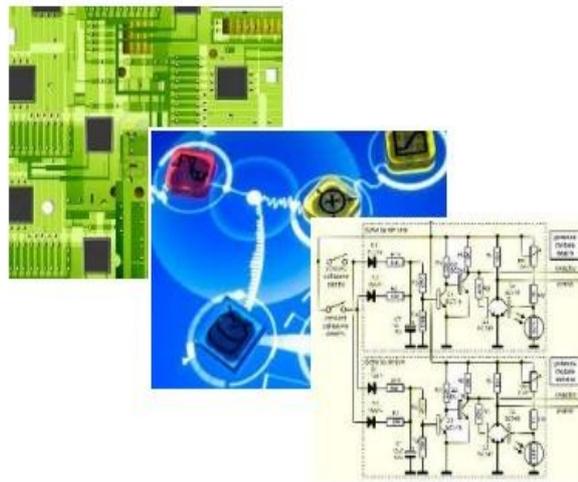


T ELECOM U N ICAZIONI I

Volume 1.2

Elettronica Digitale



CLASSI III INFORMATICA

ITIS V. Volterra

San Donà di Piave

Versione [2012/2013.v1]

PROF. SSA MARGHERITA DI LUCA

SOMMARIO

<u>PRINCIPI DELL'ALGEBRA DI BOOLE.....</u>	<u>3</u>
<u>LA LOGICA COMBINATORIA.....</u>	<u>6</u>
<u>I CIRCUITI INTEGRATI.....</u>	<u>17</u>
<u>CIRCUITI IN MEDIA SCALA DI INTEGRAZIONE.....</u>	<u>27</u>
<u>LA LOGICA SEQUENZIALE.....</u>	<u>33</u>
<u>CIRCUITI DI MEMORIA O SEQUENZIALI.....</u>	<u>33</u>
<u>SLIDE CU CIRCUITI APPLICATIVI DELLE CELLE DI MEMORIA.....</u>	<u>38</u>

PRINCIPI DELL'ALGEBRA DI BOOLE

Qualsiasi circuito logico è il risultato di una funzione logica, ossia di un insieme di variabili logiche opportunamente combinate tra di loro mediante operazioni di somma, prodotto e complementazione. Ogni variabile logica può assumere solo due valori:

stato logico 1 e stato logico 0;

essa è un segnale binario poiché contiene anche l'informazione dello stato logico.

Per indicare le variabili logiche si usano le lettere A,B,C,D,... ,

per indicare le funzioni logiche si usano le lettere X,Y,Z,...

Le variabili logiche possono essere sommate tra loro e/o moltiplicate.

Es: 1.1. $A+B$ oppure $A*B$

Le variabili logiche o le funzioni logiche complementate vengono indicate così: \bar{A} , \bar{Y} .

Sotto degli esempi di funzioni logiche a tre variabili.

Es: 1.2. $Y = \bar{A}\bar{B}\bar{C} + A\bar{B}\bar{C} + ABC$.

Es: 1.3. $Y = A\bar{B}C + \bar{A}BC + \bar{A}\bar{B}C + \bar{A}\bar{B}\bar{C}$.

1.1 TEOREMI E PROPRIETÀ DELL'ALGEBRA DI BOOLE

Le funzioni logiche, in quanto funzioni matematiche, godono delle seguenti proprietà:

1 Proprietà

Proprietà	Applicazione	
commutativa	$A + B = B + A$	$A * B = B * A$
associativa	$(A + B) + C = A + (B + C)$	$(A * B) * C = A * (B * C)$
distributiva	$(A * B) + (A * C) = A * (B + C)$	$(A + B) * (A + C) = A + (B * C)$

.3 Teoremi

Teorema	Applicazione	
identità	$A + 0 = A$	$A * 1 = A$
idempotenza	$A + A = A$	$A * A = A$
complementi	$A + \bar{A} = 1$	$A * \bar{A} = 0$
annullamento	$A + 1 = 1$	$A * 0 = 0$
assorbimento 1	$A + (AB) = A$	$A * (A + B) = A$
assorbimento 2	$A + (\bar{A}B) = A + B$	$A * (\bar{A} + B) = A * B$
De Morgan	$\bar{A} + \bar{B} = \overline{A * B}$	$\bar{A} * \bar{B} = \overline{A + B}$

