027145310013 030276124540 014426113420 004133317526 021734510013 002153407254 010426114342

Primo Scritto Parziale di Architettura degli Elaboratori

A.A. 1998/99, 18-12-98

Cognome: .... Nome: .... Login: 199...s.........

Domande

1. (punti: -1, 4)
   In un modulo RAM con N celle, ciascuna di k bit, per indirizzare le celle si usa un decoder con P ingressi (e P é il numero minimo di ingressi che permette l'indirizzamento di tutte le celle); inoltre si usano dei dispositivi a 3 stati per raggruppare i fili di ingresso e di uscita dei dati. Un altro modulo RAM, costruito nello stesso modo, ma con $6 \cdot N$ celle da $2 \cdot k$ bit:

   (a) avrà un decoder con $6 \cdot P$ ingressi;

   (b) avrà un decoder con $8 \cdot P$ ingressi;

   (c) avrà un decoder con $12 \cdot P$ ingressi;

   (d) utilizzerà $6 \cdot N$ dispositivi a 3 stati;

   (e) utilizzerà $12 \cdot N$ dispositivi a 3 stati;

   (f) nessuna delle affermazioni precedenti é corretta.

2. (punti: -1, 4)
   Nella macchina VM2:

   (a) si usano solo gli indirizzamenti: diretto, immediato, indicizzato;

   (b) il modo indicizzato utilizza sempre il registro FP;

   (c) esistono istruzioni che non hanno operandi (impliciti, o espliciti);

   (d) la codifica delle istruzioni usa un formato variabile;

   (e) non é previsto il modo "indiretto mediante cella di memoria";

   (f) nessuna delle affermazioni precedenti é corretta.

3. (punti: -1, 4)
   Nella macchina VM2, é opportuno usare la tecnica del "busy waiting" per effettuare la lettura dell'input per i seguenti motivi:

   (a) dare tempo ai circuiti del registro 4094 di assestarsi e fornire un valore stabile;

   (b) tener conto del tempo necessario a trasferire i dati dal registro 4094 all'accumulatore;

   (c) perché non é possibile leggere il contenuto del registro 4094, se il bit del segno é uguale a 1 (il codice ASCII contiene solo numeri positivi);

   (d) perché questa tecnica costringe il programma ad aspettare fino a quando l'utente non ha digitato un carattere e a leggerne la codifica dal registro 4094 una volta sola;

   (e) per evitare di effettuare simultaneamente operazioni di lettura e di scrittura (visto che nei registri non sono previsti dispositivi a tre stati);

   (f) nessuna delle affermazioni precedenti é corretta.

4. Consideriamo un codice di Hamming a 7 bit, in cui i bit 0, 1, 2, 3 sono bit di informazione ed i bit 4, 5, 6 sono bit di parità e dove:

- il bit 4 controlla i bit 0, 1, 3
- il bit 5 controlla i bit 1, 2, 3
- il bit 6 controlla i bit 0, 1, 2

(a) (punti: 3)
Determinare il valore dei bit di parità nel caso: bit0 = 0 , bit1 = 1 , bit2 = 0 , bit3 = 1;
risposta: bit4 = , bit5 = , bit6 =

(b) (punti: 3)
La configurazione 0001001 (qui il bit 0 é ovviamente quello più' a destra) é stata ottenuta da una configurazione corretta modificando un solo bit. Qual'é la configurazione di partenza (quella corretta)?
Risposta: ....

5. Consideriamo una rappresentazione per i numeri razionali in formato fisso su 16 bit suddivisi nei seguenti 3 campi:

- campo S, formato dal bit 15
- campo EXP, formato dai bit 14, 13, 12, 11, 10, 9, 8
- campo MANT, formato dai rimanenti bit.

I 7 bit del campo EXP vengono interpretati, nel modo solito, come numero intero in complemento a 2; chiamiamo E questo valore. Gli 8 bit del campo MANT vengono interpretati, nel modo solito, come numero intero senza segno; chiamiamo M questo valore. Allora, il numero razionale rappresentato, chiamiamolo X, é dato da:

$$X = (-1)^S \cdot M \cdot 2^E$$

(a) (punti: -1, 1)
Questa rappresentazione dei razionali segue il principio "virgola fissa", il principio "virgola mobile", oppure nessuno dei due?

(b) (punti: 2)
Calcolare il valore X nel caso S=1, EXP=1111110, MANT=00001010 (indicare il valore usando la rappresentazione decimale solita, ovvero la forma di frazione ridotta, cioe' non semplificabile).
$X =$ ....

