

Logica e Reti Logiche

Esercitazione

Francesco Pasquale

15 dicembre 2025

Esercizio 1. Progettare un circuito che prenda in input 4 bit, che rappresentano un numero in codifica binaria, e restituisca 1 se il numero in input è divisibile per *tre* e 0 altrimenti.

Esercizio 2. Si consideri la seguente funzione booleana:

$$y = bc + \bar{a}\bar{b}\bar{c} + b\bar{c} \quad (1)$$

1. Implementare la funzione in (1) usando soltanto un multiplexer 4:1;
2. Implementare la funzione in (1) usando un multiplexer 2:1, una porta OR e una porta NOT.

Esercizio 3. Costruire un circuito che prenda in input 3 bit e restituisca il bit di maggioranza (quindi deve restituire 1 se almeno due dei tre bit in input sono 1 e deve restituire 0 altrimenti).

Esercizio 4. Scrivere una formula in forma normale disgiuntiva (*somma di prodotti*) e una in forma normale congiuntiva (*prodotto di somme*) che corrispondano al circuito in Figura 1

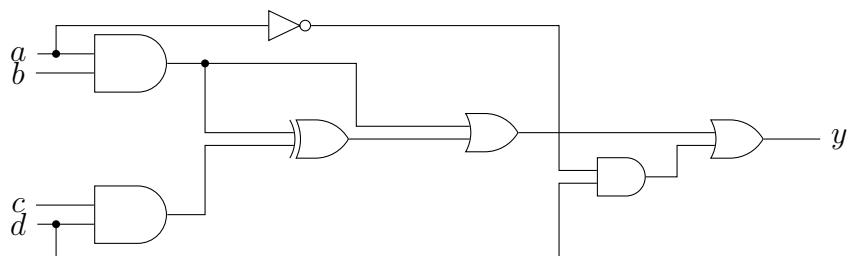


Figura 1: Un circuito combinatorio

Esercizio 5. Utilizzando soltanto blocchi *Half Adder* costruire un circuito che prenda in input un numero a 4 bit e lo incrementi di 1.

Esercizio 6. Costruire un circuito *Full Adder* utilizzando soltanto porte NOR.

Esercizio 7. Costruire un circuito che prenda in input un numero a 4-bit \mathbf{x} e restituisca il complemento a due di \mathbf{x} (per esempio, se $\mathbf{x} = 1010$ il circuito deve restituire $\mathbf{y} = 0110$).

Esercizio 8. Costruire un circuito che prenda in input un numero \mathbf{x} espresso in complemento a due a 4 bit e restituisca un numero in binario che rappresenti il valore assoluto di \mathbf{x} (per esempio, se $\mathbf{x} = 0110$ ossia il numero 6_{10} , il circuito deve restituire lo stesso numero, 0110 ; se invece $\mathbf{x} = 1011$ ossia il numero -5_{10} , il circuito deve restituire la rappresentazione binaria del numero 5_{10} : 0101).

Esercizio 9. Costruire un circuito che prenda in input due numeri \mathbf{x}_1 e \mathbf{x}_2 espressi in binario a 4 bit e restituisca 1 se $\mathbf{x}_1 \geq \mathbf{x}_2$ e 0 altrimenti.

Esercizio 10. Costruire un circuito che prenda in input un numero \mathbf{x} a 4 bit e restituisca il numero \mathbf{y} ottenuto facendo uno *shift* verso sinistra delle cifre di \mathbf{x} , riempiendo lo spazio rimanente a destra con uno zero (per esempio, se $\mathbf{x} = 0011$, il circuito deve restituire $\mathbf{y} = 0110$). Che operazione aritmetica sta eseguendo questo circuito?

