

## 7' Esercitazione

<https://politecnicomilano.webex.com/meet/gianenrico.conti>

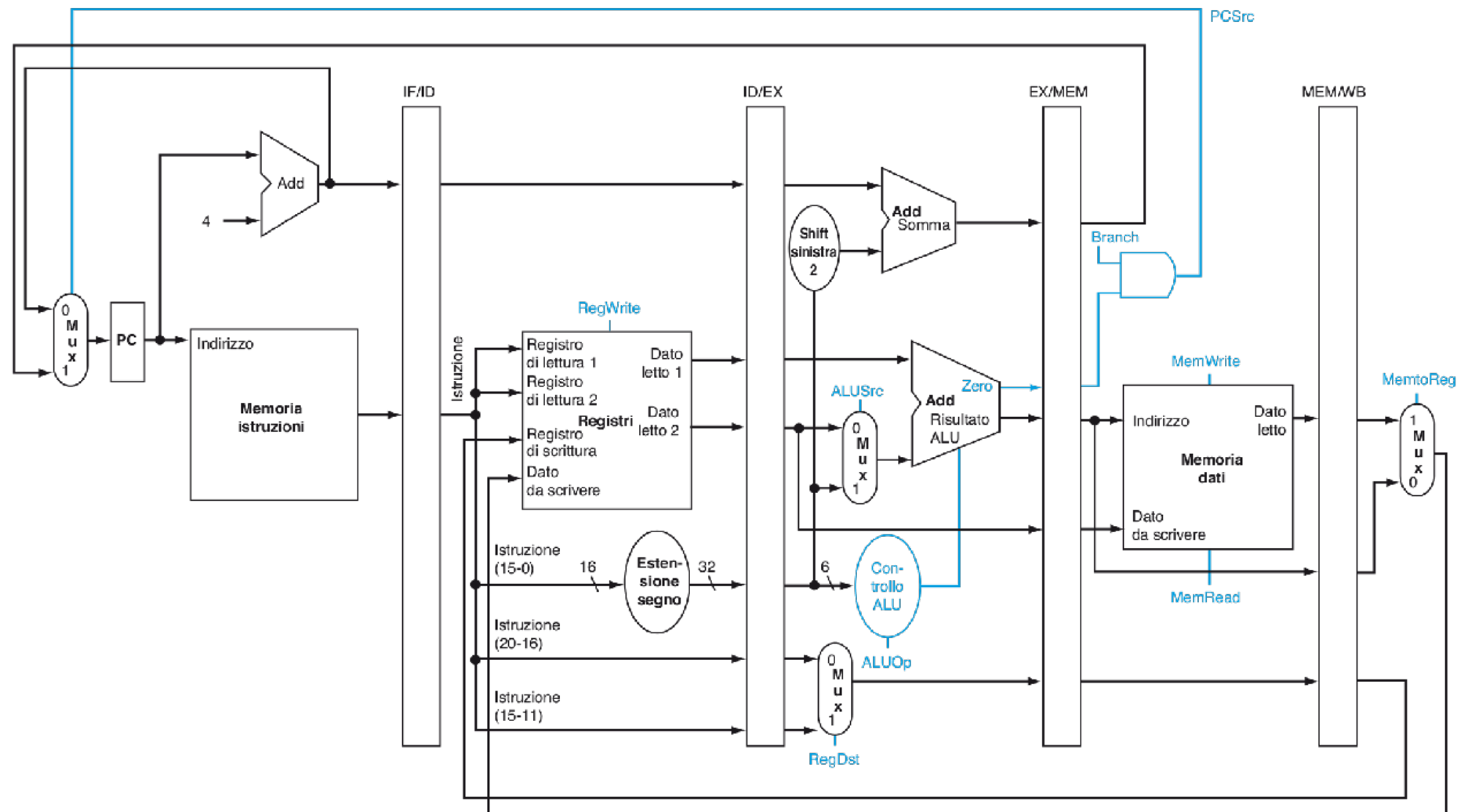
2 maggio 2022

Gian Enrico Conti  
MIPS architecture (pipelined)

**II parte**

# Architettura Pipelined: recap

- Aggiungendo dei registri tra le varie fasi si ottiene una architettura pipelined in piu punti:



# CPI recap

---

- Per misurare l'efficienza di una certa architettura si utilizza un indicatore chiamato CPI ( Cycles Per Instruction )

- $$\text{CPI} = \frac{\# \text{ TOTALE CICLI}}{\# \text{ TOTALE ISTRUZIONI}}$$

- Intuitivamente il CPI rappresenta il numero medio di cicli necessari per completare un istruzione