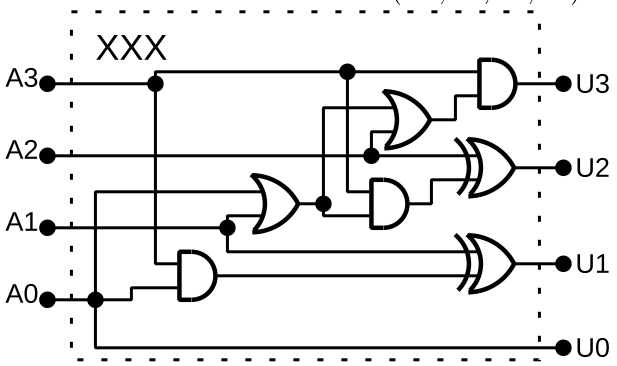
(c) (punti: 2)
Scrivere la configurazione che produce il valore massimo positivo.

bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0

(d) (punti: 2)
Scrivere la configurazione che produce il valore massimo negativo, cioé il valore negativo più vicino a zero, ma diverso da "-0".

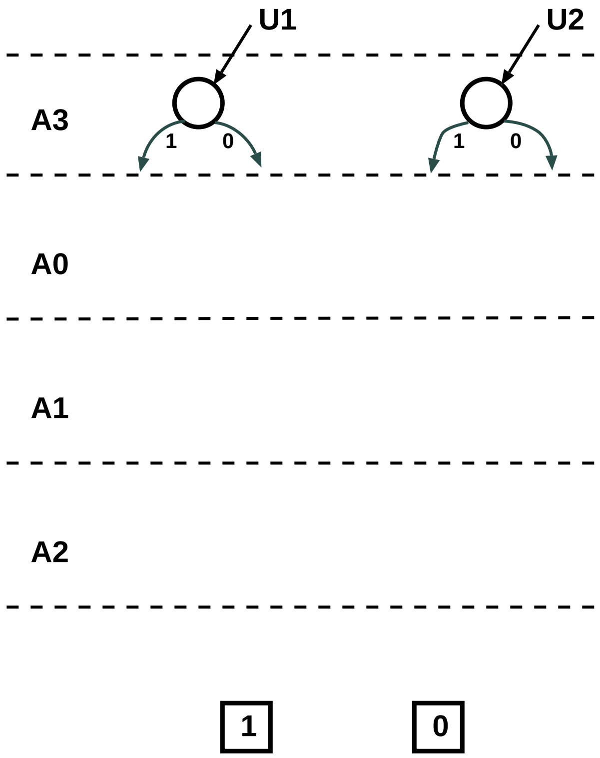
bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0

6. Considerare il modulo circuitale XXX rappresentato in figura, caratterizzato da 4 variabili booleane di ingresso (A3,A2,A1,A0) e 4 variabili booleane di uscita (U3,U2,U1,U0):



(a) (punti: -1, 1)
Il circuito é combinatorio, sequenziale asincrono o sequenziale sincrono?

(b) (punti: 4)
disegnare la rappresentazione BDD canonica delle funzioni U1 ed U2, assumendo l'ordinamento per le variabili di ingresso (dall'alto al basso): A3, A0, A1, A2



(c) (punti: 2)
completare la seguente rappresentazione mediante mappe di Karnaugh delle funzioni U1 e U2:

U1		A3	0	0	1	1
A1	A0	A2	0	1	1	0
0	0					
0	1					
1	1					
1	0					

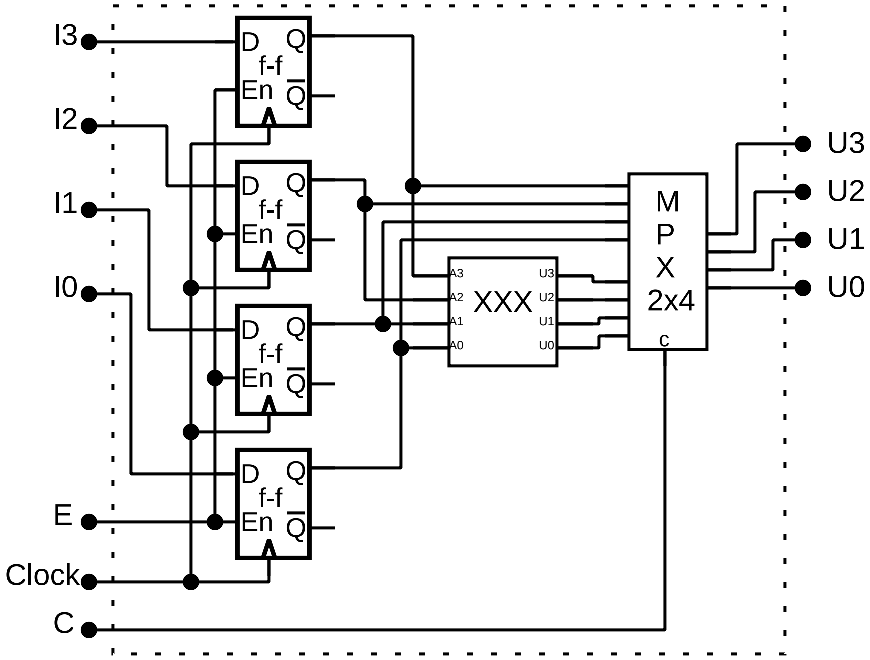
U2		A3	0	0	1	1
A1	A0	A2	0	1	1	0
0	0					
0	1					
1	1					
1	0					

(d) (punti: 4)
Scrivere la rappresentazione in forma normale disgiuntiva (somma di prodotti) minimale per le due funzioni di cui sopra:
U1 =
U2 =

(e) (punti: 3)
Definito N come il valore dell'ultima cifra decimale che compone il nome del login di laboratorio, supponete di mandare in ingresso al modulo circuitale la codifica binaria su quattro bit (con A0 bit meno significativo, A3 bit piu' significativo)) dei seguenti valori, in sequenza, e calcolate i corrispondenti valori sulle quattro uscite booleane:

valore ingr.	A3 A2 A1 A0	U3 U2 U1 U0
N		
$N+1$		
$N+2$		
$(N+8)$ mod 16		
$(N+9)$ mod 16		
$(N+10)$ mod 16		

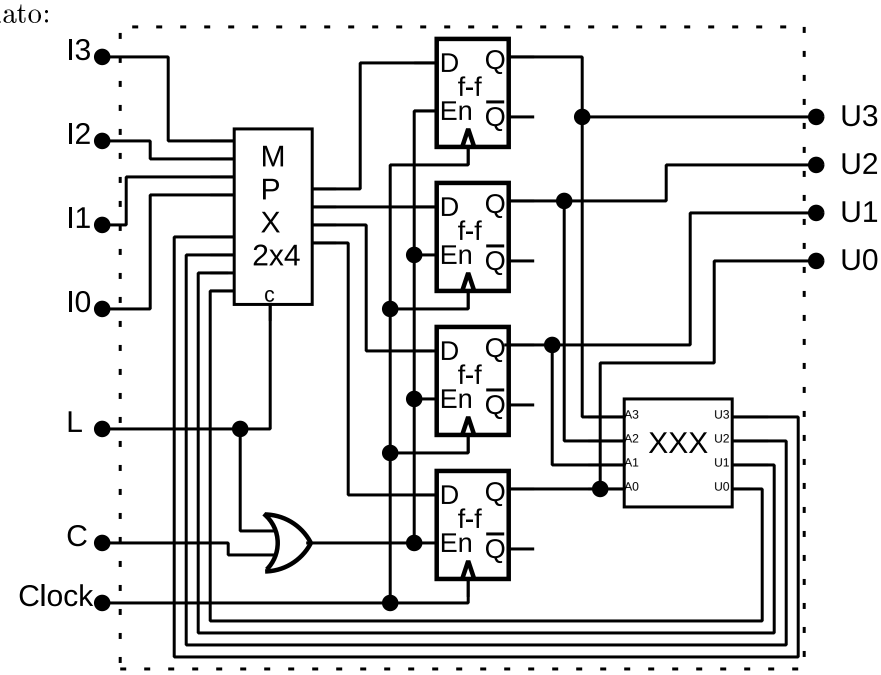
7. Considerare il seguente circuito, contenente tra l'altro una realizzazione del modulo XXX descritto al punto precedente (i flip-flop tipo D sono con segnale di abilitazione: memorizzano solo con En=1):



(a) (punti: -1, 1)
Il circuito é combinatorio, sequenziale asincrono o sequenziale sincrono?

(b) (punti: 4)
Spiegare quale funzione espleta il circuito (e con quali codifiche numeriche opera)

8. Considerare il seguente circuito, contenente tra l'altro una realizzazione del modulo XXX gia precedentemente studiato:



(a) (punti: -1, 1)
Il circuito é combinatorio, sequenziale asincrono o sequenziale sincrono?