Esercizio 11. Costruire un circuito che prenda in input un numero \mathbf{x} a 4 bit e un numero \mathbf{s} a 2 bit e faccia lo *shift* verso sinistra delle cifre di \mathbf{x} di tante posizioni quante sono quelle indicate da \mathbf{s} , riempiendo gli spazi rimanenti con degli zeri (per esempio, se $\mathbf{x} = 0011$, e $\mathbf{s} = 10$ il circuito deve fare lo shift verso sinistra di due posizioni, quindi restituire $\mathbf{y} = 1100$). Che operazione aritmetica sta eseguendo questo circuito?

Esercizio 12. Abbiamo costruito un *SR-Latch* utilizzando porte NOR. Cosa cambia se sostituiamo le porte NOR con porte NAND? Come costruiamo un *D-FlipFlop* utilizzando solo porte NAND e NOT?

Esercizio 13. Un *T-FlipFlop* ha un solo input, CLK, e un output Q . Ogni volta che CLK passa da 0 a 1, il valore di Q cambia (se è 0 diventa 1, se è 1 diventa 0). Progettare un *T-FlipFlop* utilizzando un *D-FlipFlop* e una porta NOT.

Esercizio 14. Un *JK-FlipFlop* ha un output, Q , e tre input: CLK, J e K . Ogni volta che CLK passa da 0 a 1, il valore di Q si aggiorna in questo modo:

- Se $J = K = 0$ allora Q mantiene il suo valore precedente;
- Se $J = K = 1$ allora Q cambia valore (se era 0 diventa 1, se era 1 diventa 0);
- Se $J = 1$ e $K = 0$ allora Q assume valore 1;
- Se $J = 0$ e $K = 1$ allora Q assume valore 0.

1. Progettare un *JK-FlipFlop* usando un *D-FlipFlop* e le opportune porte logiche;
2. Progettare un *D-FlipFlop* usando un *JK-FlipFlop* e le opportune porte logiche;
3. Progettare un *T-FlipFlop* usando un *JK-FlipFlop*.

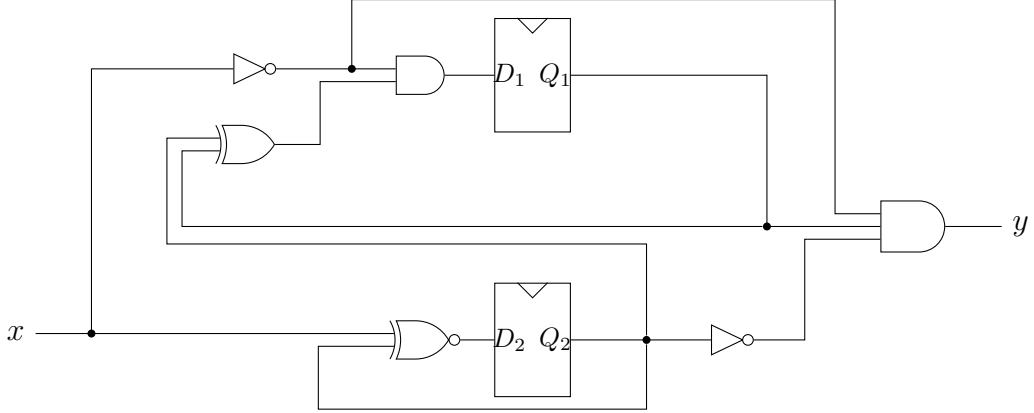


Figura 2: Un circuito sequenziale

Esercizio 15. Scrivere le equazioni, la tabella e il diagramma di stato del circuito in Figura 2.

Esercizio 16. Sia \mathcal{C} un circuito sequenziale con un *D-FlipFlop*, due input x_0, x_1 e un output y . L'output Q del *FlipFlop* coincide con l'output y del circuito, mentre l'input D del *FlipFlop* è descritto dalla seguente equazione booleana

$$D = y \oplus (x_0 + x_1).$$

1. Derivare la tabella di stato e il diagramma di stato del circuito;
2. Disegnare il circuito.

Esercizio 17. Un circuito sequenziale con un ingresso x , un'uscita y , e due *D-FlipFlop* è descritto dalle seguenti equazioni:

$$D_1 = Q_1 \overline{Q_2} + x \quad D_2 = (Q_1 \oplus Q_2) \overline{x} \quad y = Q_1 Q_2 x$$

dove Q_i e D_i indicano rispettivamente lo stato corrente e lo stato futuro dell' i -esimo *FlipFlop*, per $i = 1, 2$.

