

## Calcolatori Elettronici L A

### Prova scritta del 20/06/'03

#### Esercizio N. 1 (12 punti)

Sia data la seguente sequenza di istruzioni con le tre variabili di nome "alfa" "beta" e "gamma" eseguita su un processore DLX dotato di pipeline:

LW R28, alfa(R0)  
LW R29, beta(R0)  
AND R28, R28, R29  
XOR R28, R28, R29  
SW gamma(R0), R28

- 1.1) Scrivere in forma matematica la funzione che lega le variabili realizzata da tale sequenza di istruzioni (punti 2)
- 1.2) tracciare il diagramma clock/stadi corrispondente alla sequenza (4 punti)
- 1.3) progettare il circuito di forwarding col quale si possano ridurre gli stalli di pipeline (punti 4)
- 1.4) tracciare il nuovo diagramma clock/stadi che tiene conto del forwarding implementato (2 punti)  
*(Per la risposta alla domanda 1.3 si immagini di disporre di un registro latch ausiliario "AUX" -dello stesso tipo dei pipeline registers- a valle dello stadio WB che, al termine del WB, campiona lo stesso dato destinato al RF, nello stesso istante in cui lo campiona il RF).*

#### Esercizio N. 2 (10 punti)

Un sistema a microprocessore basato su 8088 a 5 MHz dispone di 192KB di Eprom (3 chip da 64 agli indirizzi alti, sia E0 il banco più "alto") e 224KB di Ram (3 chip da 64 e uno da 32, in modo che, dall'indirizzo 0h e per i primi 224KB, non vi siano indirizzi liberi né celle replicate). I chip di eprom hanno tutti Tacc=120ns, Toe=30ns, quelli di ram da 64 hanno Tacc=0.4μs, Toe=0.6μs e quello di ram da 32 ha Tacc=300ns, Toe=196ns.

- 2.1) disegnare lo schema del sistema e progettare il circuito per la generazione del segnale di READY, avendo cura di definire tutti i segnali, nell'ipotesi che non vi siano periferiche di I/O. (6 punti)
- 2.2) scrivere le espressioni di selezione semplificate delle memorie e dei dispositivi di I/O. (4 punti)

#### Esercizio N. 3 (9 punti)

- 3.1) Descrivere i meccanismi della selezione dei registri interni delle periferiche attraverso la decodifica sequenziale e quella indiretta. (4 punti)
- 3.2) Descrivere EU e BIU nell'architettura iA16 confrontandole con la pipeline del DLX. (5 punti)

Durante il compito non si può uscire dopo le ore 11:00  
Consegnando un elaborato per la correzione decade la validità del voto precedente  
**Correzione e registrazione il 27/6/03, ore 10:00 Via Rasi**

# SOLUZIONE del compito di Calcolatori Elettronici L A del 20/6/'03

1.1)  $\gamma = (\alpha \text{ AND } \beta) \text{ XOR } \beta$

1.2) Diagramma clock/stadi

	ck1	ck2	ck3	ck4	ck5	ck6	ck7	ck8	ck9	ck10	ck11	ck12	ck13	ck14	ck15	ck16	ck17	ck18
LW R28, $\alpha(R0)$	IF	ID	EX	MEM	WB													
LW R29, $\beta(R0)$		IF	ID	EX	MEM	WB												
AND R28, R28, R29			IF	S	S	S	ID	EX	MEM	WB								
XOR R28, R28, R29							IF	S	S	S	ID	EX	MEM	WB				
SW $\gamma(R0)$ , R28											IF	S	S	S	ID	EX	MEM	WB

1.4) Diagramma clock/stadi con FU

	ck1	ck2	ck3	ck4	ck5	ck6	ck7	ck8	ck9	ck10
LW R28, $\alpha(R0)$	IF	ID	EX	MEM	WB					
LW R29, $\beta(R0)$		IF	ID	EX	MEM	WB				
AND R28, R28, R29			IF	S	ID	EX	MEM	WB		
XOR R28, R28, R29					IF	ID	EX	MEM	WB	
SW $\gamma(R0)$ , R28						IF	ID	EX	MEM	WB

Per ridurre gli stalli dell'AND si possono introdurre due rami di forwarding: il primo, al termine di WB(LW R28,..), porta R28 dal registro AUX posto a valle del WB ad un ingresso dell'ALU, ed il secondo porta R29 dall'MDR all'altro ingresso dell'ALU al termine di MEM(LW R29,..). In questo

modo subito dopo "ck5" è possibile eseguire EX(AND). Per eliminare gli stalli della XOR e della SW, quando sono eseguiti EX(XOR) ed EX(SW) la FU attiverà il ramo che porta "x" di EX/MEM ad un ingresso del MUX aggiunto in EX a monte dell'SMDR. Si veda più avanti lo schema con il circuito di forwarding completo.

