

Istruzioni

Mon, 28 Mar

Un'istruzione è una sequenza di 32 *bit* (nell'architettura MIPS-32) interpretata dalla CPU in modo da definire un'operazione.

Un'istruzione si dice **allineata** alla parola (*word*) quando il suo **indirizzo** inizia da `*00` e finisce a `*11`, avendo la necessità di *incrementare* dunque solo i **primi due bit** del suo *indirizzo*.

Forma

I **6 bit** dal 26 a 31 sono detti **C.O.** (*Codice Operativo*) e definiscono il tipo di *operazione*.

La restante parte della sequenza di bit varia in base alle *esigenze* dell'istruzione.

Esistono due tipi di istruzioni:

▼ Register [R]

Sono le operazioni di tipo **aritmetico-logico** e operano sui *registri* [*Esempio: ADD*]

Si presentano nella forma:

```
[ CO | RS | RT | RD | Shift Amount | FUN ]
```

Il CO vale sempre 0, e i primi bit a 0 a 6 indicano la **FUNCTION**, differenziando le *operazioni*. I 15 bit (a gruppi di 5) dopo il CO, indicano gli indirizzi degli *operandi* (*Source, Target, Destination*).

▼ Esempio di istruzione:

```
ADD $6 $7 $8
```

▼ Immediate [I]

L'operazione (come ad esempio la ADDi) utilizza come *operandi* la *Source* e il valore segnato dal campo *Immediato*.

Si presentano nella forma:

```
[ CO | RS | RT | Imm ]
```

Il CO indica (normalmente) l'operazione da eseguire.

▼ Esempio di istruzione:

```
ADDi $6 $7 193
```

Esempi di Istruzioni

Branch not Equal

```
BNE $i $j label
```

```
[ CO | $i | $j | Imm ] 0 | 0
```

I due zeri alla fine "moltiplicano" per 4 l'offset, generando un'indirizzo *allineato alla word*.

Jump

```
JMP label
```

```
[ CO | ADDR ] 0 | 0
```

L'indirizzo indicato è *assoluto*, ovvero non calcolato tramite offset.

Jump and Link

```
JAL label
```

Viene salvato del registro **Return Address** (RA) l'indirizzo dell'istruzione da eseguire *dopo* aver completato la *procedura*.