1. Derivare la tabella di stato e il diagramma di stato;
2. Disegnare il circuito.

Esercizio 18. Considerare il diagramma di stato di un automa a due stati in Figura 3

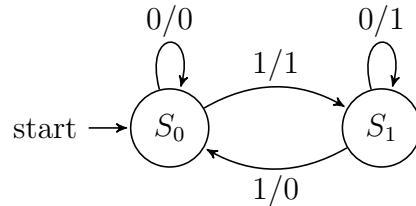


Figura 3: Un diagramma di stato

1. Qual è la sequenza dei primi sei bit in output (y_0, \dots, y_5) , se la sequenza dei primi sei bit in input (x_0, \dots, x_5) è $(0, 0, 0, 1, 1, 1)$? e se è $(1, 1, 1, 0, 0, 0)$? e se è $(0, 1, 0, 1, 0, 1)$?
2. Spiegare a parole cosa calcola l'automa;
3. Disegnare un circuito che implementi il diagramma di stato.

Esercizio 19. Progettare un automa a stati finiti “alla Mealy” che legga una sequenza di bit e restituisca 1 ogni volta che gli ultimi quattro bit della sequenza sono 1101 e 0 altrimenti: disegnare il diagramma di stato, scrivere la tabella e le equazioni di stato e disegnare lo schema del circuito.

Esercizio 20. Ripetere l'esercizio precedente progettando un automa “alla Moore” invece che “alla Mealy”.

Esercizio 21. Progettare un automa a stati finiti che legga una sequenza di bit e restituisca 1 ogni volta che negli ultimi tre bit letti c'è una maggioranza di uni e restituisca 0 altrimenti.

Esercizio 22. Progettare un circuito che implementi l'automa dell'Esercizio 21.

Esercizio 23. È possibile progettare un automa a stati finiti che legga una sequenza di bit in input (x_0, x_1, x_2, \dots) e restituisca in output una sequenza di bit (y_0, y_1, y_2, \dots) che rappresenti il complemento a due del numero in input (l'ordine con cui vengono letti e restituiti i bit va dal bit meno significativo al più significativo¹)? Se pensate che sia possibile progettate l'automa, altrimenti motivate la risposta.

Esercizio 24. Un *contatore binario* a n bit è un circuito sequenziale con un ingresso di RESET e n output, che rappresentano un numero fra 0 e $2^n - 1$ espresso in binario. Quando il RESET viene attivato, tutti gli n bit in output vengono inizializzati a 0; successivamente, ad ogni ciclo di clock gli n bit in output devono rappresentare un numero incrementato di 1 rispetto al precedente.

1. Progettare un contatore binario a n bit usando un sommatore a n bit e un registro a n bit;
2. Progettare un contatore a quattro bit usando solo blocchi HALF-ADDER e FLIP-FLOP.

¹Per esempio, se i primi quattro bit che riceve l'automa sono $x_0 = 1, x_1 = 1, x_2 = 0, x_3 = 0$, questi rappresentano il numero $0011_2 = x_02^0 + x_12^1 + x_22^2 - x_32^3 = 3_{10}$. L'automa dovrà restituire il complemento a due di questo numero, cioè $1101_2 = 1 * 2^0 + 0 * 2^1 + 1 * 2^2 - 1 * 2^3 = -3_{10}$, quindi la sequenza di bit $y_0 = 1, y_1 = 0, y_2 = 1, y_3 = 1$. Che sequenza dovrebbe restituire se il quinto bit che legge è $x_4 = 0$? e se invece il quinto bit è $x_4 = 1$?