- **Attenzione!**

- $\text{CPI} \geq 1$       al massimo termina un istruzione per ciclo se non ci sono stalli

# Terminologia: recap

---

- Five stage “RISC” load-store architecture:
  - 1. Instruction fetch (IF)
    - get instruction from memory, increment PC
  - 2. Instruction Decode (ID)
    - translate opcode into control signals and read registers
  - 3. Execute (EX)
    - perform ALU operation, compute jump/branch targets
  - 4. Memory (MEM)
    - access memory if needed
  - 5. Writeback (WB)
    - update register file
  -

# Conflitti all'interno della pipeline **recap**

---

- I conflitti sorgono nelle architetture con pipelining quando non è possibile eseguire un'istruzione nel ciclo immediatamente successivo

- Conflitti strutturali

- Tentativo di usare la stessa risorsa hardware da parte di diverse istruzioni in modi diversi nello stesso clock

- Conflitti sui dati

- Tentativo di usare un risultato prima che sia disponibile

- Conflitti di controllo

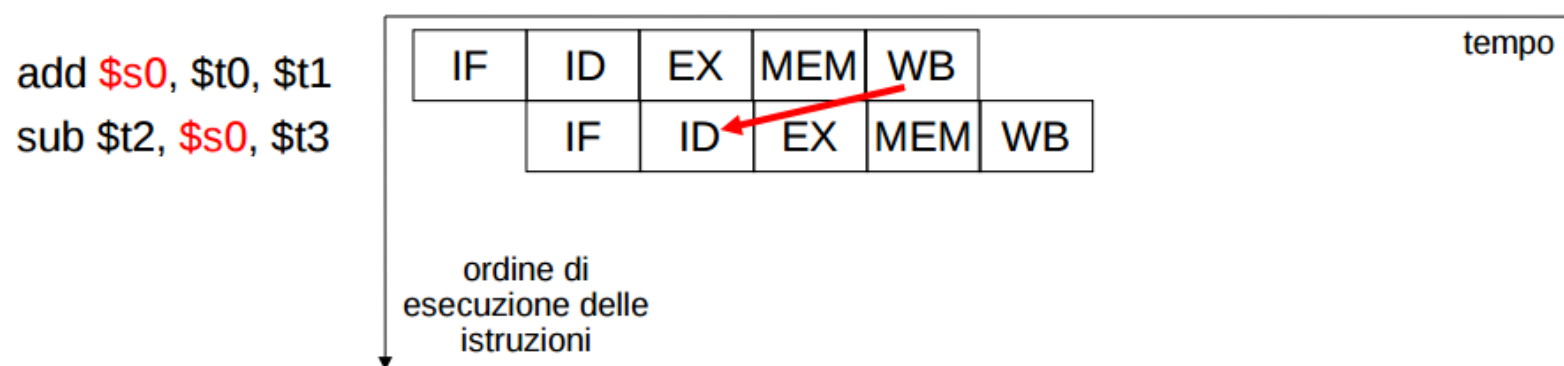
- Nel caso di salti, decidere quale prossima istruzione da eseguire prima che la condizione sia valutata

# Conflitti sui dati: recap

- Un'istruzione dipende dal risultato di un'istruzione precedente che è ancora nella pipeline

Istruzione
add \$s0, \$t0, \$t1
sub \$t2, \$s0, \$t3

- Nella pipeline queste istruzioni vengono rappresentate come



# Conflitti sui dati: **recap** soluzioni

---

- soluzioni SW

- **NOP**                      Inserimento di istruzioni NOP per evitare il conflitto
- **Scheduling**              Cambiamento dell'ordine di esecuzione delle istruzioni mantenendo equivalenza funzionale

- Soluzioni HW

- Inserimento di bolle (**bubble**) o stalli nella pipeline
  - Si inseriscono dei tempi morti
  - Peggiora il throughput
- Propagazione o scavalcamiento (**forwarding** o bypassing)
  - Si propagano i dati in avanti appena sono disponibili verso le unità che li richiedono

# Soluzioni ai conflitti di controllo **recap**

---

## ■ Soluzioni di tipo hardware

- Inserimento di bolle (**bubble**) o stalli nella pipeline (3 cicli)
  - Si inseriscono dei tempi morti
  - Peggiora il throughput
- Ridurre i ritardi associati ai salti condizionati
  - Comparatore in fase di **decode**
  - Calcolo dell'indirizzo di destinazione in fase di **decode**
- Predizione Statica
  - Si assume **branch taken** o **branch not taken**
    - In caso di errore si **invalida** l'istruzione in esecuzione
- Predizione Dinamica
  - Comparatore in fase di **decode**
  - Tabella della storia dei salti ( **branch prediction buffer** )
  - Predizione a 1 o 2 bit



