

MICROPROGRAMMA : SALTI JUMP;

sabato 26 novembre 2022 — 11:36

SALTO CONDIZIONATO : AGGIORNANO IL CONTENUTO dell' IP SE E SOLO SE SI VERIFICA UNA determinata condizione;

Come faccio a farlo a microcodice ? come faccio a dire che se si verifica una determinata condizione allora esegui un pezzo di microcodice?

Possiamo alterare il flusso di controllo del nostro microprogramma ;

"SE ad esempio $ZF == 1$, allora vai alla microoperazione successiva, altrimenti esci da questo microprogramma"

Quindi il nostro microprogramma di JZ, jump solo se $ZF == 1$, può fare un controllo di tipo IF, perché siamo in grado, all'interno del nostro microcodice, di fare dei salti IN AVANTI tra microoperazioni.

- Fetch →
- jz displacement:
 - MAR \leftarrow RIP
 - MDR \leftarrow (MAR); RIP \leftarrow RIP + 8
 - IR \leftarrow MDR
 - IF FLAGS[ZF] == 1 THEN
 - TEMP1 \leftarrow RIP
 - TEMP2 \leftarrow IR[0:31]
 - RIP \leftarrow ALU OUT[ADD]
 - ENDIF

Dopo la fase di fetch faccio un controllo su ZF , e quella sarà una variabile di condizione che entra nella mia ROM.

La mia ROM avrà una tabella di verità tale per cui: se $ZF=0$ allora salta quella microoperazione

Se $ZF == 1$, lo spiezzamento fa due somme al valore dell' IP dopo la fase di fetch.

Per implementare questo salto due appo il contenuto di RIP è sommato in tempo. Cosa lo spiezzamento, che sta nei 32 bit meno significativi dell' IP, in tempo.

IL RISULTATO della somma LO SONO IN RP
↳ Registro a 64 bit

Granidateza 32 bit

The diagram shows a horizontal line representing a 32-bit displacement. It is divided into two segments by a vertical line, each labeled "32 bit". A green bracket underneath the line indicates the split point. An arrow points from the text "IL RISULTATO della somma LO SONO IN RP" to the RP register, which is labeled "64 bit".

Significa che noi possiamo raggiungere qualsiasi punto della memoria. E salto.

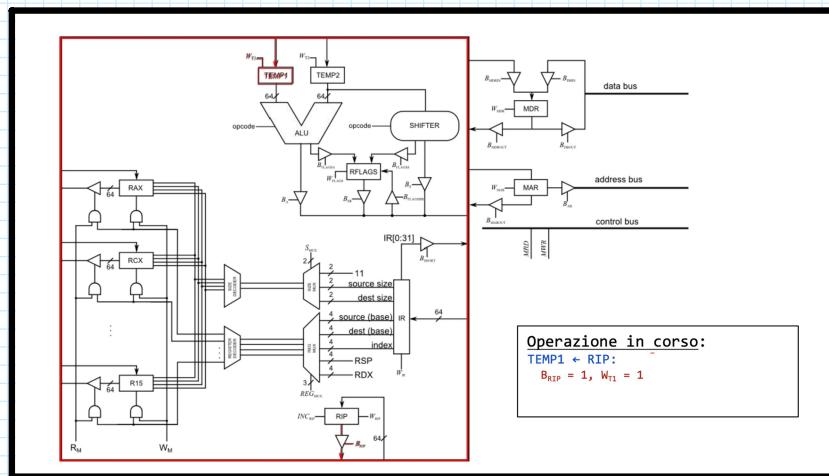
Se parto da 0 non posso raggiungere tutti i 64 bit.

memoria, se salto.

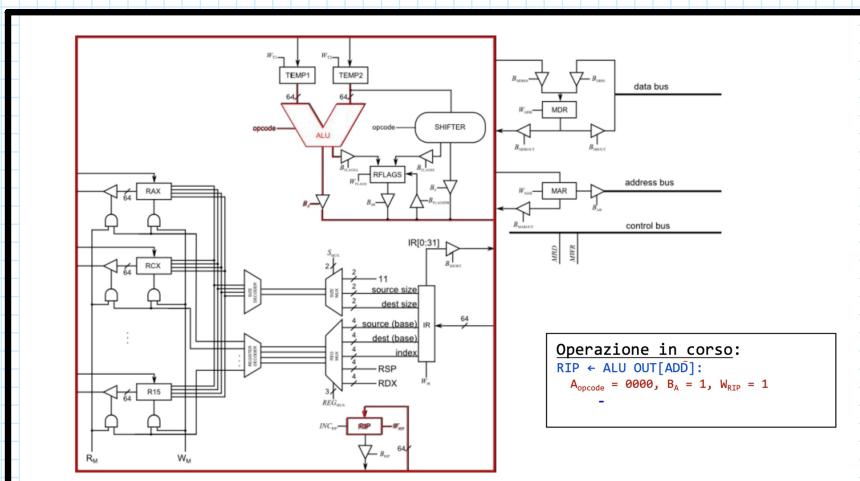
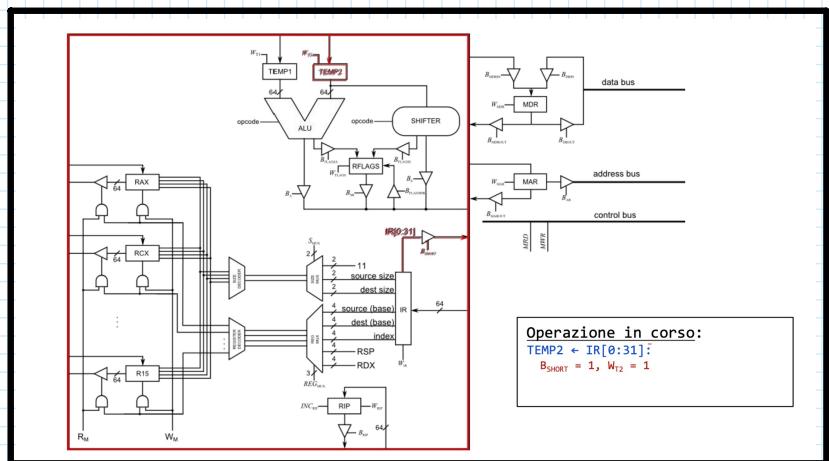
Se parto da 0 non posso indicizzare tutti i 64 bit.

È RAPIDO CHE IO HO 2^{32} byte per rappresentare i programmi;

se voglio effettuare un salto molto grande, lo faccio due volte;
uso uno Incremento;



Non scrivo su flags,
perché sto effettuando
un salto;



FINE ARCHITETTURA