

Sono qui di seguito proposti alcuni esercizi sugli argomenti trattati nella prima parte delle lezioni di Consapevolezza digitale. Questo materiale è da intendersi come ulteriore supporto didattico per gli studenti e le studentesse che hanno seguito le lezioni e non ha alcuna pretese di essere un eserciziario ufficiale. Si prega quindi di segnalare al docente (andrea.gerosa@unipd.it) eventuali errori.

Tutti i diritti sono riservati e qualsiasi uso diverso dallo studio per la preparazione personale è vietato.

1.1. Convertire in base binaria il numero in base esadecimale 1234_{16} .

Per effettuare conversioni tra sistemi di numerazione in cui la base è sempre una potenza di 2, si può procedere semplicemente con il metodo della *sostituzione*. Nel caso di questo esercizio, per convertire il numero in base 16 in una rappresentazione in base 2, basta procedere ad esprimere in base 2 ciascuna cifra del numero in base 16 (ricordando che per ogni cifra del sistema in base 8 occorrono 4 bit in base 2), ottenendo

$$1\ 2\ 3\ 4 = 0001\ 0010\ 0011\ 0100$$

Possiamo quindi colludere che:

$$1234_{16} = 0001001000110100_2$$

1.2. Convertire il numero in base ottale 1234_8 in base decimale.

Per convertire un numero in una qualsiasi base in base decimale, è sufficiente espandere il numero in forma di combinazione lineare delle potenze della base e calcolare il risultato con le regole dell'algebra decimale. Nel caso particolare di questo esercizio si ottiene:

$$1234_8 = 1 \cdot 8^3 + 2 \cdot 8^2 + 3 \cdot 8^1 + 4 \cdot 8^0 = 512 + 128 + 24 + 4 = 668_{10}$$

1.3. Convertire il numero in base decimale 108_{10} , in base binaria.

Per convertire un numero decimale in una generica base r , si procede con il metodo di *divisione*: si divide cioè il numero in base 10 per r e poi si continua ripetutamente a dividere per r il risultato di ogni divisione. I resti che si ottengono ad ogni divisione rappresentano le cifre in base r , da quella meno significativa a quella più significativa. Nel caso specifico dell'esercizio si ottiene:

$$108/2 = 54 \text{ con resto } 0$$

Il bit meno significativo (LSB) sarà quindi "0". Procedendo con l'algoritmo:

$$54/2 = 27 \text{ con resto } 0$$

$$27/2 = 13 \text{ con resto } 1$$

$$13/2 = 6 \text{ con resto } 1$$

$$6/2 = 3 \text{ con resto } 0$$

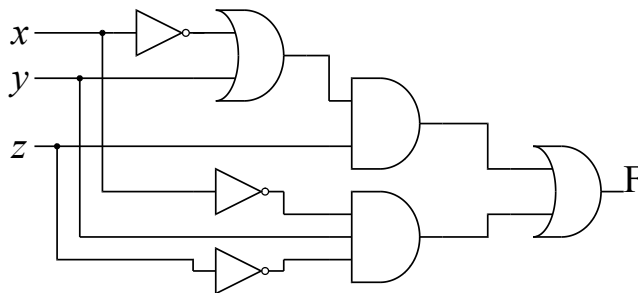
$$3/2 = 1 \text{ con resto } 1$$

$$1/2 = 0 \text{ con resto } 1$$

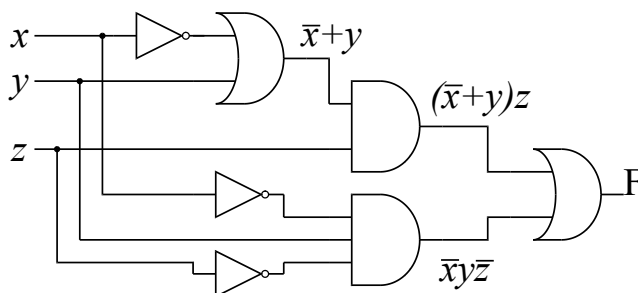
Il risultato cercato è quindi

$$108_{10} = 1101100_2$$

1.4. Valutare la funzione logica $F(x,y,z)$ valutata dal circuito logico riportato in figura.



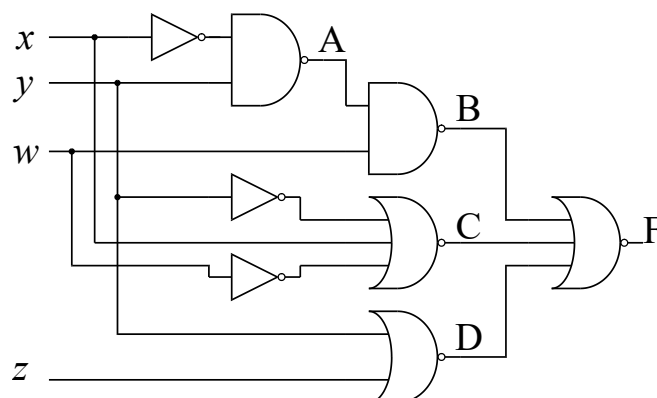
Le espressioni logiche valutate in alcuni nodi interni del circuito sono evidenziate nella figura seguente:



Sommando quindi i due termini in ingresso all'ultima porta OR, si ottiene la funzione cercata:

$$F(x, y, z) = (\bar{x} + y) \cdot z + \bar{x}y\bar{z} = \bar{x}z + yz + \bar{x}y\bar{z} = \bar{x}z + yz + \bar{x}y$$

1.5. Valutare la funzione logica $F(x,y,w,z)$ valutata dal circuito logico riportato in figura.



Per chiarezza si è assegnato un nome ai nodi interni della figura: si può quindi valutare l'espressione logica valutata in ciascuno dei nodi.

$$\begin{aligned} A &= \overline{\overline{x} \cdot y} \\ B &= \overline{A \cdot w} = \overline{\overline{\overline{x} \cdot y} \cdot w} \\ C &= \overline{x + y + w} \\ D &= \overline{y + z} \end{aligned}$$

Quindi in base al circuito disegnato si può ottenere l'espressione della funzione totale:

$$F(x, y, w, z) = \overline{B + C + D} = \overline{\overline{\overline{\overline{x} \cdot y} \cdot w} + \overline{x + y + w} + \overline{y + z}}$$

Tale espressione può essere ovviamente semplificata utilizzando i teoremi di De Morgan:

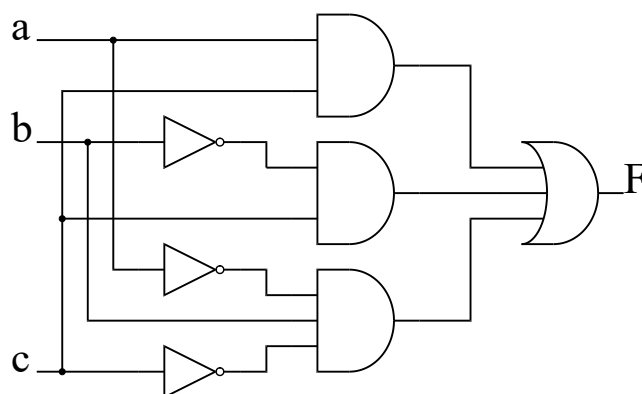
$$F(x, y, w, z) = (\overline{\overline{x} \cdot y} \cdot w) \cdot (x + \overline{y} + \overline{w}) \cdot (y + z) = w \cdot (x + \overline{y}) \cdot (x + \overline{y} + \overline{w}) \cdot (y + z)$$

1.6. Siano dati due numeri da 2 bit ciascuno $P=P_1P_0$ e $Q=Q_1Q_0$. Si deve realizzare un funzione logica $F(P_1P_0Q_1Q_0)$ che vale 1 se e solo se $P > Q$. Si realizzi la funzione F .

Soluzione:

$$F = \overline{P_1} \cdot P_0 \cdot \overline{Q_1} \cdot \overline{Q_0} + P_1 \cdot \overline{P_0} \cdot \overline{Q_1} \cdot \overline{Q_0} + P_1 \cdot \overline{P_0} \cdot \overline{Q_1} \cdot Q_0 + \\ + P_1 \cdot P_0 \cdot \overline{Q_1} \cdot \overline{Q_0} + P_1 \cdot P_0 \cdot \overline{Q_1} \cdot Q_0 + P_1 \cdot P_0 \cdot Q_1 \cdot \overline{Q_0}$$

1.7. Si identifichino i mintermini della funzione logica realizzata dal circuito logico di figura.



SOLUZIONE: $F(a,b,c)$: On-set={m1,m2,m5,m7}

1.8. Si identifichino i mintermini della funzione logica $F(a,b,c,d)$, sapendo che F vale "1" se e solo se la

combinazione delle quattro variabili di ingresso è la rappresentazione in base 2 di un numero primo.

