classmate
Date 29.1.2020
Page 1

TOMASULO SCHEME

① the distribution of the hazard detection logic
② the elimination of stalls for CWAW and CWAR hazards.

assume the following latencies.
load - 1 cycle add - 2 cycles
mult - 6 cycles div - 12 cycles.

when the final d. c's ready to write the result.

INSTRUCTION STATION

instruction	issue	execute	write result.
fld f6, 32 (x2)	V	V	V
fld f2, 44 (x3)	V	V	V
fmuld f0, f2, f4	V	V	
fsubd f8, f2, f6	V	V	V
fdivd f10, f0, f6	V		
faddd f6, f8, f2	V	V	V

RESERVATION STATION

name	busy	op	V_j	V_k	q_j	q_k	A
load1	x						
load2	x						
add1	x						
add2	x						
add3	x						

name	busy	Vj	Vk	qj	qk	A
mul1	V	MUL mem[44+reg[x3]]				
mul2	V	DIV	mem[32+x2]	mul1		

REGISTER STATUS

field	f0	f2	f4	f6	f8	f10	...	f30
Qi	mul1					mul2		

tomasulo's scheme is applied in multiple-issue processes

Advantage: tomasulo's algo was designed before caches. the presence of cache - inherently unpredictable delays.

this has been one of the major motivations for dynamic scheduling (1) processors have become more aggressive in their issue capability.

(2) register renaming, dynamic scheduling and speculation became more important.

(3) it can achieve high performance without requiring the compiler to target code to a specific pipeline structure.

NOTE: branches may be better.

HARDWARE BASED SPECULATION

try to exploit more ILP maintaining control deps. becomes an increasing burden. so branch prediction reduces the direct stalls attributable to branches.

Overcoming control dependence is done by speculating on the outcome of branches. With speculation, we fetch, issue, and execute instructions.

whereas dynamic scheduling only fetches and then issues instructions. we need mechanism to handle the situation where speculation is incorrect.

hardware based speculation combines 3 key ideas:

1. dynamic branch prediction
2. speculation to allow the execution of instruction before the control dependencies are resolved.
3. dynamic scheduling to deal with the scheduling of different combinations of basic blocks.

deterministic

predicative algo.

hardware based speculation follows the predictive flow of data values to choose when to execute instructions. this is referred to as data flow execution.

slow

promise async.

issue execute write (buffer) commit.

reservation station

+ROB

ROB must be free

the ROB has 4 fields.

instruction time, destination, value, ready.

REORDER BUFFER

entry	busy	inst.	state	dest.	value
1	x	fld f6, 32(x2)	commit	f6	mem[32 + reg[x2]]
2	x	fld f2, 44(x3)	commit	f2	mem[44 + reg[x3]]
3	✓	fmul.d f0, f2, f4	write result	f0	#2 x reg[f4]
4	✓	fsub.d f8, f2, f6	write result	f8	#2 - #1
5	✓	fdiv.d f10, f0, f6	execute	f10	
6	✓	fadd.d f6, f8, f2	write result	f6	#4 + #2

RESERVATION STATION

name	busy	op	v_j	v_k	q_j	q_k	dest.	A
load1	no							
load2	no							
add1	no							
add2	no							
add3	no							
mul1	no	fmul.d	mem[44+x3]	reg[f4]			#3	
mul2	yes	fdiv.d	mem[32+x2]			#3	#5	

(ROB numbers)

FP REGISTER STATUS

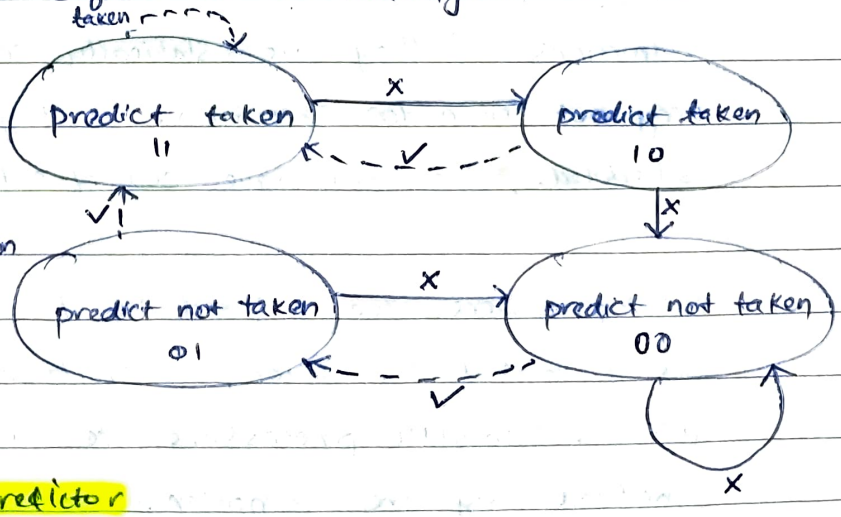
field	f0	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10
reorder#	3						6		4		5
busy	yes						y		y		y

-

5th edi Appendix 3.

DYNAMIC BRANCH PREDICTION AND BRANCH PREDICTION BUFFER

2-bit prediction scheme: a prediction unit must miss twice before it is changed.



states in a 2-bit prediction scheme.

- tournament predictor
- correlative predictor