



Struttura della pipeline

Prof. Alberto Borghese
Dipartimento di Scienze dell'Informazione
borgnese@dsi.unimi.it

Università degli Studi di Milano

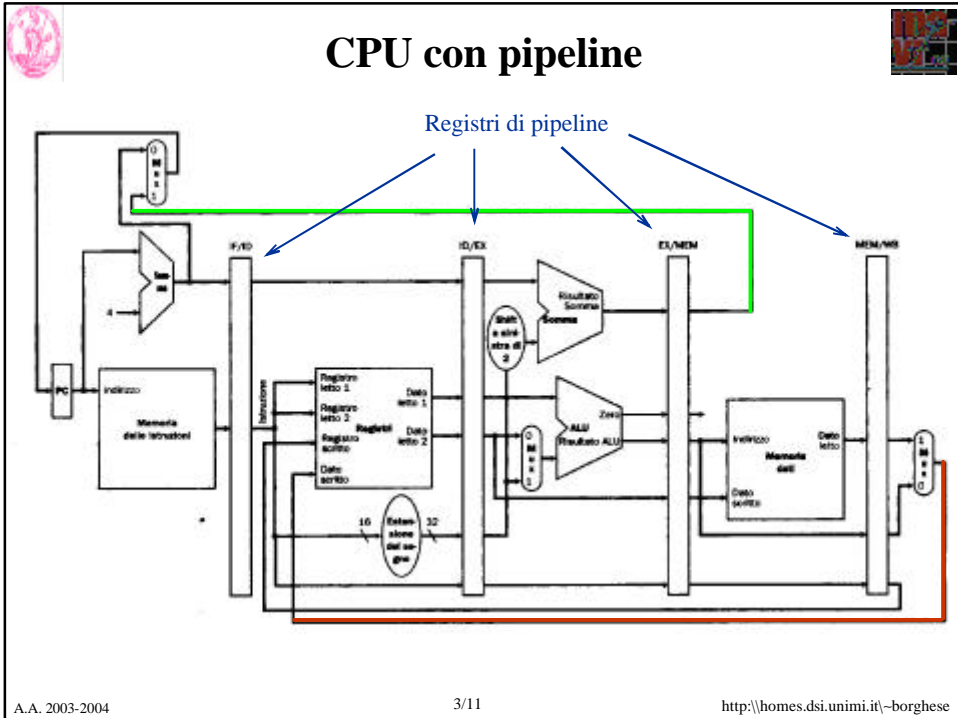


Sommario

La CPU con pipeline

Esecuzione di istruzioni nella CPU con pipeline

Le criticità della pipeline



Gli stadi di esecuzione

IF – Instruction Fetch
 ID – Instruction Decode (e lettura register file)
 EX – Esecuzione o calcolo dell'indirizzo di memoria.
 MEM – Accesso alla memoria dati.
 WB – Write Back (scrittura del risultato nel register file).

NB: I registri al termine di ogni fase prendono il nome dalle 2 fasi:
 IF/ID ID/EX EX/MEM MEM/WB

Perchè non c'è un registro WB/IF?

Il data-path procede da sx a dx.

Da dx a sx si ha la scrittura del PC e la scrittura nel Register File che creano criticità (vanno contro-corrente).

Supponiamo che ciascuno stadio abbia la sua unità di controllo.

A.A. 2003-2004 4/11 http://homes.dsi.unimi.it/~borgnese

Pipeline per le istruzioni lw

Passo esecuzione	ALU	ALU PC	ALU branch	Memoria Dati	Memoria Istruzioni	Register File
IF (Fase fetch)	NO	Yes	NO	NO	Yes	NO
ID (Decodifica)	NO	NO	NO	NO	NO	Yes
EX (Esecuzione)	Yes	NO	Yes	NO	NO	NO
MEM (Accesso memoria)	NO	NO	NO	Yes	NO	NO
WB (riscrittura)	NO	NO	NO	NO	NO	Yes

Passi della 1a istruzione lw	IF	ID	EX	MEM	WB		
.....							
lw \$t0, 8(\$t2)	ALU-PC Mem-Istr	RF	ALU	Mem	RF		
lw \$t1, 12(\$t2)		ALU-PC Mem-Istr	RF	ALU	Mem	RF	
lw \$t1, 16(\$t2)			ALU-PC Mem-Istr	RF	ALU	Mem	RF
.....							

Il ruolo dei registri

Ciascuno stadio produce un risultato. La parte di risultato che serve agli stadi successivi deve essere memorizzata in un registro.

Il registro mantiene l'informazione anche se lo stadio in questione riutilizza l'unità funzionale.

Esempio: l'istruzione letta viene salvata nel registro IF/ID (cf. Instruction Register).



Sommario



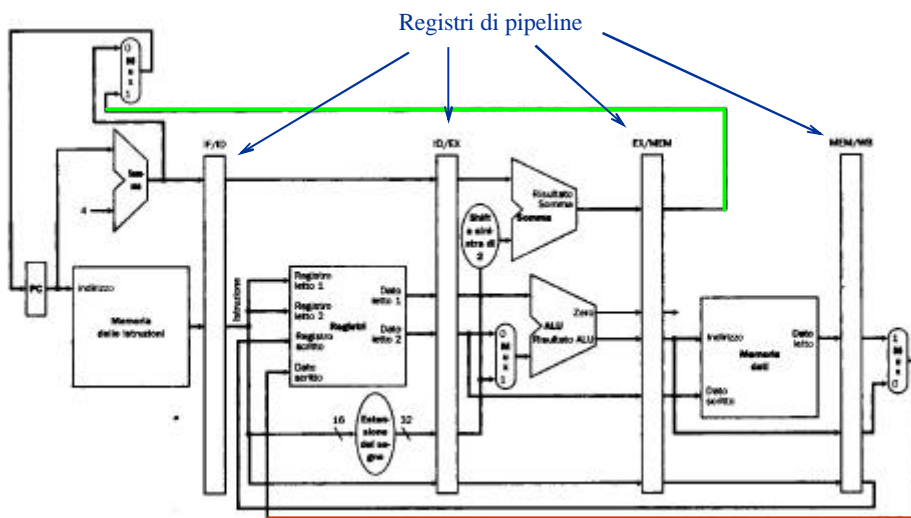
La CPU con pipeline

Esecuzione di istruzioni nella CPU con pipeline

Le criticità della pipeline



Esecuzione su una CPU con pipeline



Istruzioni di lw/sw, di tipo R e di branch.



Sommario



La CPU con pipeline

Esecuzione di istruzioni nella CPU con pipeline

Le criticità della pipeline



Criticità (hazard)



Un'istruzione non può essere eseguita nel ciclo di clock immediatamente successivo a quella precedente (mancano i dati necessari alla lavorazione di un qualche suo stadio).

Strutturali:

- Dovrei utilizzare la stessa unità funzionale due volte nello stesso ciclo di clock (e.g. se non avessi duplicato la memoria)..

Controllo:

- Dovrei prendere una decisione (sull'istruzione successiva) prima che l'esecuzione dell'istruzione corrente sia terminata (e.g. Istruzioni successive ad una branch).

Dati:

- Dovrei eseguire un'istruzione in cui uno dei dati è il risultato dell'esecuzione di un'istruzione precedente.

Esempio:

```
add $s0, $t1, $t1
add $s2, $s0, $t3
```



Sommario



La CPU con pipeline

Esecuzione di istruzioni nella CPU con pipeline

Le criticità della pipeline