

PEMBAHASAN SOAL UAS OAK 2019

1.

1) Diket : CPU time A = 10 detik
 clock rate A = clock rate B = 2 GHz
 $= 2 \times 10^9$
 Instruksi B = $1,5 \times$ Instruksi A

a) clock cycle A = CPU time A \times clock rate A
 $= 10 \times 2 \times 10^9$
 $= 20 \times 10^9$
 clock cycle B = $1,5 \times 20 \times 10^9$
 $= 30 \times 10^9$

clock rate A = clock rate B
 $\frac{\text{clock cycle A}}{\text{CPU time A}} = \frac{\text{clock cycle B}}{\text{CPU time B}}$
 $\frac{20 \times 10^9}{10} = \frac{30 \times 10^9}{\text{CPU time B}}$
 CPU time B = $\frac{10 \times 30 \times 10^9}{20 \times 10^9}$
 $= 15 \text{ detik}$

b) CPI B = 0,8 CPI A
 jika CPI B = 1
 maka CPU time B = $1 \times 15 \text{ detik}$
 $= 15 \text{ detik}$
 jika CPI B = 0,8
 maka CPU time B = $0,8 \times 15 \text{ detik}$
 $= 12 \text{ detik}$

2.

a) add \$t0, \$s1, \$s2
 mult \$t0, \$s0, \$t0
 div \$t0, \$t0, \$s3
 sw \$t0, 0(\$s0)

b) $t0 = a \times 4$
 $t1 = b \times 4$
 $t0 = X[a]$
 $t1 = Y[b]$
 $s0 = X[a]$
 $t2 = X[a] + 8$
 $t0 = X[a] + 8$
 $t0 = X[a] + 8 + X[a]$
 $t1 = X[a] + 8 + X[a]$
 $Y[b] = X[a] + 8 + X[a]$

3.

Control Hazard

	offset	t =	1	2	3	4	5	6	7	8	9
a:=0	0100	MOV AX, 0000	IF	DE	IF	DE	EX				
repeat	0103	MOV CX, 000A		IF	DE	IF	DE	EX			
a:=a+1	0106	ADD AX, 0001			IF	DE	IF	DE	EX		
until a=10	0109	XOR CX, AX				IF	DE	IF	DE	EX	
	010B	JNZ 0103					IF	DE	IF	DE	EX
a:=b	010D	POP AX						IF	DE	EX	

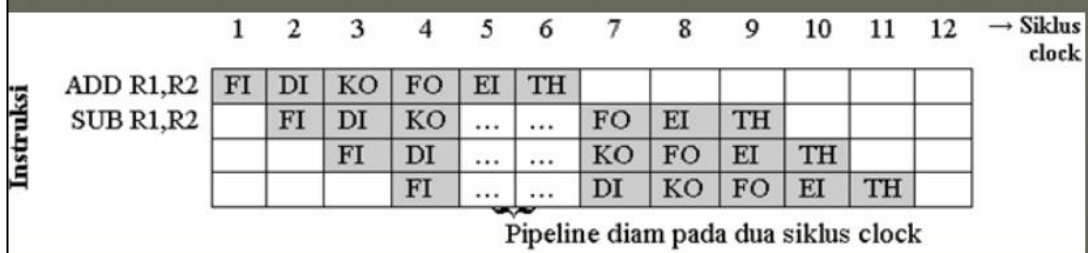
Saat instruksi JNZ 0103 dieksekusi (t9), instruksi POP AX telah selesai dieksekusi (t8). Jika lompatan ternyata tidak dilakukan, maka hal ini tidak menjadi masalah karena memang instruksi berikutnya adalah POP AX. Namun jika lompatan ternyata dilakukan, maka hal ini akan menjadi masalah karena register AX telah diubah oleh instruksi POP AX.

Data Hazard

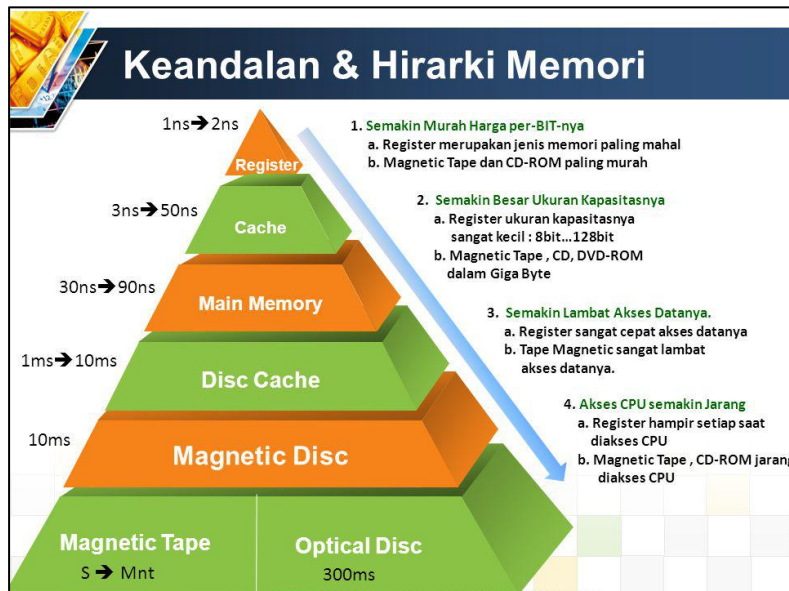
Ketergantungan Data (data hazard)

Jika operand untuk suatu instruksi merupakan hasil dari instruksi sebelumnya yang belum selesai dalam pipeline, maka kasus ini disebut ketergantungan data (*data dependency*).

Contoh:



4. a)



b)

- Pemroses memberi instruksi ke perangkat I/O kemudian melanjutkan melakukan pekerjaan lainnya.
- Perangkat I/O akan menginterupsi meminta layanan saat perangkat telah siap bertukar data dengan pemroses.
- Saat menerima interupsi perangkat keras (yang memberitahukan bahwa perangkat siap melakukan transfer), pemroses segera mengeksekusi transfer data.

- c)
- Program Driven digunakan CPU untuk menyelesaikan hanya satu tugas saja, yang mana prosesnya akan begitu lama karena belum bisa multitasking.
 - Namun pada Interrupt Driven I/O ketika proses lain ada yang masuk akan disanggah dan difilter proses mana yang akan dikerjakan terlebih dahulu, yang memungkinkan Interrupt Driven I/O bisa multitasking.
 - DMA menyimpan instruksi-intruksi yang diterima oleh I/O, sehingga CPU tidak mengalami crash dan dipastikan bisa multitasking.

5.

5) a) Multiplicand (M) = 7 = 0111
Multiplier (Q) = 8 = 1000

•> Start rep = 1
Q = 1000 → M shift left = 1110
Q shift right = 0100
rep 1 < 32 → Start

•> Start rep = 2
Q = 0100 → M shift left = 11100
Q shift right = 0010
rep 2 < 32 → Start

•> start rep = 3
Q = 0010 → M shift left = 111000
Q shift right = 0001
rep 3 < 32 → Start

•> start rep = 4
Q = 0001 → Result = M = 111000
M shift left = 1110000
Q shift right = 0000
rep 4 < 32 → Start

•> Start rep = 5
dan seterusnya result tidak berubah
karena Q = 0000 (tidak ada angka 1)

b) Result sudah ditemukan pada repetisi ke 4, namun repetisi tetap dilakukan sebanyak 32 kali. Sehingga tidak efektif dalam waktu karena harus menunggu 32 repetisi.