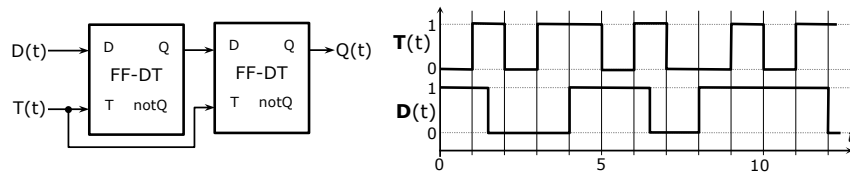




1. [2] Si rappresenti il numero $N = -\frac{1}{2048} - \frac{1}{32}$ secondo lo standard IEEE-754, singola precisione, in formato esadecimale.

2. [3] Dare la definizione e scrivere l'espressione generale della prima forma canonica di una funzione logica. Esprimere: $f(a,b,c) = (a + \bar{b})(b + c)$ in prima forma canonica.

3. [4] Si calcoli il cammino critico del circuito in figura e si tracci l'andamento dell'uscita Q in corrispondenza degli ingressi in figura. Si considerino inizialmente le uscite di entrambi i flip-flop a '0'.



4. [5] Si progetti un circuito caratterizzato da 3 ingressi (a_2, a_1, a_0) rappresentanti un numero intero A con segno (in complemento a 2) e due uscite Y e Z. L'uscita Y vale '1' se e solo se $|A| < 2$, mentre Z vale '1' se e solo se $|A| > 1$.

a) Determinare le tabelle di verità di Y e Z; b) esprimerle nella forma canonica più adatta; c) semplificarle mediante mappe di Karnaugh; d) semplificarle ulteriormente, se possibile, mediante semplificazioni algebriche; e) disegnare lo schema circuitale.

5. [8] Si sintetizzi una macchina a stati finiti di Moore sincrona, caratterizzata da due linee d'ingresso A e B, osservate ogni secondo, ed una linea di uscita Y. In corrispondenza di un fronte di salita su A, Y si porta a '1' e vi rimane per 1 secondo se, in tale istante, B=0, oppure per 3 secondi se B=1, quindi torna a '0'. Mentre Y=1, la macchina non reagisce ad altri fronti di salita su A. Si considerino inizialmente A, B e Y a '0'. Si determinino: STG, STT, STT codificata e struttura circuitale del sistema completo, gestendo il segnale di clock ed avendo cura di semplificare il più possibile le funzioni prima di tradurle in circuito.

6. [7] Si scriva un programma Assembly, per ambiente SPIM, che richieda all'utente da tastiera 25 numeri interi, memorizzandoli in un array (allocato staticamente), quindi chiami una funzione a cui si passa l'indirizzo base dell'array e che restituisce il valore massimo dell'array, e infine stampi tale valore. Il programma deve apparire a video come nell'esempio a lato.

```
Elemento 0 > 45
Elemento 1 > 12
...
Elemento 24 > 9
Il massimo è: 45
```

7. [5] Si traduca il seguente frammento di codice Assembly MIPS in linguaggio macchina, in formato esadecimale, calcolando prima i valori esadecimali **N1** e **N2** che permettono di saltare esattamente all'indirizzo indicato in ciascun commento.

```
0x80808080:    j N1          # salta a: 0x81818180
               beq $s0, $s4, N2    # salta a: 0x808080A0
```

System calls

	codice (\$v0)	argomenti	risultato
print int	1	\$a0	
print float	2	\$f12	
print double	3	\$f12	
print string	4	\$a0	
read int	5		\$v0
read float	6		\$f0
read double	7		\$f0
read string	8	\$a0, \$a1	
sbrk	9	\$a0	\$v0
exit	10		

Registri MIPS

		24-25	t8 - t9
0	zero		
1	at	26-27	k0 - k1
2-3	v0 - v1	28	Gp
4-7	a0 - a3	29	Sp
8-15	t0 - t7	30	s8
16-23	s0 - s7	31	Ra

MIPS Instruction Set:

