

Calcolatori Elettronici L A

Prova scritta del 12/9/'03

Esercizio N. 1 (11 punti)

Si vuole estendere il set di istruzioni del DLX *sequenziale* con l'istruzione:

WSWP [Ra], [Rb]

(Word SWaP) che consente di scambiare in memoria le parole contenute agli indirizzi puntati dalla coppia di registri Ra ed Rb. Prima di terminare, la stessa istruzione incrementa entrambi i puntatori contenuti nei registri all'indirizzo delle locazioni successive.

1.1) si mostri come potrebbe essere codificata la nuova istruzione "WSWP [R25], [R7]" utilizzando il formato che si ritiene più opportuno tra quelli delle istruzioni del DLX. (2 punti)

1.2) con riferimento al datapath *sequenziale* del DLX visto a lezione e coerentemente con la risposta data al punto 1.1, si disegni il diagramma degli stati il più possibile ottimizzato che controlla l'esecuzione della nuova istruzione, completo degli stati di fetch e decodifica. (7 punti)

1.3) Si determini il CPI della nuova istruzione, considerato che ogni accesso in memoria richiede in totale 2 periodi di clock. (2 punti)

Esercizio N. 2 (13 punti)

Un sistema a microprocessore basato su 8088 a 5 MHz dispone di 64KB di Eprom (1 chip agli indirizzi alti) e 256KB di Ram (2 chip contigui a partire dall'indirizzo 0h). Il sistema riceve dati dalla porta A di un'interfaccia parallela 8255 programmata in modo I-input e gestita ad interrupt. Si supponga di disporre di un contatore di fronti positivi "PETC" dotato di ingresso di reset del conteggio "RST#", che al 65535-esimo fronte ricevuto sull'ingresso "TRIG" attiva la sua uscita "DONE" fino alla ricezione del fronte positivo seguente, in conseguenza del quale disattiva "DONE" e ricomincia a contare da 0. Progettando una opportuna rete logica, interfacciare il contatore ai segnali del bus di sistema in modo tale che, senza gravare sulla CPU, "PETC" controlli che il μP abbia letto 64K-caratteri dall'8255, prima di inviare una richiesta di interrupt all'8259. Come risposta alla richiesta di interrupt il μP accenderà un led collegato ad un FF-D mappato all'indirizzo FAH dello spazio di I/O e collegato a BD0.

2.1) Progettare il circuito per la generazione del segnale di READY nell'ipotesi che né le memorie, né i dispositivi di I/O richiedano periodi di wait. (1 punto)

2.2) Disegnare lo schema del progetto includendo anche "PETC" ed il led collegato al FF-D e facendo in modo che all'avvio del sistema il led sia spento e il contatore sia impostato a "0". (8 punti)

2.3) Scrivere le espressioni di selezione semplificate delle memorie e dei dispositivi di I/O. (3 punti)

2.4) Scrivere la coppia di istruzioni (contenuta nella routine di risposta all'interrupt) necessaria ad accendere il led. (1 punto)

Esercizio N. 3 (9 punti)

3.1) Descrivere le funzioni dello stadio EX del datapath del DLX pipelined in funzione delle varie classi di istruzioni. (3 punti)

3.2) Dire perché nel datapath del DLX è conveniente già in "ID" caricare su A e B i valori di RS1 e RS2, nei formati di istruzioni R e I. (3 punti)

3.3) Descrivere l'uso accoppiato delle istruzioni "Set condition" e "Branch" nell'ISA del DLX. (3 punti)

SCRIVERE IN MODO LEGGIBILE

Durante il compito non si può uscire dopo le ore 11:00

Consegnando un elaborato per la correzione decade la validità del voto precedente

Correzione e registrazione il 19/9/03, ore 10:00 Via Rasi

I risultati, appena disponibili, saranno pubblicati in Segreteria e sul sito:

<http://www.ing2.unibo.it> → Didattica → Lauree Triennali (selez. il proprio C.d.L.) → piano degli studi → Calcolatori Elettronici L A

SOLUZIONE del compito di Calcolatori Elettronici L A del 12/9/03

1.1) (2 punti) si possono utilizzare entrambi i formati I oppure R. Segue la codifica dell'esempio del testo:

| Opcode (6bit) | Rs1 (5bit) | Rs2 (5bit) | Offset pc-relative (16bit) |
|---------------|------------|------------|----------------------------|
| WSWP | 25 | 7 | Inutilizzati |

1.3) (2 punti) Nel diagramma di WSWP mostrato a fianco si contano 12 stati, dei quali 5 comportano accessi in memoria:

$$CPI = 7 + (2 \cdot 5) = 17 Tck$$

2.1) (1 punto) Vedere schema nella pagina seguente.