(b) (punti: 6)
Calcolare i valori prodotti in uscita a fronte della seguente sequenza di valori di ingresso (una configurazione diversa di ingresso presa successivamente dalla sequenza per ogni variazione significativa dell'ingresso Clock):

L	C	I (val bin)	I3 I2 I1 I0	U3 U2 U1 U0
1	1	N		
0	1	$N+1$		
0	1	$N+2$		
1	0	$(N+8)$ mod 16		
0	0	$(N+9)$ mod 16		
0	1	$(N+10)$ mod 16		

Nota: con N si intende lo stesso valore calcolato in precedenza durante lo studio del module XXX, e si vuole mandare la rappresentazione binaria del valore sui 4 ingressi I. Inoltre si intende che la configurazione di ingresso sia quella applicata prima della variazione del clock, mentre l'uscita da determinare é quella prodotta dopo la variazione significativa del clock.

9. (punti: 6)
Considerare il seguente programma in codice macchina VM-2:

```
 0: LOC8 56        (56 e' decimale)
 1: LOCH 10000000  (questo e' binario....)
 2: SWAS
 3: LOC8 58        (58 e' decimale)
 4: LOCH 10000000  (questo e' binario....)
 5: PUSH
 6: SWAF
 7: POP
 8: SWAF
 9: INSP 2
10: PUSH
11: SIGN
12: ADDL 0
13: PUSH
14: JZER 18
15: SWAF
16: SWAS
17: POP
18: POP
19: SWAS
20: SWAS
21: HALT
```

Questo pezzo di programma NON segue gli schemi visti a lezione per programmare in modo "sensato" (?) la VM2. Però fa qualcosa ... Si richiede di indicare cosa contengono i registri e lo stack alla fine dell'esecuzione del programma. In particolare, quali sono i valori contenuti:
nel registro ACC (in decimale): ....
nel registro FP (in decimale): ....
nel registro SP (in decimale): ....
nelle celle di RAM il cui indirizzo é compreso tra il valore di FP e quello di SP (estremi inclusi): ...., ..., ..., ..., ..., ...

10. (punti: 12)
Il seguente programma VM-2 calcola il massimo di una successione di numeri.

- Input: successione di numeri positivi: $x_1, x_2, \ldots, x_k$. Supponiamo l'input corretto, con almeno un numero (ovvero $k \geq 1$).
- Output: max $\{ x_1,\ x_2,\ \ldots,\ x_k \}$

Il programma usa due procedure (corrette) una per leggere ed una per scrivere i numeri.
Procedura per leggere: il codice inizia all'indirizzo 50 (decimale); legge dall'input un numero; calcola il valore e lo mette nello stack in posizione M[FP+1] (usando il suo FP !!) e lascia zero nell'accumulatore; quando trova la fine dell'input, cioé della successione di numeri, mette un valore negativo nell'accumulatore.
Procedura per scrivere: il codice inizia all'indirizzo 100 (decimale); esegue la stampa del numero che si trova nell'accumulatore.

Idea per il programma:

- riserva, sullo stack, una cella per max ed una per il numero corrente
- legge il primo numero e lo mette in max (é il massimo corrente)
- legge i numeri successivi (uno ad uno) e li confronta con max; se necessario, cambia il valore di max
- alla fine stampa.

Come al solito, gli indirizzi (es 253, 15, ...) sono in decimale.

```
0:  LOC8 253
1:  LOCH 15       oppure LOCH 3840  come su dispense
2:  SWAS
3:  LOC8 255
4:  LOCH 15       oppure LOCH 3840  come su dispense
5:  SWAF
6:  DESP 2
7:  CALL 50
8:  LODL -3
9:  STOL -2
10: CALL 50
11:
12: LODL -3
13:
14: ADDL -2
15: JPOS 10
16:
17:
18: JUMP 10
19: LODL -2
20: CALL 100
21: HALT
```

Scrivere le 4 istruzioni mancanti, agli indirizzi: 11, 13, 16, 17.

