

CORSO DI LAUREA IN INGEGNERIA INFORMATICA

Calcolatori — a.a. 2018–2019

Compito del 15 aprile 2019

Cognome e Nome dello studente: _____

Microprocessore Un micro byte-addressable con architettura a singolo bus interno, memoria non segmentata, e dati e indirizzi a 16 bits, annovera nel suo instruction set l'istruzione

load <op>, <reg> .

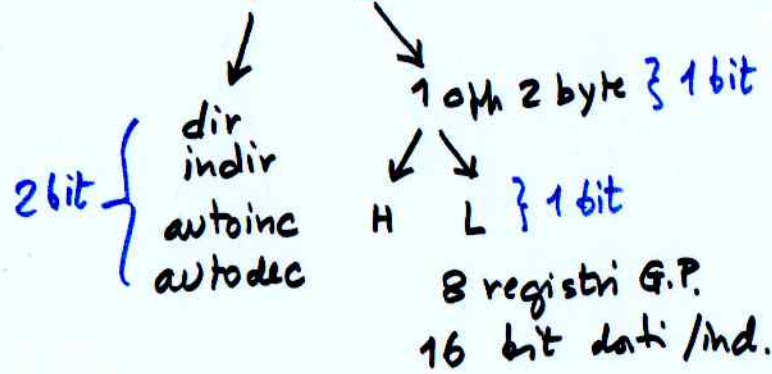
L'istruzione carica nel registro destinazione <reg> un dato di memoria specificato dall'operando <op>. Il tipo di dato da trasferire è di 1 o 2 bytes, in relazione al tipo di registro destinazione impiegato nell'istruzione. Infatti, l' i -simo registro di uso generale R_i ($i = 0, 1, \dots, 7$), di 2 bytes, è riguardabile come composto da due registri da 1 byte: $R_i.H$ (bits 15-8) e $R_i.L$ (bits 7-0). L'operando <op> è esprimibile in uno dei seguenti modi di indirizzamento: (1) diretto di memoria, (2) indiretto di registro, (3) autoincremento, (4) autodecremento. Il modo autoincremento è simile all'indiretto di registro, salvo che *dopo* la lettura in memoria *il registro puntatore dev'essere incrementato del numero di bytes letti*. Analogamente, nel modo autodecremento il registro puntatore dev'essere decrementato del numero di bytes da leggere *prima* di effettuare l'operazione di lettura.

Esempi di sintassi: load VAR, R3.H; load [R6], R1; load [R5]+, R2 (autoincremento); load -[R3], R4.L (autodecremento).

Dopo aver disegnato la sezione di parte operativa strettamente necessaria, fornire una codifica plausibile dell'istruzione e disegnare l'automa di controllo relativo alla sua esecuzione, evidenziando con chiarezza e completezza ingressi e uscite della parte di controllo in ogni stato (ciò sarà alla base della soluzione dell'esercizio successivo).

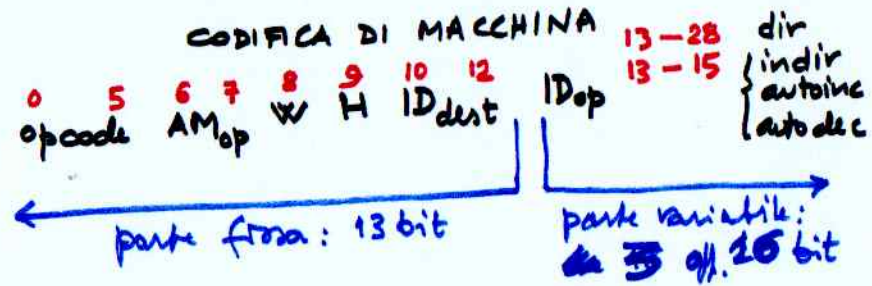
Reti Logiche Progettare l'hardware di controllo per l'automa disegnato nell'esercizio precedente (dell'automa devono fare parte anche i due stati di partenza dal fetch e di rientro al fetch) secondo il classico schema "monoblocco": funzione di transizione di stato f , registro di stato M , funzione di uscita g (macchina di Moore). In particolare, specificare la funzione f attraverso il metodo orientato alla sintesi con multiplexer, cioè scrivendo una diversa tabella per ciascuno stato S_i dell'automa ($i = 1 \dots n$, con n numero degli stati della macchina). La tabella i -sima deve specificare lo stato futuro in funzione dei soli ingressi campionati in S_i . Infine, valutare il costo approssimativo del progetto assegnando un costo p a ciascuna porta AND e OR, e un costo r ad ogni suo ingresso (ad esempio, una porta a 3 ingressi ha un costo complessivo $p + 3r$).