Come si può notare ogni proprietà o teorema è esprimibile con due espressioni tra loro duali

1.1.1 DIMOSTRAZIONE DI ALCUNI TEOREMI

Teorema dell'assorbimento 1: $A + (AB) = A$

Il teorema si dimostra considerando nella prima parte dell'uguaglianza la variabile A e raccogliendola:

$$A + AB = A * \underbrace{(1 + B)}_{\text{uguale a 1}} = A * 1 = A$$

Teorema dell'assorbimento 2: $A + (\bar{A}B) = A + B$

Il teorema si dimostra considerando nella prima parte dell'uguaglianza la variabile A e, per il teorema precedente, sostituire al posto di A A+AB

$$A + (\bar{A}B) = A + AB + \bar{A}B = A + B \underbrace{(A + \bar{A})}_{\text{uguale a 1}} = A + \underbrace{B * 1}_{\text{uguale a B}} = A + B$$

Teoremi di De Morgan:

- $\bar{A} + \bar{B} = \overline{A * B}$

Il teorema si dimostra a partire dalle tabelle di verità delle due parti della uguaglianza

A	\bar{A}	B	\bar{B}	$\bar{A} + \bar{B}$
0	1	0	1	1
0	1	1	0	1
1	0	0	1	1
1	0	1	0	0

A	B	A*B	$\overline{A * B}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

- $\bar{A} * \bar{B} = \overline{A + B}$

Il teorema si dimostra a partire dalle tabelle di verità delle due parti della uguaglianza

A	\bar{A}	B	\bar{B}	$\bar{A} * \bar{B}$
0	1	0	1	1
0	1	1	0	0
1	0	0	1	0
1	0	1	0	0

1	1	1	0
---	---	---	----------

A	B	A+B	$\overline{A + B}$
0	0	0	1
0	1	1	0
1	0	1	0

LA LOGICA COMBINATORIA

1.2 CONNETTIVI LOGICI, CIRCUITI LOGICI E PORTE LOGICHE

1.2.1 AND

Il connettivo logico **E**, usato nella logica delle proposizioni, in Elettronica trova applicazione nel circuito logico seguente cui corrisponde la funzione logica AND e la corrispondente porta logica.

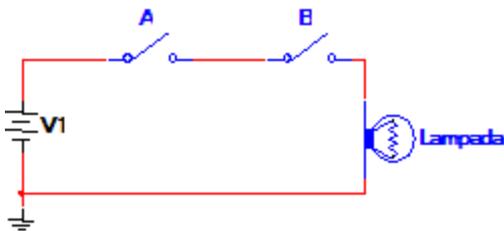


Fig. 1 Circuito logico AND

La lampada si accenderà solo se entrambi gli interruttori **A e B** saranno chiusi, in modo da consentire alla corrente di percorrere l'intero circuito.

Il simbolo logico che esprime il comportamento del circuito sopra mostrato è il seguente e la funzione è espressa dall'operazione prodotto:

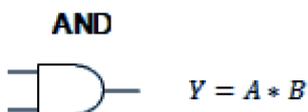


Fig. 2 Porta logica AND

Alla funzione inoltre viene associata anche una tabella di verità che mostra per ogni stato delle variabili la risposta dell'uscita.

Tabella 4 Tabella di verità

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

1.2.2 OR

Il connettivo logico **O**, usato nella logica delle proposizioni, in Elettronica trova applicazione nel circuito logico seguente cui corrisponde la funzione logica OR e la corrispondente porta logica.

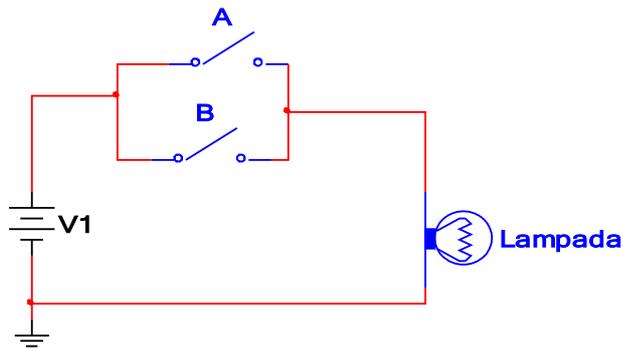


Fig. 3 Circuito logico OR

La lampada si accenderà solo se gli interruttori **A o B** saranno chiusi, in modo da consentire alla corrente di percorrere l'intero circuito.

