

PROVA SCRITTA DEL CORSO DI
CALCOLATORI ELETTRONICI
NUOVO ORDINAMENTO DIDATTICO (5 CFU)

NOME:

COGNOME:

MATRICOLA:

ESERCIZIO 1 (9 punti)

Progettare una rete sequenziale che presenti un ingresso X e un'uscita Z posta a 1 ogni volta che viene riconosciuta la sequenza 10111. Si richiede:

1. (4 punti) il diagramma degli stati, la tabella di flusso e la tabella delle transizioni;
2. (3 punti) il calcolo delle forme minime delle variabili di eccitazione dei flip flop con le mappe di Karnaugh. Si usino flip flop JK. Calcolare anche la rete combinatoria per l'uscita Z.
3. (2 punti) Realizzare un flip flop T a partire da un flip flop JK.

ESERCIZIO 2 (6 punti)

Si supponga di disporre di tre macchine: a pila, a uno e a due indirizzi. Per ognuna di queste si abbiano le seguenti istruzioni:

A pila		A un indirizzo		A due indirizzi	
Istr.	Semantica	Istr.	Semantica	Istruzione	Semantica
PUSH X	$M[X] \rightarrow \text{push}$	STORE X	$\text{ACC} \rightarrow M[X]$	MOV X1, X2	$M[X1] \rightarrow M[X2]$
POP X	$\text{pop} \rightarrow M[X]$	LOAD X	$M[X] \rightarrow \text{ACC}$	ADD X1, X2	$M[X1] + M[X2] \rightarrow M[X2]$
ADD	$\text{pop} + \text{pop} \rightarrow \text{push}$	ADD X	$\text{ACC} + M[X] \rightarrow \text{ACC}$	DIV X1, X2	$M[X1] / M[X2] \rightarrow M[X2]$
DIV	$\text{pop} / \text{pop} \rightarrow \text{push}$	DIV X	$\text{ACC} / M[X] \rightarrow \text{ACC}$		

ACC è il registro accumulatore della macchina a un indirizzo. $M[X]$ indica il dato nella locazione di memoria X. Si scriva, per ognuna delle tre macchine, la sequenza delle istruzioni necessarie per realizzare la seguente operazione:

$$Z = (A + B) / C.$$

Le lettere indicano le locazioni di memoria dove si trovano i dati. Nella macchina a due indirizzi si faccia uso di una ulteriore locazione P dove introdurre i risultati parziali, pena la perdita dei dati iniziali nelle locazioni A, B o C.

ESERCIZIO 3 (8 punti)

Si consideri il seguente formato per la rappresentazione binaria dei numeri in virgola mobile: 24 bit, con esponente a 8 bit in eccesso **125**, mantissa frazionaria e normalizzata in segno e valore (1.M).