2.2) (8 punti) Vedere schema nella pagina seguente.

2.3) (3 punti) Espressioni di selezione semplificate.

memorie:

$$CSE0 = BA19 \cdot !IO/M\#$$

$$CSR0 = !BA19 \cdot !BA17 \cdot !IO/M\#$$

$$CSR1 = !BA19 \cdot BA17 \cdot !IO/M\#$$

periferiche di I/O:

$$CS8259 = BA15 \cdot IO/M\#$$

$$CS8255 = !BA15 \cdot BA14 \cdot IO/M\#$$

$$CSLED = !BA15 \cdot !BA14 \cdot IO/M\# \text{ (accessibile ad FAH)}$$

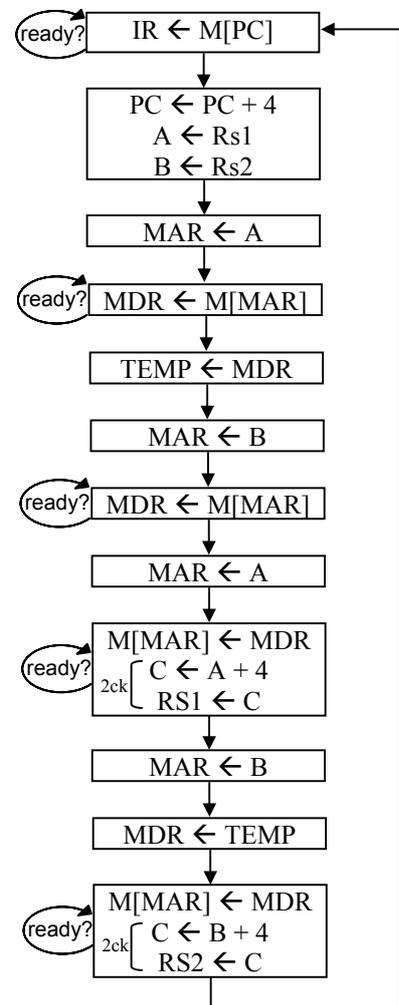
2.4) (1 punto) Le istruzioni sono MOV AL, 01H e OUT FAH, AL

3.1). (3 punti) Nello stadio EX del DLX pipelined la ALU esegue i calcoli di tutte le istruzioni ALU (o aritmetico-logiche: AND, ADD,...). Per le istruzioni di tipo LOAD/STORE, la ALU calcola l'indirizzo dell'accesso in memoria ed eventualmente l'estensione del segno nel caso di offset negativo. Per i BRANCH, in ID si controlla la validità della condizione e si calcola l'indirizzo dell'istruzione a cui saltare se la verifica risulterà positiva.

3.2). (3 punti) Una caratteristica dei formati di istruzione R ed I del DLX è quella di contenere gli indirizzi per l'accesso al Register File sempre nella stessa posizione della codifica a 32 bit dell'istruzione. Ciò consente di salvare sui registri di ingresso della ALU, A e B, il contenuto del RF a prescindere dal fatto che esso sia o meno utile per il completamento dell'istruzione. Tale operazione viene eseguita mentre l'ALU incrementa il Program Counter e occupa quindi risorse non coinvolte da quest'ultima operazione: in questo modo si guadagna tempo anticipando due operandi nei registri all'ingresso dell'ALU che molto probabilmente saranno utilizzati proprio nello stato successivo.

3.3). (3 punti) Nel rispetto dell'architettura RISC, con le semplici SET+BRANCH l'ISA del DLX consente di implementare un costrutto diffuso in parecchi Assembler CISC: Compare and Branch. Il veicolo per il passaggio di informazione da SET a BRANCH è un registro generale e non un flag dedicato, anche questo in conformità alla non-specificità delle risorse, caratteristica dei processori RISC. Nell'uso combinato SET+BRANCH l'istruzione SET controlla la condizione specificata nella sua sintassi e setta registro generale conseguentemente all'esito della verifica. Successivamente la BRANCH esegue il trasferimento di controllo in funzione del valore letto in tale registro. L'esistenza di due istruzioni distinte consente ovviamente di combinarle anche con altre dell'ISA del DLX per scopi diversi.

1.2) (7 punti) soluzione proposta



Schema compito Calcolatori Elettronici del 12/9/'03 - Domande 2.1 (1 punto), 2.2 (8 punti)