Il simbolo logico che esprime il comportamento del circuito sopra mostrato è il seguente e la funzione è espressa dall'operazione somma:

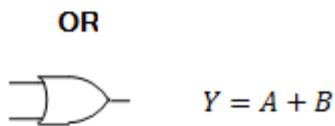


Fig. 4 Porta logica OR

Alla funzione viene associata, anche in questo caso la tabella di verità che mostra per ogni stato delle variabili la risposta dell'uscita .

4 Tabella di verità

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

1.2.3 NOT

Il connettivo logico **NEGAZIONE** , usato nella logica delle proposizioni, in Elettronica trova applicazione nel circuito logico seguente cui corrisponde la funzione logica NOT e la corrispondente porta logica.

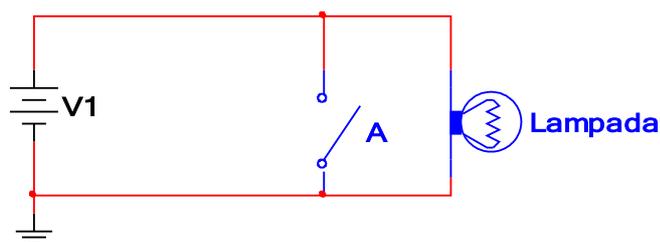


Fig. 5 Circuito logico NOT

La lampada si accenderà solo se l' interruttore **A** sarà aperto , in modo da consentire alla corrente di percorrere l'intero circuito.

Il simbolo logico che esprime il comportamento del circuito sopra mostrato è il seguente e la funzione è espressa dall'operazione **complementazione** o **negazione**:

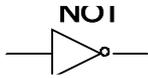


Fig. 6 Porta logica NOT

$$Y = \bar{A}$$

Alla funzione viene associata, anche in questo caso la tabella di verità che mostra per ogni stato della variabile la risposta dell'uscita

5 Tabella di verità

A	Y
0	1
1	0

1.3 ALTRE PORTE LOGICHE

Le porte logiche AND, OR, NOT, sono dette porte logiche **fondamentali**.

Combinandole opportunamente si ottengono altre porte logiche come la NAND, la NOR, la X-OR e la X-NOR.

1.3.1 NAND

La porta NAND è ottenuta combinando una porta AND e una porta NOT. Il risultato è il prodotto complementato di due variabili.

Il simbolo logico che esprime tale comportamento è il seguente e la funzione è espressa dall'operazione prodotto complementato:

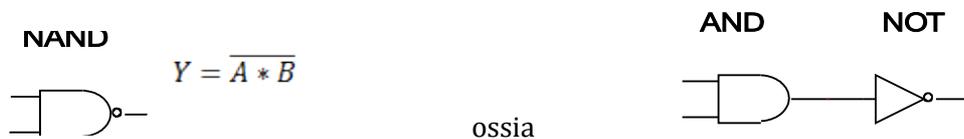


Fig. 7 Porta logica NAND

Alla funzione inoltre viene associata anche una tabella di verità che mostra per ogni stato delle variabili la risposta dell'uscita

6 Tabella di verità

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

1.3.2 NOR

La porta NOR è ottenuta combinando una porta OR e una porta NOT. Il risultato è la somma complementata di due variabili.

Il simbolo logico che esprime tale comportamento è il seguente e la funzione è espressa dall'operazione somma complementata:

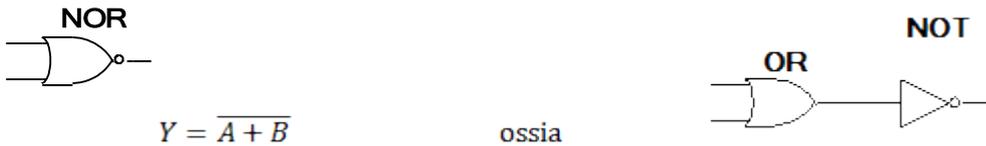


Fig. 8 Porta logica NOR

Alla funzione inoltre viene associata anche una tabella di verità che mostra per ogni stato delle variabili la risposta dell'uscita

7 Tabella di verità

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

1.3.3 XOR

La porta XOR è ottenuta combinando delle porte AND una porta OR e delle porte NOT. Il risultato è la somma esclusiva di due variabili.