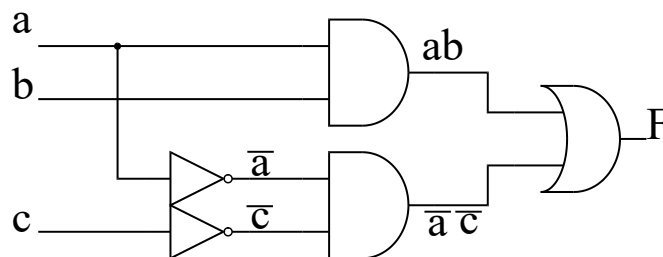
SOLUZIONE:

$F(a,b,c,d)$: On-set = {m1, m2, m3, m5, m7, m11, m13}.

1.9. Disegnare il circuito logico che implementa la funzione

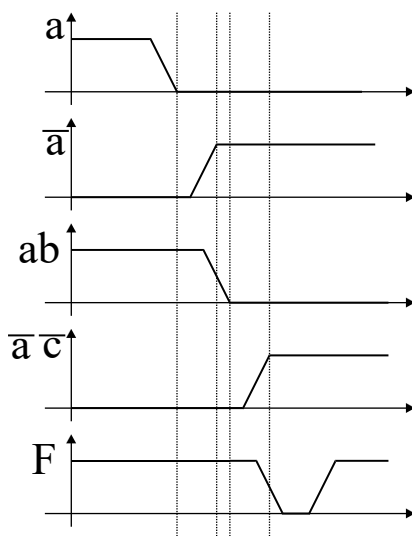
$F(a,b,c) = ab + \overline{a}c$ ed individuare possibili funzionamenti anomali nel caso di una commutazione 1→0 dell'ingresso a . [Suggerimento: il funzionamento anomalo si riscontra esclusivamente tenendo conto dei tempi di propagazione delle porte logiche diversi da 0].

Il circuito logico si ricava facilmente dall'equazione:



Un possibile funzionamento anomalo è legato al ritardo di propagazione dell'invertitore collegato all'ingresso a : a causa di questo esisterà un intervallo temporale in cui a (ingresso della prima porta AND) avrà lo stesso valore del suo negato (ingresso della seconda porta AND). Tale situazione è esemplificata nel seguente

diagramma temporale, disegnato ipotizzando che l'ingresso b rimanga costante a "1" e c rimanga costante a "0".



Si noti che $F(1,1,0)=1$ e che $F(0,1,0)=1$, pertanto con l'andamento degli ingressi ipotizzato per disegnare il diagramma temporale, il valore di uscita della funzione logica dovrebbe rimanere costante a 1, mentre si rileva un (breve) intervallo di tempo in cui la funzione si porta a 0.

Per evitare tale problema è possibile aggiungere un porta AND nel primo livello del circuito, la cui uscita rimanga costantemente a "1" durante la commutazione di a. In base al teorema del consenso si può affermare che:

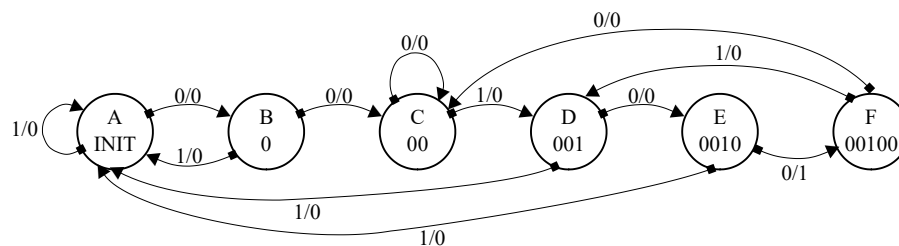
$$F(a,b,c) = ab + \overline{a}c = ab + \overline{a}c + bc$$

L'ultimo termine prodotto in tale formula è palesemente indipendente da a . Nelle condizioni del diagramma temporale considerato in precedenza ($b=1$, $c=0$) si avrà quindi che tale termine rimane costantemente a valore logico "1", indipendentemente dalla commutazione di a . In tal modo si è ottenuto che uno degli ingressi della porta OR del secondo livello sia costante a valore logico "1", quindi anche l'uscita F rimarrà costante a tale valore, nonostante la commutazione di a .

1.10. Si progetti una macchina sequenziale sincrona con un ingresso x ed un'uscita y tale che l'uscita y vale "1" se e solo se in ingresso si è presentata la sequenza "00100".