Load op, reg



IDop : { 16 bit
3 bit

IDdest : 3 bit



ALGORITMO

AM: diretto 00

IDop → MAR, read
MDR → dest

AMop: indirutto 01

IDop → RF → MAR, read
MDR → dest

AMop: autoinc 10

IDop → RF → { MAR, read
Y

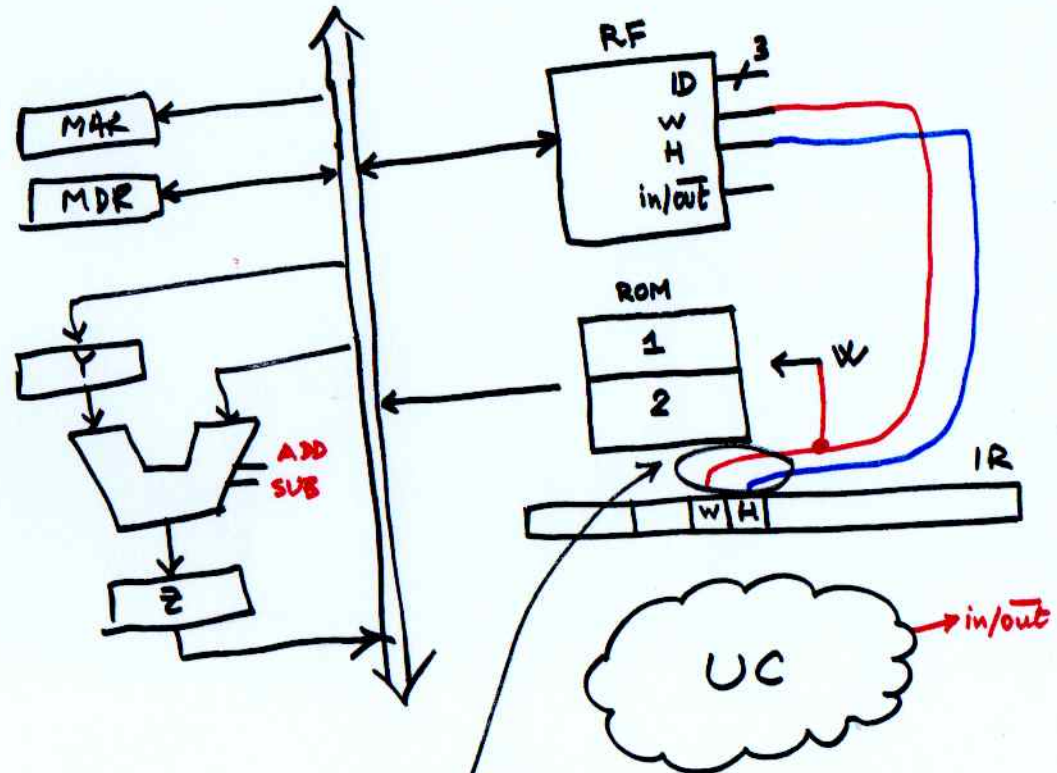
(1 v 2) → bus interno, add
ROM(W)

AMop: auto dec 11

IDop → RF → Y
ROM(W) → bus int., sub
Z → { MAR, read

Leggi:
RF(IDdest)

stati
unificabili
(cfr.
diagn.
stati)



buffer
tristate!
(c'è ma non è mostrato)