## Esercizi II

---

- Dipendenze
- Calcolo del CPI(clock cycles per instruction)

$$CPI = \frac{\# \text{ TOTALE CICLI}}{\# \text{ TOTALE ISTRUZIONI}}$$

# Esercizio 4

---

Dato il seguente codice assembly:

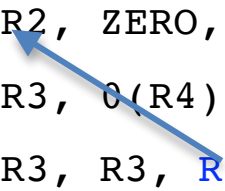
```
        ADDI    R1, ZERO, 2
        ADDI    R2, ZERO, 7
TAG:    LW      R3, 0(R4)
        ADD     R3, R3, R2
        SW      R3, 0(R4)
        ADDI    R4, R4, 8
        SUBI    R1, R1, 1
        BNEQ    R1, ZERO, TAG
        NOP
```

1. Identificare le dipendenze dati del seguente codice
2. Risolvere i conflitti presenti utilizzando gli STALLI e calcolare il CPI

## Esercizio 4: Identificare le dipendenze dati del seguente codice

---

```
      ADDI    R1, ZERO, 2
      ADDI    R2, ZERO, 7
TAG:  LW      R3, 0(R4)
      ADD     R3, R3, R2
      SW      R3, 0(R4)
      ADDI    R4, R4, 8
      SUBI    R1, R1, 1
      BNEQ    R1, ZERO, TAG
      NOP
```



1 dip: R2 in ADD4 da ADDI2

E così via.

## Esercizio 4: Identificare le dipendenze dati del seguente codice

---

```
      ADDI    R1, ZERO, 2
      ADDI    R2, ZERO, 7
TAG:  LW      R3, 0(R4)
      ADD     R3, R3, R2
      SW      R3, 0(R4)
      ADDI    R4, R4, 8
      SUBI    R1, R1, 1
      BNEQ    R1, ZERO, TAG
      NOP
```

Dipendenza RAW su R2 tra ADDI2 e ADD4

Dipendenza RAW su R3 tra LW3 e ADD4

Dipendenza RAW su R3 tra ADD4 e SW5

Dipendenza RAW su R1 tra ADD1 e SUBI7

Dipendenza RAW su R1 tra SUBI7 e BNEQ8

Ma anche dip interciclo a causa del salto ...

# Esercizio 4: Identificare le dipendenze dati del seguente codice

---

```
ADDI    R1, ZERO, 2
ADDI    R2, ZERO, 7
TAG:    LW     R3, 0(R4)
        ADD     R3, R3, R2
        SW     R3, 0(R4)
        ADDI    R4, R4, 8
        SUBI    R1, R1, 1
        BNEQ   R1, ZERO, TAG
        NOP
```

Dipendenza RAW su R2 tra ADDI2 e ADD4

Dipendenza RAW su R3 tra LW3 e ADD4

Dipendenza RAW su R3 tra ADD4 e SW5

Dipendenza RAW su R1 tra ADD1 e SUBI7

Dipendenza RAW su R1 tra SUBI7 e BNEQ8

## dip interciclo: (salto)

Dipendenza RAW su R4 tra ADDI6 e LW3

Dipendenza RAW su R4 tra ADDI6 e SW4

Dipendenza RAW su R4 tra ADDI6 e ADDI6

Dipendenza RAW su R3 tra ADD4 e ADD4

Dipendenza RAW su R1 tra SUBI7 e SUBI7

**Dipendenza di controllo su BNEQ8**

## Esercizio 4: Risolvere i conflitti presenti utilizzando gli STALLI e calcolare il CPI

ADDI R1, ZERO, 2  
ADDI R2, ZERO, 7  
TAG: LW R3, 0(R4)  
ADD R3, R3, R2  
SW R3, 0(R4)  
ADDI R4, R4, 8  
SUBI R1, R1, 1  
BNEQ R1, ZERO, TAG  
NOP

CICLO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
ADDI1	F	D	E	M	W																
ADDI2		F	D	E	M	W															
LW3			F	D	E	M	W														
ADD4				F	X	X	D	E	M	W											

Dipendenza RAW su R2 tra ADDI2 e ADD4

Ma anche

Dipendenza RAW su R3 tra LW3 e ADD4

...

