

Compito del 22 novembre 2002

SCELTA DI ESERCIZI

¶ In un'architettura 8086 l'indirizzo fisico di una locazione di memoria è dato da $PA=SR \times 16 + EA$. Si determini (usando la rappresentazione in complemento a due) il valore di SR, nel caso in cui $PA=53394H$ e $EA=70B4H$.

¶ Determinare la rappresentazione in base 8 del numero decimale 1724. Esprimere il risultato in esadecimale.

¶ Determinare la rappresentazione esadecimale del numero decimale 1492. Esprimere il risultato in base 8.

¶ Determinare la rappresentazione in complemento a 2 del numero decimale -578 e sommarli l'intero positivo 5C4H. Esprimere il risultato in decimale.

¶ Data la funzione logica di 4 variabili $y(x_1, x_2, x_3, x_4) = x_2 \cdot (x_1 + \bar{x}_3 \cdot \bar{x}_4) + \bar{x}_2 \cdot \bar{x}_3 \cdot (\bar{x}_1 + x_4)$, (1) rappresentarla nella forma di tabella della verità; (2) riscriverla quindi nella forma canonica somma di prodotti.

¶ Data la funzione logica di 4 variabili $y(x_1, x_2, x_3, x_4) = [x_4 + x_2 \cdot (x_1 + x_3)] \cdot [(x_2 + x_4) \cdot (\bar{x}_1 + \bar{x}_3)]$, (1) rappresentarla nella forma di tabella della verità; (2) riscriverla quindi nella forma canonica prodotto di somme.

¶ Data la funzione logica di 4 variabili $y(x_1, x_2, x_3, x_4) = x_2 \cdot (x_1 + \bar{x}_3 \cdot \bar{x}_4) + \bar{x}_2 \cdot \bar{x}_3 \cdot (\bar{x}_1 + x_4)$, (1) rappresentarla nella forma di tabella della verità; (2) riscriverla quindi nella forma canonica prodotto di somme.

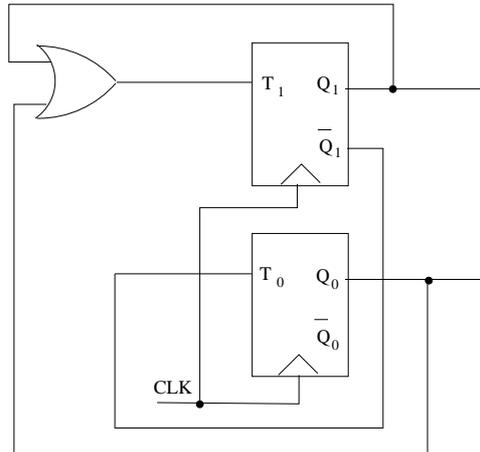
¶ Data la funzione logica di 4 variabili $y(x_1, x_2, x_3, x_4) = [x_4 + x_2 \cdot (x_1 + x_3)] \cdot [(x_2 + x_4) \cdot (\bar{x}_1 + \bar{x}_3)]$, (1) rappresentarla nella forma di tabella della verità; (2) riscriverla quindi nella forma canonica somma di prodotti.

¶ Data l'espressione logica di 4 variabili $x_2 \cdot (x_1 + \bar{x}_3 \cdot \bar{x}_4) + \bar{x}_2 \cdot \bar{x}_3 \cdot (\bar{x}_1 + x_4)$, verificarne l'uguaglianza con l'espressione in somma di prodotti $\bar{x}_1 \bar{x}_2 x_3 \bar{x}_4 + \bar{x}_1 \bar{x}_2 x_3 x_4 + \bar{x}_1 x_2 \bar{x}_3 \bar{x}_4 + x_1 \bar{x}_2 x_3 \bar{x}_4 + x_1 x_2 \bar{x}_3 \bar{x}_4 + x_1 x_2 x_3 \bar{x}_4 + x_1 x_2 x_3 x_4$.

¶ Data l'espressione logica di 4 variabili $[x_4 + x_2 \cdot (x_1 + x_3)] \cdot [(x_2 + x_4) \cdot (\bar{x}_1 + \bar{x}_3)]$, verificarne l'uguaglianza con l'espressione in prodotto di somme $(x_1 + x_2 + x_3 + x_4) \cdot (x_1 + x_2 + \bar{x}_3 + x_4) \cdot (x_1 + x_2 + \bar{x}_3 + x_4) \cdot (x_1 + \bar{x}_2 + x_3 + x_4) \cdot (\bar{x}_1 + \bar{x}_2 + \bar{x}_3 + x_4) \cdot (\bar{x}_1 + x_2 + x_3 + x_4) \cdot (\bar{x}_1 + x_2 + \bar{x}_3 + \bar{x}_4) \cdot (\bar{x}_1 + \bar{x}_2 + \bar{x}_3 + x_4)$.

¶ Disegnare il diagramma degli stati del flip-flop JK e darne la tabella di transizione di stato.