Il simbolo logico che esprime tale comportamento è il seguente:

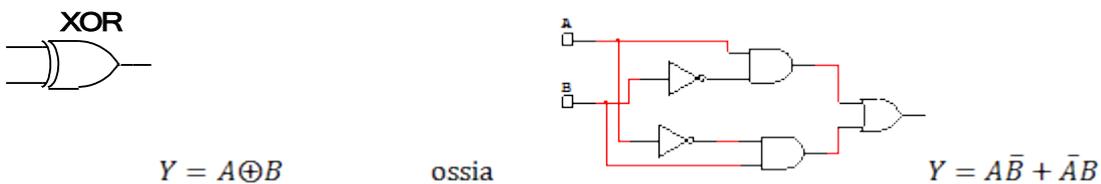


Fig. 9 Porta logica XOR

Alla funzione inoltre viene associata anche una tabella di verità che mostra per ogni stato delle variabili la risposta dell'uscita .

8 Tabella di verità

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

La porta logica confronta due variabili e dà risposta positiva quando le due variabili assumono valore diverso

1.3.4 XNOR

La porta XNOR è ottenuta combinando delle porte AND una porta OR e delle porte NOT. Il risultato è la somma esclusiva complementata di due variabili.

Il simbolo logico che esprime tale comportamento è il seguente:

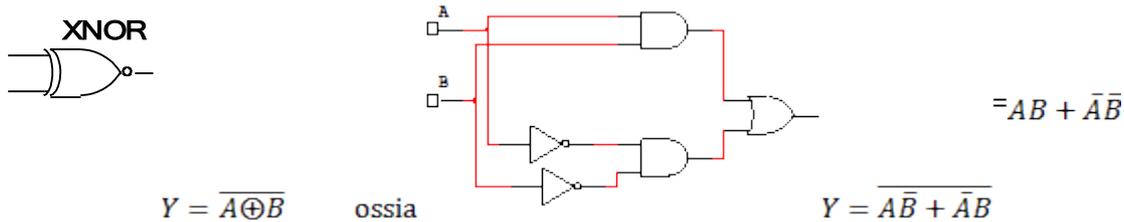


Fig. 10 Porta logica XOR

Alla funzione inoltre viene associata anche una tabella di verità che mostra per ogni stato delle variabili la risposta dell'uscita

9 Tabella di verità

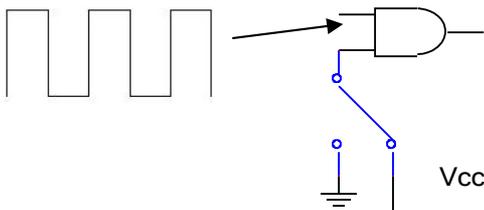
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

La porta logica confronta due variabili e dà risposta positiva quando le due variabili assumono valore uguale

1.4 GATING DI SEGNALE

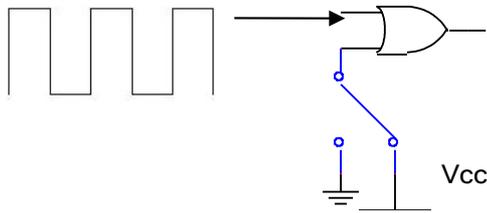
E' possibile realizzare un **gating** di segnale quando ai due ingressi di una porta logica si applicano rispettivamente un segnale digitale e un livello di tensione, detto **segnale di abilitazione**. La combinazione dei segnali di ingresso permette di trasferire il segnale digitale in uscita a seconda del livello del segnale di abilitazione, 0 o 1, applicato all'ingresso.

1.4.1 GATING CON PORTA AND



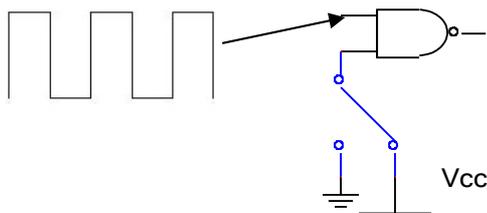
Quando l'interruttore è su massa $\underline{\underline{0}}$ l'uscita è sempre 0 , quando l'interruttore è su V_{cc} l'uscita coincide con il segnale applicato all'altro ingresso. Quindi **si ha trasferimento di segnale in uscita quando l'ingresso di abilitazione è a livello alto (1 logico).**