# Esercizio 4: Risolvere i conflitti presenti utilizzando gli STALLI e calcolare il CPI

```
ADDI    R1, ZERO, 2
ADDI    R2, ZERO, 7
TAG: LW   R3, 0(R4)
ADD     R3, R3, R2
SW      R3, 0(R4)
ADDI    R4, R4, 8
SUBI    R1, R1, 1
BNEQ    R1, ZERO, TAG
NOP
```

CICLO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
ADDI1	F	D	E	M	W																
ADDI2		F	D	E	M	W															
LW3			F	D	E	M	W														
ADD4				F	X	X	D	E	M	W											
SW5							F	X	X	D	E	M	W								

Dipendenza RAW su R2 tra ADDI2 e ADD4

Dipendenza RAW su R3 tra ADD4 e SW5

..

# Esercizio 4: Risolvere i conflitti presenti utilizzando gli STALLI e calcolare il CPI

```
ADDI    R1, ZERO, 2
ADDI    R2, ZERO, 7
TAG: LW   R3, 0(R4)
ADD     R3, R3, R2
SW      R3, 0(R4)
ADDI    R4, R4, 8
SUBI    R1, R1, 1
BNEQ    R1, ZERO, TAG
NOP
```

CICLO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
ADDI1	F	D	E	M	W																
ADDI2		F	D	E	M	W															
LW3			F	D	E	M	W														
ADD4				F	X	X	D	E	M	W											
SW5							F	X	X	D	E	M	W								
ADDI6										F	D	E	M	W							
SUBI7											F	D	E	M	W						

Dipendenza RAW su R2 tra ADDI2 e ADD4

Dipendenza RAW su R3 tra ADD4 e SW5

Dipendenza RAW su R1 tra ADD1 e SUBI7 già risolta...

...



# Esercizio 4: Risolvere i conflitti presenti utilizzando gli STALLI e calcolare il CPI

```
ADDI    R1, ZERO, 2
ADDI    R2, ZERO, 7
TAG:    LW     R3, 0(R4)
ADD     R3, R3, R2
SW      R3, 0(R4)
ADDI    R4, R4, 8
SUBI    R1, R1, 1
BNEQ    R1, ZERO, TAG
NOP
```

CICLO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
ADDI1	F	D	E	M	W																
ADDI2		F	D	E	M	W															
LW3			F	D	E	M	W														
ADD4				F	X	X	D	E	M	W											
SW5							F	X	X	D	E	M	W								
ADDI6										F	D	E	M	W							
SUBI7											F	D	E	M	W						
BNEQ8												F	X	X	D	E	M	W			

Dipendenza RAW su R2 tra ADDI2 e ADD4

Dipendenza RAW su R3 tra ADD4 e SW5

Dipendenza RAW su R1 tra ADD1 e SUBI7 già risolta

Dipendenza RAW su R1 tra SUBI7 e BNEQ8

ora interciclo...

# Esercizio 4: Risolvere i conflitti presenti utilizzando gli STALLI e calcolare il CPI

```
ADDI    R1, ZERO, 2
ADDI    R2, ZERO, 7
TAG:    LW     R3, 0(R4)
ADD     R3, R3, R2
SW      R3, 0(R4)
ADDI    R4, R4, 8
SUBI    R1, R1, 1
BNEQ    R1, ZERO, TAG
NOP
```

CICLO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
ADDI1	F	D	E	M	W																
ADDI2		F	D	E	M	W															
LW3			F	D	E	M	W														
ADD4				F	X	X	D	E	M	W											
SW5							F	X	X	D	E	M	W								
ADDI6										F	D	E	M	W							
SUBI7											F	D	E	M	W						
BNEQ8												F	X	X	D	E	M	W			
LW3																		F	D	E	M

interciclo:

Dipendenza RAW su R4 tra ADDI6 e LW3 già risolta.

# Esercizio 4: Risolvere i conflitti presenti utilizzando gli STALLI e calcolare il CPI

```
ADDI    R1, ZERO, 2
ADDI    R2, ZERO, 7
TAG:    LW     R3, 0(R4)
ADD     R3, R3, R2
SW      R3, 0(R4)
ADDI    R4, R4, 8
SUBI    R1, R1, 1
BNEQ    R1, ZERO, TAG
NOP
```

CICLO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
ADDI1	F	D	E	M	W																
ADDI2		F	D	E	M	W															
LW3			F	D	E	M	W														
ADD4				F	X	X	D	E	M	W											
SW5							F	X	X	D	E	M	W								
ADDI6										F	D	E	M	W							
SUBI7											F	D	E	M	W						
BNEQ8												F	X	X	D	E	M	W			
LW3																		F	D	E	M

interciclo:

Dipendenza RAW su R4 tra ADDI6 e LW3 già risolta.