Si può costruire il digramma degli stati del sistema, partendo da uno stato iniziale A . Se da tale stato si riceve in ingresso $x=1$ la sequenza attesa non comincia, pertanto si permane nello stato A e l'uscita è "0". Al contrario se in ingresso arriva $x=0$, questo può rappresentare l'inizio della sequenza: si aggiorna quindi lo stato del sistema allo stato B . Tale transizione è mostrata nel diagramma di stato seguente, dove all'interno degli stati è indicato anche il significato associato a ciascuno stato. Lo stato A quindi è semplicemente uno stato iniziale, mentre all'interno dello stato B è indicato "0" che rappresenta la porzione della sequenza ricevuta fino a quel punto. Se a partire dallo stato B si riceve in ingresso il resto della sequenza il sistema evolve attraverso gli stati C , D , E ed F .

Nell'ultima transizione da E ad F inoltre l'uscita assume valore "1", dato che si è ricevuta la sequenza corretta.



Il diagramma mostra naturalmente anche l'evoluzione del sistema nel caso di sequenze di ingresso diverse: se dopo il primo "0" (stato B) si riceve un "1" il sistema torna allo stato iniziale, dato che nessun simbolo ricevuto può comporre la sequenza cercata. Osservazioni analoghe si possono trarre nel caso ci si trovi negli stati D ed E. Nel caso dello stato C, se si riceve un ulteriore simbolo "0" in ingresso non si procede nella ricezione della sequenza, ma comunque si sono potenzialmente ricevuti i primi due simboli della sequenza, quindi si permane nello stato C. Infine una volta ricevuta la sequenza (stato F) si procede nello stato C o D a seconda che il nuovo ingresso valga rispettivamente "0" o "1": il sistema riconosce quindi anche sequenze concatenate.

A partire dal diagramma degli stati si può ora ricavare la seguente tabella delle transizioni:

	x=0	x=1
A	B,0	A,0
B	C,0	A,0
C	C,0	D,0
D	E,0	A,0
E	F,1	A,0
F	C,0	D,0

Prima di procedere si può verificare se sussistono stati equivalenti. Come si vede dal diagramma lo stato C ed F sono equivalenti: infatti hanno gli stessi stati futuri e la stessa uscita a parità di ingressi. La tabella delle transizioni è quindi ridotta di una riga:

	x=0	x=1
A	B,0	A,0
B	C,0	A,0
C	C,0	D,0
D	E,0	A,0
E	C,1	A,0

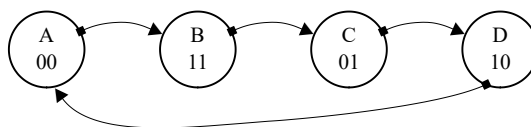
Passando all'assegnazione di una codifica agli stati, si nota prima di tutto che, avendo 5 stati, sono necessarie almeno 3 variabili di

stato binarie. Una possibile codifica è A=000, B=001, C=011, D=010 e E=100. La tabella delle transizioni diviene quindi:

$Q_2Q_1Q_0$	$x=0$	$x=1$
000	001,0	000,0
001	011,0	000,0
011	011,0	010,0
010	100,0	000,0
100	011,1	000,0

1.11. Progettare una macchina sequenziale sincrona che genera in uscita la sequenza 0, 3, 1, 2. Si assuma di rappresentare i numeri di tale sequenza in base 2.

Il generatore di sequenza può essere realizzato utilizzando una macchina sequenziale sincrona, senza nessun segnale di ingresso e con un'uscita a 2 bit che fornisce la sequenza desiderata. Un possibile diagramma degli stati del sistema è riportato di seguito, evidenziando in questo caso il valore dell'uscita all'interno degli stati, vista la mancanza di ingressi.



Dal digramma degli stati si ricava la seguente tabella delle transizioni.

		Z_1, Z_0
A	B	00
B	C	11
C	D	01
D	A	10

Una possibile codifica per gli stati è: A=00, B=01, C=11, D=10. La tabella delle transizioni diviene quindi:

		Z_1, Z_0
00	01	00
01	11	11
11	10	01
10	00	10

Osservando quest'ultima tabella è facile ottenere le funzioni logiche di uscita e di aggiornamento dello stato:

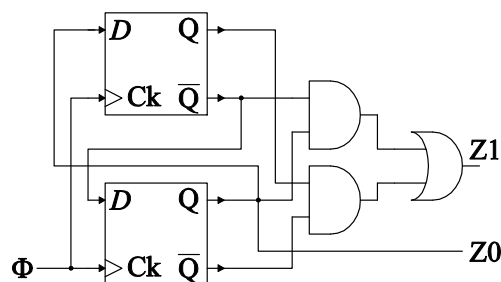
$$Q_1^* = Q_0$$

$$Q_0^* = \overline{Q_1}$$

$$Z_1 = \overline{Q_1} \cdot Q_0 + Q_1 \cdot \overline{Q_0}$$

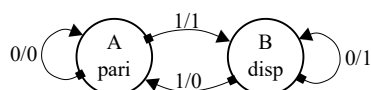
$$Z_0 = Q_0$$

Il circuito corrispondente è riportato di seguito.



1.12. Progettare una macchina sequenziale sincrona che realizza un generatore di parità. Il sistema riceve in ingresso una sequenza di bit e genera per ogni nuovo ingresso la parità della sequenza ricevuta fino a quel momento. La parità vale "1" se la sequenza contiene un numero dispari di simboli "1".

Il sistema richiesto può essere realizzato molto facilmente con un macchina sequenziale sincrona a due soli stati, come mostra il digramma seguente.



A questo digramma è associata un semplice tabella delle transizioni:

	x=0	x=1
A=0	0,0	1,1
B=1	1,1	0,0

da cui si ricavano le equazioni per aggiornare lo stato e per calcolare la parità di uscita:

$$S^* = S \cdot \overline{X} + \overline{S} \cdot X = S \oplus X$$

$$Z = S \cdot \overline{X} + \overline{S} \cdot X = S \oplus X$$

1.13. Si deve progettare una macchina sequenziale sincrona che è parte di un sistema più complesso utilizzato per il riconoscimento di codici numerici (PIN) a 4 cifre (Es. BANCOMAT). La macchina ha due ingressi: EN (Enable) che abilita la macchina a verificare la correttezza del codice e EQ (Equal) che è generato da un comparatore che confronta in modo sequenziale (sincrono con la macchina sequenziale) le 4 cifre immesse dall'utente con le corrispondenti 4 cifre del PIN corretto. La macchina ha due uscite: OK che segnala la correttezza del PIN (EQ=1 per 4 cicli di clock) ed ERR che segnala un qualsiasi malfunzionamento del sistema (es. EN torna a 0 prima di aver eseguito il confronto su 4 cifre). La macchina deve soddisfare le seguenti specifiche:

Con EN=0:

- Le due uscite rimangono a 0 e la macchina è insensibile al valore di EQ

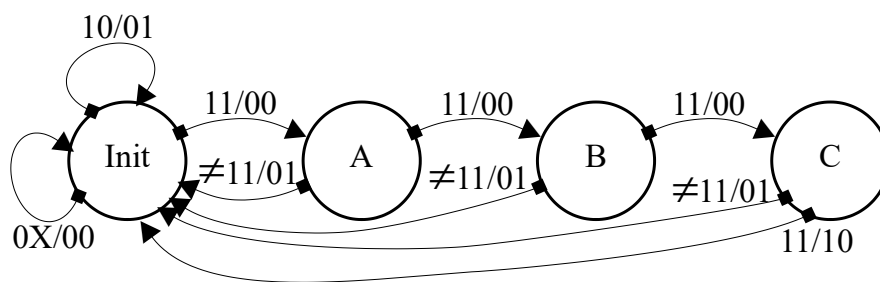
Con EN=1:

- La macchina comincia a monitorare il valore di EQ, tenendo traccia dell'evoluzione di tale ingresso per verificare che EQ rimanga ad 1 per 4 cicli di clock mantenendo le due uscite a 0;
- Al quarto ciclo di clock in cui EQ=1, l'uscita OK viene portata ad "1" e quindi la macchina torna ad operare nelle condizioni del punto precedente;

- Se si verifica $EQ=0$, l'uscita ERR viene portata ad "1" e quindi la macchina torna ad operare nelle condizioni del primo punto;
- Se si verifica $EN=0$ prima di aver terminato la verifica sul valore di EQ per quattro cicli di clock, l'uscita ERR viene portata ad "1" e quindi la macchina torna ad operare nelle condizioni del primo punto;

Progettare la macchina sequenziale.