¶ L'uscita della macchina sequenziale sincrona in figura è data dalla parola Q_0Q_1 ($Q_1 = \text{LSB}$). Al tempo $t = 0$, subito prima dell'arrivo del primo colpo di clock, la parola di uscita vale $P_0 = (00)_2 = (0)_{10}$. Stabilire il valore decimale di P_i , $i = 1, \dots, 10$. Commentare il risultato.



¶ Una CPU a singolo bus interno ha registri a 16 bit, ed è connessa con l'esterno attraverso un bus dati a 8 bit (e registro MDR) e un bus indirizzi a 16 bit (MAR). Scrivere la microsequenza di fetch dell'istruzione $\text{MOVE VAR}[R_3], R_6$. Si ipotizzi che opcode e modi di indirizzamento siano codificati nel primo byte dell'istruzione. (N.B.: Il numero di micropassi necessari può differire da quello riportato di seguito.)

- 1.
- 2.
- 3.
- 4.
- 5.

¶ Tra le seguenti affermazioni sullo stack dell'8086 selezionare quelle vere:

- l'indirizzo fisico dell'ultima locazione scritta è SS:SP
- ci si accede solo per word a indirizzi pari
- ci si accede per word a indirizzi pari o dispari
- ci si può accedere con l'istruzione MOV
- il registro BP punta alla prima locazione libera

¶ In un sistema 8088, un'interfaccia di sola uscita è mappata su due indirizzi di porta consecutivi (dati, controllo) a partire da FAH. Lo stato è leggibile ad entrambi gli indirizzi. Il contenuto della parola di stato (8 bit) è $SxxxxC_2C_1C_0$, dove $S=1$ indica dispositivo occupato e i tre bit meno significativi forniscono il contenuto della parola di controllo. Scrivere

di seguito la sequenza assembly che consenta di leggere in AL lo stato, e di porre a 10 i bit di controllo C_1C_0 , lasciando inalterato il bit C_2 . (N.B.: ci sono varie soluzioni, cui corrisponde un tipo e un numero diverso di istruzioni utilizzate.)

-
-
-
-
-

¶ La codifica in linguaggio macchina della classe di istruzioni 8086 “MOV register/memory to/from register” (dove register sta per un qualsiasi registro non di segmento) è su 4 byte:

100010dw	mod reg r/m	<DISP-LO>	<DISP-HI>
----------	-------------	-----------	-----------

. Utilizzare la tabella dei modi di indirizzamento sottostante per codificare in esadecimale l’istruzione `MOV AX, VAR[SI]`, dove VAR è uno spiazzamento a 16 bit. Qual è il registro di segmento di default per l’accesso in memoria? Supponendo che l’istruzione sia posta in memoria a un indirizzo pari, calcolare inoltre il numero di accessi al bus richiesti per completarne il fetch.

r/m	mod				reg	
	00	01	10	11		
000	BX+SI	BX+SI+D8	BX+SI+D16	AL	AX	000
001	BX+DI	BX+DI+D8	BX+DI+D16	CL	CX	001
010	BP+SI	BP+SI+D8	BP+SI+D16	DL	DX	010
011	BP+DI	BP+DI+D8	BP+DI+D16	BL	BX	011
100	SI	SI+D8	SI+D16	AH	SP	100
101	DI	DI+D8	DI+D16	CH	BP	101
110	D16	BP+D8	BP+D16	DH	SI	110
111	BX	BX+D8	BX+D16	BH	DI	111
				w=0	w=1	

¶ Di ciascuna delle dichiarazioni assembly 8086 sotto riportate indicare il significato e il valore dello spiazzamento all’interno del segmento:

spiazzamento	dichiarazione	significato
	<code>dati SEGMENT DATA PUBLIC</code>	
	<code>string DB 'a', 'b', 'c', 'd'</code>	
	<code>etich LABEL BYTE</code>	
	<code>length EQU etich - string</code>	
	<code>buffer_1 DW 4 DUP (0)</code>	
	<code> DW 1234</code>	
	<code>var_1 DW FFH</code>	
	<code>var_2 EQU FFH</code>	
	<code>buffer_2 DB 4 DUP("abcd")</code>	
	<code>dati ENDS</code>	

¶ Per il seguente segmento dati in assembly 8086, che si assume mappato in memoria all'indirizzo *C54D0*, riportare in ordine i dati effettivamente allocati in memoria e il loro valore di inizializzazione:

```

dati SEGMENT DATA PUBLIC
Bstring DB
'a','b','c','d'
Wstring DW
'1','2','3','4'
buffer DB 2 DUP (?)
etich LABEL BYTE
var_1 DW 1234
var_2 EQU 1234
dati ENDS

```

	0	1	2	3
C54D0				
C54D4				

¶ Dato il seguente spezzone di codice in assembly 8086, indicare (1) in corrispondenza di quali istruzioni il registro SP viene aggiornato, e (2) annotare a fianco delle istruzioni di cui sopra il nuovo valore esadecimale di SP, assumendo che il suo valore iniziale sia 3200H.

```

push ax ;
mov ax,0009h ;
int 21h ;
mov dx, OFFSET var ;
push dx ;
call NEAR PTR subrout ;
pop dx ;
pop ax ;
mov var,dx ;

```