

Nome, MATRICOLAPierpaolo...Bene...S319841.....

[illegible][illegible]

08 Luglio, 2024 – Architetture dei Sistemi di Elaborazione

Nome, MATRICOLAPierpaolo...Bene...S319841.....

Domanda 2

Considerando il programma precedente, calcolare la miss prediction ratio per i seguenti casi:

1. processore con predittore di salti statico di tipo always taken
2. processore con predittore di salti statico di tipo always not taken
3. processore con predittore di salti dinamico di tipo BHT di 2-bit con 1024 linee, con tutti i predittori inizializzati a 0.

1. In un processore con predittore di salti statico di tipo always taken avrei le prime 99 predizioni corrette e l'ultima (quella in cui il salto non viene preso) errata. Perciò avrei un'unica misprediction -> $\text{misprediction ratio} = 1/100 = 0.01$, in percentuale -> $0.01 * 100 \rightarrow 1\%$

2. Processore di tipo always not taken ho la situazione opposta, cioè 99 volte la prediction è errata, una volta giusta.

Ho quindi 99 misprediction -> il ratio di misprediction è $= 99/100 = 0.99 * 100 = 99\%$

3. Nel caso di un BHT di 2-bit -> prima iterazione il bht passa da 0 a 1, (misprediction count = 1), nella seconda da 1 a 2 (misprediction count = 2), nella terza da 2 a 3 (misprediction count = 2), e rimane a 3 sino all'ultima iterazione in cui passa da 3 a 2 (misprediction count = 3). Perciò il ratio di misprediction è dello 3%.

In questa situazione sarebbe stato più conveniente un predittore di salti statico di tipo always taken.

08 Luglio, 2024 – Architetture dei Sistemi di Elaborazione

Nome, MATRICOLAPierpaolo...Bene...S319841.....

Domanda 3

Considerando il processore e il programma riportati nella prima domanda, si ottimizzi il programma utilizzando tutte le strategie di ottimizzazione statica inclusa l'abilitazione del branch delay slot.

Si riporti il nuovo codice e si calcoli il tempo di esecuzione dell'intero programma in colpi di clock.

																																			Clock cycles
.data																																			
V1:	.double	"100 values"																																	
V2:	.double	"100 values"																																	
V3:	.double	"100 values"																																	
...																																			
V5:	.double	"100 zeros"																																	
V4:	.double	"100 values"																																	
V5:	.double	"100 values"																																	
V1:	.double	"100 values"																																	
.text																																			
main:	daddui	r1,r0,0	F	D	E	M	W																										5		
	daddui	r2,r0,100		F	D	E	M	W																									1		
loop:	l.d	f1,v1(r1)			F	D	E	M	W																								1		
	l.d	f2,v2(r1)				F	D	E	M	W																							1		
	l.d	f3,v3(r1)					F	D	E	M	W																						1		
	mul.d	f4,f1,f2					F	D	*	*	*	*	*	*	*	M	W																8		
	mul.d	f5,f1,f3					F	D	*	*	*	*	*	*	*	M	W																1		
	l.d	f6,v1(r1+8)						F	D	E	M	W																					0		
	l.d	f7,v2(r1+8)						F	D	E	M	W																					0		
	l.d	f8,v3(r1+8)							F	D	E	M	W																				0		
	daddui	r1,r1,16								F	D	E	M	W																			0		
	mul.d	f9,f6,f7								F	D	*	*	*	*	*	*	*	*	M	W												5		
	mul.d	f10,f6,f8								F	D	*	*	*	*	*	*	*	*	M	W												1		
	div.d	f4,f4,f3									F	D	/	/	/	/	/	/	/	/	M	W											1		
	s.d	f5,v5(r1)									F	D	E	M	W																		0		
	s.d	f4,v4(r1)										F	D	E	s	s	s	s	s	S	M	W											1		
	s.d	f10,v5(r1+8)										F	D	s	s	s	s	s	S	E	M	W											1		
	div.d	f9,f9,f8														F	D	s	s	S	S	/	/	/	/	/	/	/	/	/	M	W		6	

Nome, MATRICOLAPierpaolo...Bene...S319841.....

[illegible][illegible]

08 Luglio, 2024 – Architetture dei Sistemi di Elaborazione

Nome, MATRICOLAPierpaolo...Bene...S319841.....

Domanda 3

Considerando il programma ottimizzato della domanda precedente, che benefici si potrebbero trarre dall'introduzione di un'unità di divisione floating point pipelined di 8 stages?

Introducendo una unità di divisione pipeline da 8 stages , per come ho strutturato il programma si eviterebbero li stalli sulal div f9,f9,f8 poiché non dovrebbe attendere che la prima div sia finita, ma potrebbe già partire. In totale si risparmierebbero 2 colpi di clock ad ogni loop.

08 Luglio, 2024 – Architetture dei Sistemi di Elaborazione

Nome, MATRICOLAPierpaolo...Bene...S319841.....