2.1)  $T_{373}=18\text{ns}$ ,  $T_{245}=10\text{ns}$ ,  $T_{\text{pl}}=30\text{ns}$

Clock di sistema a 5 MHz:  $T_{\text{cl}}=200\text{ns}$ ,  $T_{\text{clav}}=110\text{ns}$ ,  $T_{\text{clrl}}=165\text{ns}$ ,  $T_{\text{dvc}}=30\text{ns}$

RAM64:  $T_{\text{acc}}=0.4\mu\text{s}$ ,  $T_{\text{oe}}=0.6\mu\text{s}$

$$(3+n) \cdot 200 \geq \max \begin{cases} 110+18+30+400+10+30 = 598 \\ \text{verificate per } n=3: 3 T_w \\ 200+165+600+10+30 = 1005 \end{cases}$$

EPROM:  $T_{\text{acc}}=120\text{ns}$ ,  $T_{\text{oe}}=30\text{ns}$

$$(3+n) \cdot 200 \geq \max \begin{cases} 110+18+30+120+10+30 = 318 \\ \text{verificate per } n=0: 0 T_w \\ 200+165+30+10+30 = 435 \end{cases}$$

RAM32:  $T_{\text{acc}}=300\text{ns}$ ,  $T_{\text{oe}}=196\text{ns}$

$$(3+n) \cdot 200 \geq \max \begin{cases} 110+18+30+300+10+30 = 498 \\ \text{verificate per } n=1: 1 T_w \\ 200+165+196+10+30 = 601 \end{cases}$$

Vedere schema per il circuito del ready e la connessione delle periferiche e delle memorie al bus di sistema.

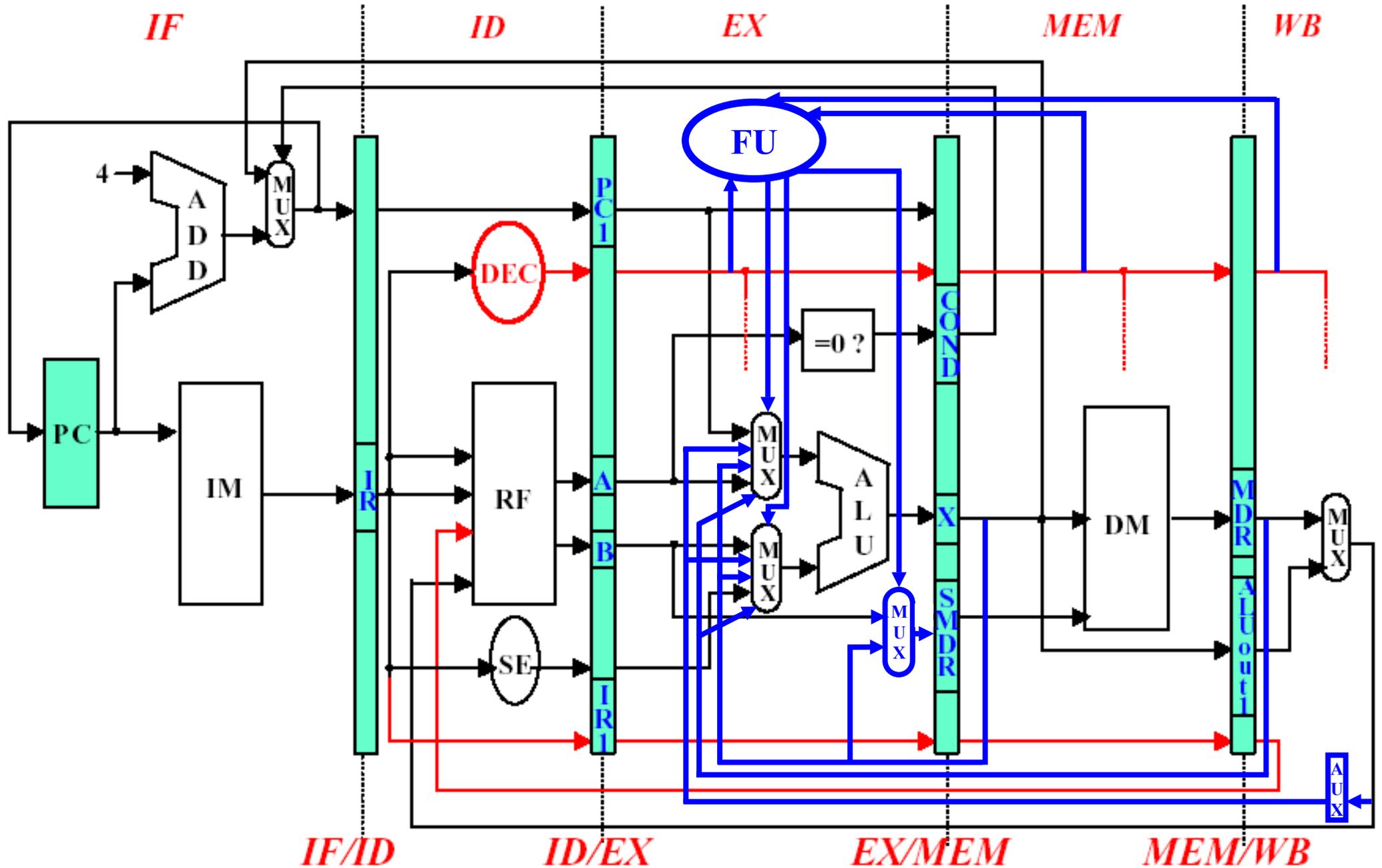
2.2) Espressioni di selezione semplificate delle memorie:

$$\begin{aligned} \text{CSE0} &= \text{BA19} \cdot \text{BA17} \cdot \text{BA16} \cdot \text{!IO/M\#} \\ \text{CSE1} &= \text{BA19} \cdot \text{BA17} \cdot \text{!BA16} \cdot \text{!IO/M\#} \\ \text{CSE2} &= \text{BA19} \cdot \text{!BA17} \cdot \text{!IO/M\#} \end{aligned}$$

$$\begin{aligned} \text{CSR0} &= \text{!BA19} \cdot \text{!BA17} \cdot \text{BA16} \cdot \text{!IO/M\#} \\ \text{CSR1} &= \text{!BA19} \cdot \text{!BA17} \cdot \text{BA16} \cdot \text{!IO/M\#} \\ \text{CSR2} &= \text{!BA19} \cdot \text{BA17} \cdot \text{!BA16} \cdot \text{!IO/M\#} \\ \text{CSR3} &= \text{!BA19} \cdot \text{BA17} \cdot \text{BA16} \cdot \text{!IO/M\#} \end{aligned}$$

3.1) Sono entrambi metodi per ridurre le linee di indirizzamento da collegare al bus esterno, nell'accedere ai registri interni delle periferiche di I/O (che ne contengono i parametri di funzionamento). La decodifica sequenziale è utile quando è possibile definire una sequenza predefinita di accesso ai registri: al primo accesso successivo al reset si scrive su R0, poi su R1, e così via. Non serve quindi indicare l'indirizzo interno del registro da accedere, perché tale compito è svolto da un contatore che li genera automaticamente in sequenza. La decodifica indiretta invece presuppone l'esistenza di registri interni di dimensione inferiore al parallelismo del bus dati; tale ipotesi è verosimile se si pensa che normalmente non è necessaria una grossa capacità di memoria per contenere i parametri di funzionamento di una periferica. Supponendo di avere registri a 6 bit, si usano i rimanenti 2 bit del bus dati a mo' di indirizzi per definire su quale registro si sta scrivendo.

3.2) EU e BIU sono due stadi dell'IA16 in grado di operare contemporaneamente e quindi capaci di introdurre una parallelizzazione ai  $\mu\text{P}$  basati su questa architettura. Mentre in EU la ALU esegue i calcoli e aggiorna i registri, la BIU calcola gli indirizzi ed accede sui dispositivi del bus per le operazioni di fetch, lettura o scrittura. Tale meccanismo può considerarsi, alla stregua di una pipeline più complessa (come quella del DLX), capace migliorare le prestazioni del  $\mu\text{P}$  in quanto ottimizza l'occupazione del bus sfruttandone meglio la banda.



Schema compito Calcolatori Elettronici del 20/6/'03 - Domanda 2.1