1.4.2 GATING CON PORTA OR



Quando l'interruttore è su V_{cc} l'uscita è sempre 1 , quando l'interruttore è su $\underline{\underline{0}}$ l'uscita coincide con il segnale applicato all'altro ingresso. Quindi **si ha trasferimento di segnale in uscita quando l'ingresso di abilitazione è a livello basso(0 logico).**

1.4.3 GATING CON PORTA NAND



Quando l'interruttore è su $\underline{\underline{0}}$ l'uscita è sempre 1 , quando l'interruttore è su V_{cc} l'uscita coincide con il segnale applicato all'altro ingresso complementato . Quindi, per riottenere il segnale applicato in ingresso, è necessario invertirlo ancora , usando una NAND come se fosse una NOT.

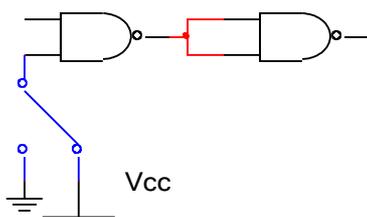


Fig. 11 Gating completo con porte NAND

1.5 LIVELLI LOGICI

Ciascuna variabile logica, sia in ingresso che in uscita ad un circuito, può assumere un valore di tensione corrispondente ad uno dei due livelli logici, **zero** e **uno**. Se al valore di tensione più alto si associa l'**uno**, e al valore più basso lo **zero**, si parla di **logica positiva**. Se al valore di tensione più basso si associa l'**uno**, e al valore più alto lo **zero**, si parla di **logica negativa**.

N.B. non esiste una corrispondenza precisa tra valore di tensione e livello logico, quindi si associa a ciascun livello logico una fascia di valori.

1.6 SEMPLIFICAZIONE DELLE FUNZIONI LOGICHE

Applicando le regole delle proprietà e dei teoremi visti in precedenza, è possibile semplificare le funzioni logiche e successivamente ricavarne il circuito logico, come pure procedere in senso inverso, ossia risalire dal circuito alla funzione che lo ha generato.

Per fare ciò però è necessario avere la funzione espressa in *modo canonico*.

Definizione 1.1. *Si definisce forma canonica di una funzione l'espressione in cui compaiono in tutti i termini, tutte le variabili.*

Si possono avere due forme canoniche :

- la prima, espressa come somma di **mintermini**
- la seconda, espressa come prodotto di **maxtermini**

$$\text{Es: 1} \quad Y = \underbrace{\bar{A}\bar{B}\bar{C}}_{\text{mintermine}} + \underbrace{\bar{A}BC}_{\text{mintermine}} + \underbrace{ABC}_{\text{mintermine}}$$

prima forma canonica

$$\text{Es: 1.5} \quad Y = \underbrace{(\bar{A} + B + \bar{C})}_{\text{maxtermini}} * \underbrace{(A + \bar{B} + C)}_{\text{maxtermini}} * \underbrace{(A + B + C)}_{\text{maxtermini}}$$

seconda forma canonica

Data una qualsiasi funzione, espressa in forma non canonica, è possibile trasformarla in forma canonica sfruttando i teoremi dell'algebra di Boole; in particolare si sfrutta il **teorema dei complementi**.

$$\text{Es: 1.6} \quad Y = \bar{A}\bar{B}\bar{C} + \bar{A}B + BC$$

basta moltiplicare il secondo mintermine per $(C + \bar{C})$ ed il terzo mintermine per $(A + \bar{A})$ e successivamente sviluppare tutti i prodotti.

$$Y = \bar{A}\bar{B}\bar{C} + \bar{A}B * (C + \bar{C}) + BC * (A + \bar{A})$$

$$Y = \bar{A}\bar{B}\bar{C} + \bar{A}BC + \bar{A}B\bar{C} + ABC + \bar{A}BC.$$

Se nello sviluppo dei prodotti compare più volte lo stesso mintermine, esso andrà considerato una volta soltanto.

Quando viene assegnata una funzione logica e di essa si compila la tabella di verità, ciascuna combinazione (prodotto) delle variabili assegnate si chiama **mintermine**

Es: 1.7

A	B	Y	mintermini
0	0	1	AB
0	1	0	
1	0	1	$A\bar{B}$

1	1	0	
---	---	---	--

$$Y = \bar{A}\bar{B} + A\bar{B}$$

la funzione logica Y è il risultato della somma dei mintermini relativi ai casi in cui la Y è uguale a 1

N.B. In una generica funzione logica a n variabili, l'uscita è sempre esprimibile nella somma di mintermini (prima forma canonica) . Ogni mintermine deve contenere tutte le variabili di ingresso moltiplicate tra loro e considerate complementate se uguali a zero, non complementate se uguali a uno.

