				
R9 (t1)	0	F9		0.0	8228		0		
R10 (t2)	3	F10		0.0	8232				
R11 (t3)	0	F11		0.0	8236		5		
R12 (t4)	0	F12		0.0	8240				
R13 (t5)	0	F13		0.0	8244		3		
R14 (t6)	0	F14		0.0	8248				
R15 (t7)	0	F15		0.0	8252		1		
R16 (s0)	0	F16		0.0	8256				
R17 (s1)	0	F17		0.0	8260		2		
R18 (s2)	0	F18		0.0	8264		0		

Simplement comentar que el numero de zeros que conta en el vector, els enmagatzema en el registre r10 →

Z: > AIC > prac2 > *ASM* search.s

```
1      .data
2  a:    .dword 9,8,0,1,0,5,3,1,2,0
3  tam:   .dword 10      ; Tamaño del vector
4  cont:  .dword 0       ; Número de componentes == 0
5
6      .text
7  start: dadd r1,$gp,a   ; Puntero
8         ld r4,tam($gp) ; Tamaño lista
9         dadd r10,r0,r0 ; Contador de ceros
10
11  loop:
12         ld r12,0(r1)   ;carregar dada del vector
13         dadd r1,r1,#8   ;moure punter a seguent dada del vector
14         daddi r4, r4, -1 ;restar 1 dada per llegir
15         beqz r12, mas1  ;si es un 0 el element del vector...
16         bnez r4, loop   ;si encara queden dades per llegir, iterar loop
17  acabar:
18         nop
19         nop
20         trap #0
21
22  mas1:  daddi r10,r10,1 ;sumar 1 al contador de zeros.
23         bnez r4, loop   ;si encara queden dades per llegir, iterar loop
24         j acabar
25
26
27
28
```