027145310013 030276124540 014426113420 004133317526 021734510013 002153407254 010426114342

Risposte primo scritto parziale di Architettura degli Elaboratori, 18/12/1998

Cognome : **Nome**: **Matricola**:

1. (punti: -1, 4) **a** **b** **c** **d** **e** **X f**
2. (punti: -1, 4) **a** **b** **X c** **d** **X e** **f**
3. (punti: -1, 4) **a** **b** **c** **X d** **e** **f**
4.
   (a) (punti: 3) bit4 = 0, bit5 = 0, bit6 = 1

   (b) (punti: 3) Risposta: 0001101
5.
   (a) (punti: -1, 1) Virgola fissa X Virgola mobile Nessuno dei due

   (b) (punti: 2) $X = -5/2$

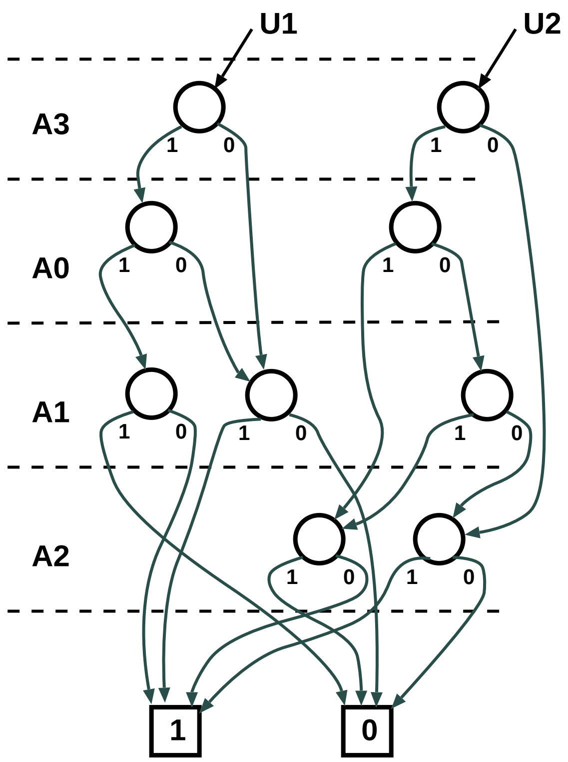
   (c) (punti: 2)

bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1

   (d) (punti: 2)

bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1

6.
   (a) (punti: -1, 1) Combinatorio



   (b) (punti: 4)

(c) (punti: 2)

U1		A3	0	0	1	1
A1	A0	A2	0	1	1	0
0	0		0	0	0	0
0	1		0	0	1	1
1	1		1	1	0	0
1	0		1	1	1	1

U2		A3	0	0	1	1
A1	A0	A2	0	1	1	0
0	0		0	1	1	0
0	1		0	1	0	1
1	1		0	1	0	1
1	0		0	1	0	1

(d) (punti: 4)
U1 = ¬A0·A1 + A1·¬A3 + A0·¬A1·A3
U2 = A2·¬A3 + ¬A0·¬A1·A2 + A0·¬A2·A3 + A1·¬A2·A3

(e) (punti: 3) Converte modulo e segno (A) in complemento a 2 (U). Esempio con $N = 1$:

valore ingr.	A3	A2	A1	A0	U3	U2	U1	U0
N	0	0	0	1	0	0	0	1
$N+1$	0	0	1	0	0	0	1	0
$N+2$	0	0	1	1	0	0	1	1
$(N+8)$ mod 16	1	0	0	1	1	1	1	1
$(N+9)$ mod 16	1	0	1	0	1	1	1	0
$(N+10)$ mod 16	1	0	1	1	1	1	0	1

7.

(a) (punti: -1, 1) Sequenziale Asincrono

(b) (punti: 4)
Memorizza numero in modulo e segno quando E=1, produce in uscita lo stesso numero in modulo e segno con C=1 e in complemento a 2 con C=0.

8.

(a) (punti: -1, 1)
Sequenziale Sincrono

(b) (punti: 6)

L	C	I (val bin)	I3 I2 I1 I0	U3 U2 U1 U0
1	1	N	come 6.e col.A N	come 6.e col.A N
0	1	$N+1$	come 6.e col.A $N+1$	come 6.e col.U N
0	1	$N+2$	come 6.e col.A $N+2$	come 6.e col.A N
1	0	$(N+8)$ mod 16	come 6.e col.A $N+8$	come 6.e col.A $N+8$
0	0	$(N+9)$ mod 16	come 6.e col.A $N+9$	come 6.e col.A $N+8$
0	1	$(N+10)$ mod 16	come 6.e col.A $N+10$	come 6.e col.U $N+8$

9. (punti: 6)
ACC: 0, FP: 58, SP: 57, M[58]: 58, M[57]: 0

10. (punti: 12)

```
11: JNEG 19

13: SIGN

16:  LODL -3
17:  STOL -2
```