La macchina sequenziale da realizzare deve a tutti gli effetti riconoscere una sequenza di quattro simboli "1" sull'ingresso EQ, quando l'altro ingresso è abilitato; ogni altro evento in ingresso riporta la macchina nello stato iniziale ed abilita l'uscita ERR. Il diagramma di stato può quindi essere rappresentato come segue, avendo indicato su ciascun arco la sequenza EN EQ/OK ERR. Si può notare chiaramente come si proceda ciclicamente attraverso i 4 stati, riconoscendo la sequenza di 4 simboli "1" su EQ solo con EN EQ = 11; qualsiasi altra combinazione di ingresso riporta nello stato iniziale, attivando l'uscita ERR.



Dal diagramma si ricava la tabella delle transizioni:

	00	01	11	10
Init	Init,00	Init,00	A,00	Init,01
A	Init,01	Init,01	B,00	Init,01
B	Init,01	Init,01	C,00	Init,01
C	Init,01	Init,01	Init,10	Init,01

Si nota che lo stato Init non può essere equivalente a nessun altro stato, poiché a parità di configurazioni di ingresso mostra valori diversi delle uscite. Analoga considerazione si può applicare allo stato C per ingresso "11". Infine A e B non possono essere equivalenti perché tale equivalenza è vincolata a quella tra B e C, che si è appena dimostrata non valere. In conclusione il numero di stati usati è minimo.

Si può per esempio assegnare la codifica mostrata nella tabella seguente:

Q_1Q_0	00	01	11	10
Init=00	00,00	00,00	11,00	00,01
C=01	00,01	00,01	00,10	00,01
A=11	00,01	00,01	10,00	00,01
B=10	00,01	00,01	01,00	00,01

Le funzioni di aggiornamento dello stato e per il calcolo delle uscite in forma minima risultano:

$$D_1 = \overline{Q_1}\overline{Q_0}ENEQ + Q_1Q_0ENEQ$$

$$D_0 = \overline{Q_0}ENEQ$$

$$OK = \overline{Q_1}Q_0ENEQ$$

$$ERR = Q_0\overline{EN} + Q_1\overline{EN} + EN\overline{EQ}$$

Appendice A: Esempi di quiz di teoria

- 1) L'equivalente in base 10 del numero in base 2 (MSB a sinistra) 101010, è:
 - a) 21
 - b) 42
 - c) 18
- 2) Quali dei seguenti numeri in base 2 (MSB a sinistra) è divisibile per 8:
 - a) 100010101001001
 - b) 100101110101010
 - c) 111010010001000
- 3) Quante cifre sono necessarie in un numero in base 8 (ottale) per rappresentare un numero binario composto da 3 parole (word) da 6 bit ciascuna?
 - a) 6
 - b) 3
 - c) 8
- 4) Data la funzione logica $F = E \cdot (A + C \cdot D)$, la funzione logica F^D , duale di F vale:
 - a) $F^D = E + (A \cdot C + D)$
 - b) $F^D = E + [A \cdot (C + D)]$
 - c) $F^D = E \cdot (A \cdot C + A \cdot D)$
- 5) Quale delle seguenti funzioni logiche è equivalente a $A \cdot B + \bar{A} \cdot C$
 - a) $A \cdot B + B \cdot C + \bar{A} \cdot B$
 - b) $A \cdot B + B \cdot C + \bar{A} \cdot C$
 - c) $B + C$
- 6) La funzione logica $F = a \cdot b \cdot c + a \cdot \bar{b} \cdot \bar{c}$, può essere semplificata come:
 - a) $F = a$
 - b) $F = a \cdot \bar{b}$
 - c) F non può essere ulteriormente semplificata come somma di prodotti
- 7) Un circuito decodificatore (decoder) è caratterizzato dal fatto di avere:
 - a) N ingressi e 2^N uscite
 - b) N uscite e 2^N ingressi
 - c) 1 ingresso, N segnali di enable e 2^N uscite
- 8) In una macchina sequenziale descritta da un diagramma di stati con N stati (nessuno dei quali equivalenti), il numero di registri necessari per memorizzare lo stato è:
 - a) almeno N
 - b) almeno $\log_2(N)$
 - c) almeno $N/2$
- 9) Considerata la tabella di transizione degli stati di una macchina sequenziale, se in corrispondenza di due stati si ha lo stesso valore delle uscite della macchina sequenziale:
 - a) i due stati sono equivalenti
 - b) i due stati sono equivalenti solo se anche gli stati futuri

- coincidono per ogni
configurazione degli
ingressi
- degli ingressi gli stati futuri
coincidono o sono a loro
volta equivalenti
- c) i due stati sono equivalenti
se per ogni configurazione

Soluzioni

1	2	3	4	5	6	7	8	9
b	c	a	b	b	c	a	b	c