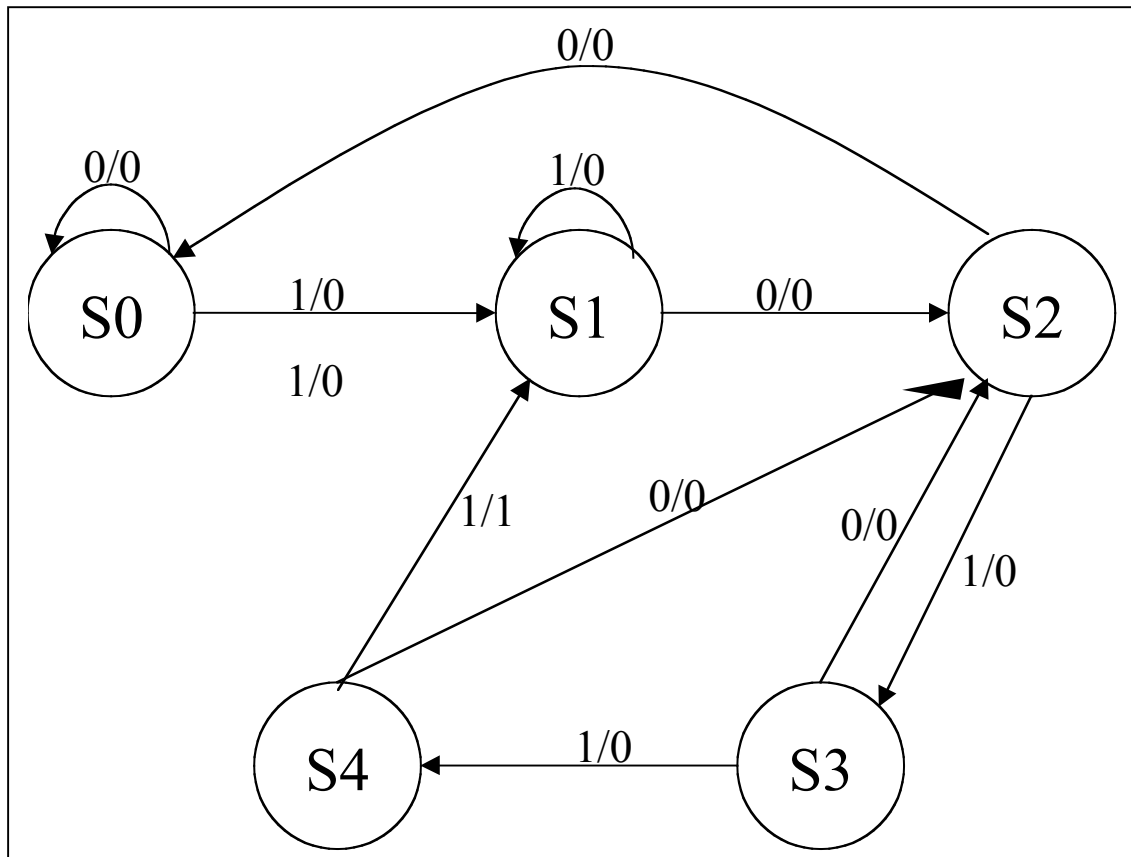
1. (3 punti) Si calcoli il minimo e il massimo numero **positivo** rappresentabili, escluso lo zero.
2. (3 punti) Si rappresentino nel formato dato i numeri 130.25 e 120.75.
3. (2 punti) Si sommino i due numeri seguendo i passi usati nell'algoritmo dei calcolatori.

ESERCIZIO 4 (10 punti)

1. (6 punti) Si progetti una funzione assembly MIPS che, ricevendo in ingresso un vettore v, e due interi i e j, permuti $v[i]$ con $v[j]$. Il passaggio dei parametri presenta le specifiche: $\&v[0] \rightarrow \$4$, $i \rightarrow \$5$, $j \rightarrow \$6$.
2. (4 punti) Si spieghi brevemente in cosa consiste il concetto di "JumpTable" usato in assembly MIPS, ad esempio nell'implementazione del codice corrispondente al costrutto C "switch/case".

ESERCIZIO 1**Soluzione.**

Il diagramma degli stati è il seguente:



La tabella di flusso è data da:

Stato presente	Stato successivo/Uscita	
	X=0	X=1
S0	S0/0	S1/0
S1	S2/0	S1/0
S2	S0/0	S3/0
S3	S2/0	S4/0
S4	S2/0	S1/1

Per codificare 5 stati occorrono tre flip flop. La codifica è la seguente:

S0 → 0 0 0; ...; S4 → 1 0 0. Nel seguito indicheremo ciascun bit della codifica con le lettere A, B, C. L'apice indicherà il bit nell'istante successivo a quello considerato.

A partire dalla tabella di eccitazione del flip flop JK:

