12/2019

Monday	30	2	9	16	23
Tuesday	31	3	10	17	24
Wednesday	–	4	11	18	25
Thursday	–	5	12	19	26
Friday	–	6	13	20	27
Saturday	–	7	14	21	28
Sunday	1	8	15	22	29

180010011

NOVEMBER' 19

Wk-48 Day 329-036

MONDAY

25

Assignment-7

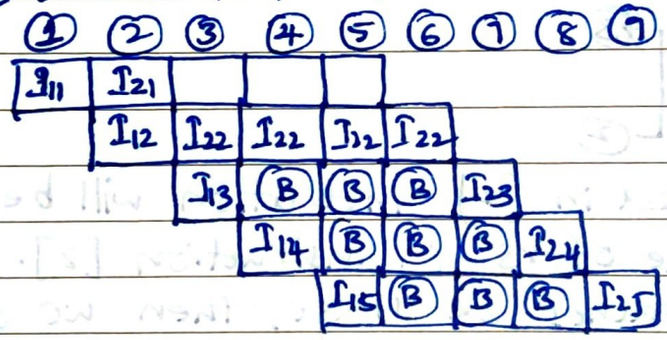
1Ans) Compilers may or maynot execute NOP instructions properly.

Relying on compilers would make them hardware-specific for every different hardware, a different compiler would be needed.

Pipeline Interlock is a mechanism to detect data hazards/control hazards and resolve them, which is necessary to preserve original data dependencies specified in a sequence of the instructions, so basically, an interlock prevents instructions from being executed in a wrong sequence.

Eg:- [1] ADD R1, R2, R3

[2] SUB R4, R1, R2



Depending upon the conflicting instructions is in with the instruction at OF stage. When the instruction is at OF stage, the conflicting instruction can be at EX, MA & RW stages.

∴ Minimum no. of bubbles if there is a conflict can be 1.

2Ans) Yes, there are ways of reducing performance impacts introduced through interlocking-based handling of control hazards.

i) Convert the instruction in the IF and OF stages, to bubble once a branch instruction reaches the EX stage. Start fetching from the next PC (not taken) or the branch target (taken).

notes

Phone

email

ii) If the branch instruction in the EX stage is taken, then invalidate the instruction in the IF and OF stages. Start fetching from the branch target. Otherwise do not take any spacial.

3) Let us consider the example.

[1] : Add r5, r3, r2
[2] : Add r5, r1, r2
[3] : Add r7, r2, r5

As per the question,

	①	②	③	④	⑤
IF					
OF	③				
EX		~~③~~			
MA	②				
RW	①	②			

R5 value executed in [1] instruction will be over written by the executed value of r5 in instruction [2]. So in the next cycle, [1] reaches ex stage, then we can forward the value of r5 from [2] in RW stage to [3] in EX stage.

4 AM) Given,
two processors P_1 & P_2. P_1 runs at a frequency of 1GHz, while P_2 runs at a frequency of 1.2 GHz.
P_1 has a branch misprediction penalty of 2 cycles, while P_2 has a penalty of 3 cycles.

→ From P_1

Clock frequency = 1 GHz.
Branch misprediction penalty = 2 cycles.
Ideal CPI = 1

notes
Phone
email
website

New CPI = CPI^{P1}_{NEW}

= 1 + 2 × (30% + 80% of 30%)

= 1 + 2 × (30% + 24%)

= 1 + 1.08

CPI^{P1}_{NEW} = 2.08.

Taking a 5-stage Pipeline

$Speedup_{P1} \Rightarrow \frac{5}{CPI^{P1}_{NEW}}$

$= \frac{5}{2.08}$

$Speedup_{P1} = 2.404$

→ From P2

Clock frequency = 1.2 GHz, Branch misprediction penalty = 3

Ideal CPI = 1

New CPI = CPI^{P2}_{NEW}

= 1 + 3 × (30% + 80% of 30%)

= 1 + 3 × (0.3 + 0.8 × 0.3)

= 1 + 3 × 0.54

= 1 + 1.62

CPI^{P2}_{NEW} = 2.62

Taking a 5stage pipeline $Speedup_{P2} = \frac{5}{CPI^{P2}_{NEW}}$

$= \frac{5}{2.62}$

$Speedup_{P2} = 1.908$

∴ The speedup of processor P1 is better than that of P2.

notes

Phone

email

website