Dipendenza di controllo su BNEQ8:

Come detto "o stalli nella pipeline (3 cicli)"

# Esercizio 4: Risolvere i conflitti presenti utilizzando gli STALLI e calcolare il CPI

CICLO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
ADDI1	F	D	E	M	W																
ADDI2		F	D	E	M	W															
LW3			F	D	E	M	W														
ADD4				F	X	X	D	E	M	W											
SW5							F	X	X	D	E	M	W								
ADDI6										F	D	E	M	W							
SUBI7										F	D	E	M	W							
BNEQ8											F	X	X	D	E	M	W				
LW3																		F	D	E	M

CICLO	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
BNEQ8	W														
LW3	F	D	E	M	W										
ADD4		F	X	X	D	E	M	W							
SW5					F	X	X	D	E	M	W				
ADDI6								F	D	E	M	W			
SUBI7									F	D	E	M	W		
BNEQ8										F	X	X	D	E	M
NOP9															

Proseguendo... Ritorna:

Dipendenza RAW su R3 tra LW3 e ADD4

# Esercizio 4: Risolvere i conflitti presenti utilizzando gli STALLI e calcolare il CPI

CICLO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
ADDI1	F	D	E	M	W																
ADDI2		F	D	E	M	W															
LW3			F	D	E	M	W														
ADD4				F	X	X	D	E	M	W											
SW5							F	X	X	D	E	M	W								
ADDI6										F	D	E	M	W							
SUBI7										F	D	E	M	W							
BNEQ8											F	X	X	D	E	M	W				
LW3																		F	D	E	M

Completo:

CICLO	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37
BNEQ8	W																			
LW3	F	D	E	M	W															
ADD4		F	X	X	D	E	M	W												
SW5					F	X	X	D	E	M	W									
ADDI6								F	D	E	M	W								
SUBI7									F	D	E	M	W							
BNEQ8										F	X	X	D	E	M	W				
NOP9																F	D	E	M	W

## Esercizio 4: calcolare il CPI

---

CPI:

$$CPI = \frac{\# \text{ TOTALE CICLI}}{\# \text{ TOTALE ISTRUZIONI}} = \frac{37}{15} = 2.47$$

# Esercizio 5:

---

```
LOOP:LW    R2, A(R3)
      ADDI  R5, R5, 8
      SW    R5, A(R4)
      ADDI  R3, R3, R4
      BNEQ  R3, R8, LOOP
      ADD   R2, R2, R5
```

Con  $R3 = 0$ ,  $R4 = 1$  e  $R8 = 2$

1. Identificare le dipende del seguente codice
2. Risolvere i conflitti presenti utilizzando gli STALLI e calcolare il CPI

# Esercizio 5: dipendenze

---

```
LOOP: LW    R2, A(R3)
      ADDI   R5, R5, 8
      SW     R5, A(R4)
      ADDI   R3, R3, R4
      BNEQ   R3, R8, LOOP
      ADD    R2, R2, R5
```

Con R3 = 0, R4 = 1 e R8 = 2

Dipendenza RAW su R2 tra LW1 e ADD6

Dipendenza RAW su R5 tra ADDI2 e SW3

Dipendenza RAW su R5 tra ADDI2 e ADD6

Dipendenza RAW su R3 tra ADDI4 e BNEQ5

Dipendenza RAW su R5 tra ADDI2 e ADDI2 ( Interciclo )

Dipendenza RAW su R3 tra ADDI4 e LW1 ( Interciclo )

Dipendenza RAW su R3 tra ADDI4 e ADDI4 ( Interciclo )

Dipendenza di controllo su BNEQ5



# Esercizio 5: Risolvere i conflitti presenti utilizzando gli STALLI

```
LOOP: LW    R2, A(R3)
      ADDI   R5, R5, 8
      SW     R5, A(R4)
      ADDI   R3, R3, R4
      BNEQ   R3, R8, LOOP
      ADD    R2, R2, R5
```

CICLO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
LW1	F	D	E	M	W																
ADDI2		F	D	E	M	W															
SW3			F	X	X	D	E	M	W												
ADDI4						F	D	E	M	W											
BNEQ5							F	X	X	D	E	M	W								
LW1												F	D	E	M	W					
ADDI2													F	D	E	M	W				
SW3														F	X	X	D	E	M	W	
ADDI4																	F	D	E	M	
BNEQ5																		F	X	X	

CICLO	18	19	20	21	22	23	24	25	26	27	28	29
ADDI4	F	D	E	M	W							
BNEQ5		F	X	X	D	E	M	W				
ADD6								F	D	E	M	W