Q	Q'	J	K
0	0	0	D
0	1	1	D
1	0	D	1
1	1	D	0

A	B	C	X	A'	Ja	Ka	B'	Jb	Kb	C'	Jc	Kc	Z
0	0	0	0	0	0	D	0	0	D	0	0	D	0
0	0	0	1	0	0	D	0	0	D	1	1	D	0
0	0	1	0	0	0	D	1	1	D	0	D	1	0
0	0	1	1	0	0	D	0	0	D	1	D	0	0
0	1	0	0	0	0	D	0	D	1	0	0	D	0
0	1	0	1	0	0	D	1	D	0	1	1	D	0
0	1	1	0	0	0	D	1	D	0	0	D	1	0
0	1	1	1	1	1	D	0	D	1	0	D	1	0
1	0	0	0	0	D	1	1	1	D	0	0	D	0
1	0	0	1	0	D	1	0	0	D	1	1	D	1
1	0	1	0	D	D	D	D	D	D	D	D	D	D
1	0	1	1	D	D	D	D	D	D	D	D	D	D
1	1	0	0	D	D	D	D	D	D	D	D	D	D
1	1	0	1	D	D	D	D	D	D	D	D	D	D
1	1	1	0	D	D	D	D	D	D	D	D	D	D
1	1	1	1	D	D	D	D	D	D	D	D	D	D

Ora possiamo disegnare le mappe di Karnaugh

AB		CX			
		00	01	11	10
CX	00			d	d
	01			d	d
	11		1	d	d
	10			d	d
$J_A = BCX$					

AB		CX			
		00	01	11	10
CX	00	d	d	d	1
	01	d	d	d	1
	11	d	d	d	d
	10	d	d	d	d
$K_A = 1$					

		AB			
		00	01	11	10
CX	00		d	d	1
	01		d	d	
	11		d	d	d
	10	1	d	d	d

$$J_B = C\bar{X} + A\bar{X}$$

		AB			
		00	01	11	10
CX	00	d	1	d	d
	01	d		d	d
	11	d	1	d	d
	10	d		d	d

$$K_B = \bar{C} \cdot \bar{X} + CX$$

		AB				
		CX	00	01	11	10
CX	00			d		
	01	1	1	d	1	
	11	d	d	d	d	
	10	d	d	d	d	

$$J_c = X$$

		AB				
		CX	00	01	11	10
CX	00	d	d	d	d	
	01	d	d	d	d	
	11		1	d	d	
	10	1	1	d	d	

$$K_C = B + \bar{X}$$

Infine, per quanto riguarda l'uscita Z:

$$Z = A \cdot \bar{B} \cdot \bar{C} \cdot X$$

Volendo utilizzare anche i don't care:

		AB			
		00	01	11	10
CX	00			d	
	01			d	1
	11			d	d
	10			d	d

$$Z = AX$$

Per realizzare un flip flop T da un JK, è sufficiente connettere fra loro gli ingressi J e K, come si può vedere confrontando la tabella delle transizioni del JK con quella del T:

	J	K	Q	Q'	T	Q	Q'
{	0	0	0	0	0	0	0
	0	0	1	1	0	1	1
	0	1	0	0	1	0	1
	0	1	1	0	1	1	0
	1	0	0	1			
	1	0	1	1			
{	1	1	0	1			
	1	1	1	0			

ESERCIZIO 2**Soluzione.**

A pila	A un indirizzo	A due indirizzi	
PUSH C	LOAD A	ADD A,B	MOV B,P
PUSH A	ADD B	DIV B,C	ADD A,P
PUSH B	DIV C	MOV C,Z	MOV C,Z
ADD	STORE Z		DIV P,Z
DIV			
POP Z			
I contenuti delle locazioni A, B e C rimangono intatti.	I contenuti delle locazioni A, B e C rimangono intatti.	Vengono perduti i contenuti delle locazioni B e C.	I contenuti delle locazioni A, B e C rimangono intatti.

ESERCIZIO 3

Soluzione.

1. Minimo numero: 2^{-125} .
Massimo numero: $(2-2^{-15}) * 2^{130}$.
2. $130.25 = 10000010.01 = 1.000001001 * 2^{-7}$.
 $120.75 = 1111000.11 = 1.11100011 * 2^{-6}$.

	S	Esponente	Mantissa
130.25	0	10000100	000001001000000
120.75	0	10000011	111000110000000

3. Allineando le mantisse e sommando si ottiene:

$$\begin{array}{r} 130.25 = 1.000001001 * 2^{-7} + \\ 120.75 = 0.111100011 * 2^{-7} = \\ \hline 1.111101100 * 2^{-7} = 251 \end{array}$$

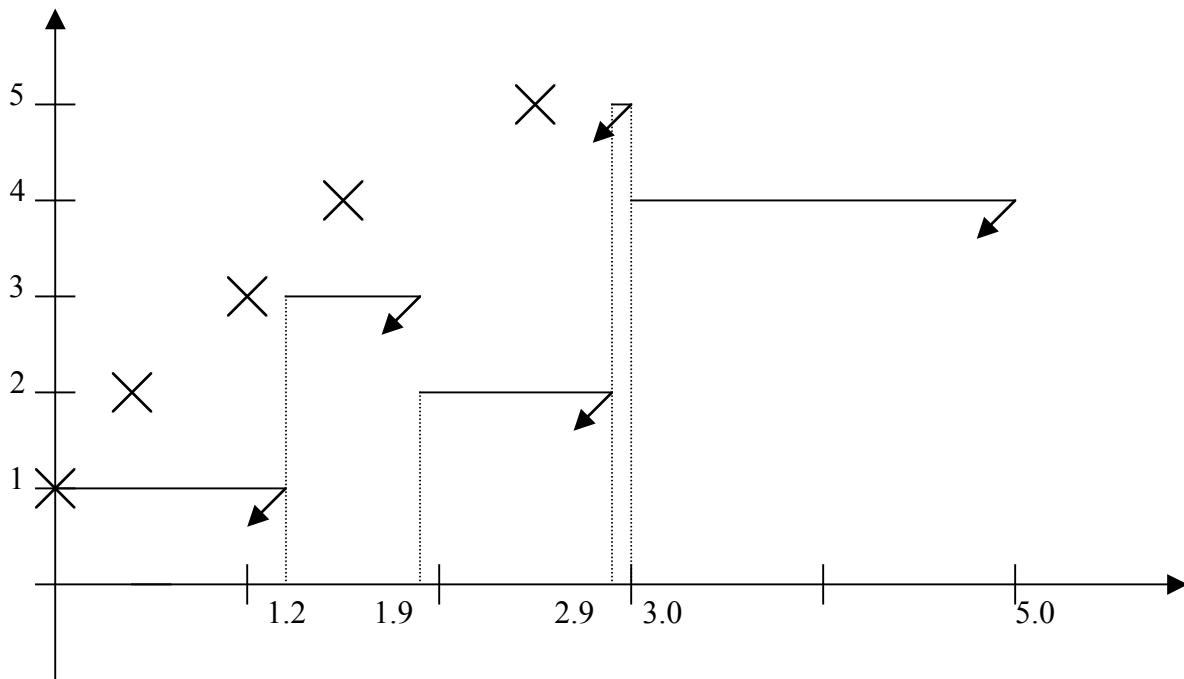
La cui rappresentazione è:

	S	Esponente	Mantissa
251	0	10000100	111101100000000

ESERCIZIO 4

Soluzione.

Grafico processo-tempo con politica SJF monoprogrammata:



Processo	Arrivo	Inizio	Fine	Turnaround	WT
1	0.00	0.00	1.20	1.20	1.00
2	0.40	1.90	2.90	2.50	2.50
3	1.00	1.20	1.90	0.90	1.29
4	1.50	3.00	5.00	3.50	1.75
5	2.50	2.90	3.00	0.50	5.00
Media				1.72	2.31

ESERCIZIO 5

Soluzione.

1.

\$9 → v[i], \$10 → v[j]

swap:

```
    addi $29, $29, -20    #salvataggio del contesto
    sw $4, 0($29)
    sw $5, 4($29)
    sw $6, 8($29)
    sw $9, 12($29)
    sw $10, 15($29)

    muli $5, $5, 4        #calcolo indirizzo di v[i] e v[j]
    muli $6, $6, 4
    add $5, $5, $4        #i x 4 + &v[0]
    add $6, $6, $4        #j x 4 + &v[0]
    lw $9, 0($5)          #prelevo v[i] e v[j]
    lw $10, 0($6)
    sw $9, 0($6)          #permuto v[i] con v[j]
    sw $10, 0($5)

    lw $4, 0($29)         #ripristino del contesto
    lw $5, 4($29)
    lw $6, 8($29)
    lw $9, 12($29)
    lw $10, 15($29)
    addi $29, $29, 20

    jr $31                #ritorno al chiamante
```

2. si veda la slide 25, cap. 5