1.7 ESERCIZI SULLE FUNZIONI LOGICHE

i) Risalire alla prima forma canonica della seguente funzione logica:

$$(a) Y = A\bar{B}C + AB + B\bar{C}$$

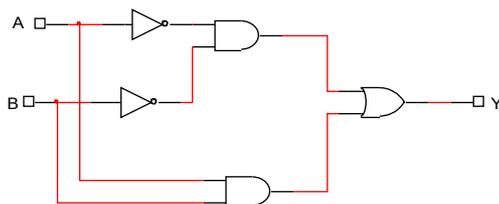
ii) Semplificare mediante i teoremi dell'algebra di Boole le seguenti funzioni logiche:

$$(a) Y = \bar{A}\bar{B}\bar{C} + ABC + A\bar{B}C + \bar{A}B\bar{C}$$

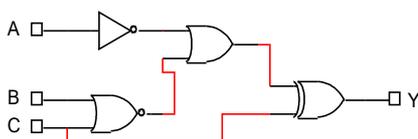
$$(b) Y = \bar{A}\bar{B} + \bar{A}B\bar{C} + A\bar{B}\bar{C} + AB + A\bar{B}C$$

iii) Realizzare gli schemi logici delle funzioni precedentemente semplificate

iv) Dato il seguente schema ricavare la funzione logica corrispondente



v) Dato il seguente schema ricavare la funzione logica e la corrispondente tabella di verità

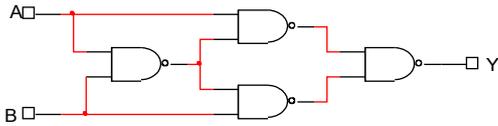


vi) Applicare il teorema di DE MORGAN alle seguenti espressioni:

$$(1) \overline{\bar{A}\bar{B}\bar{C}D}$$

$$(2) \overline{(\bar{A} + \bar{B})}C$$

vii) Verificare che lo schema a porte NAND della figura sottostante è equivalente alla funzione XOR



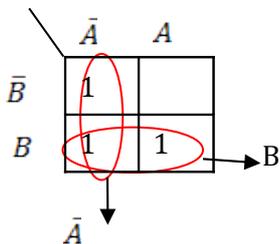
1.8 MAPPE DI KARNAUGH

Le mappe di Karnaugh sono uno strumento utile a semplificare le funzioni logiche . Si possono costruire per 2 , 3, 4 variabili

Es. nel caso di funzione a due variabili

	\bar{A}	A
\bar{B}	$\bar{A}\bar{B}$	$A\bar{B}$
B	$\bar{A}B$	AB

1. Si costruisce una mappa di due righe e due colonne , a ciascuna riga e colonna si assegna una variabile complementata o non complementata. Ogni cella sarà identificata da coordinate di riga e colonna come nell'esempio.
2. Ad ogni cella corrisponderà un mintermine della tabella di verità relativa alla funzione logica
3. Si compila quindi la mappa inserendo gli uno della colonna y della tabella di verità nelle relative celle, e lasciando vuote le celle che corrisponderanno agli zeri della stessa colonna .
4. Dopo la compilazione si raggruppano le celle adiacenti per riga o per colonna , riportando per ciascun raggruppamento la variabile che non si modifica ossia l'intestazione di riga se il raggruppamento viene fatto in riga, o l'intestazione di colonna se il raggruppamento è fatto in colonna.



La funzione minima sarà: $Y_{min} = \bar{A} + B$

Es. nel caso di funzione a tre variabili

$\bar{A}\bar{B}$	$\bar{A}B$	AB	$A\bar{B}$
------------------	------------	------	------------

\bar{C}	$\bar{A}\bar{B}\bar{C}$	$\bar{A}B\bar{C}$	$A\bar{B}\bar{C}$	$AB\bar{C}$
C	$\bar{A}\bar{B}C$	$\bar{A}BC$	ABC	$AB\bar{C}$

1. Si costruisce una mappa di due righe e quattro colonne , a ciascuna riga e colonna si assegna una variabile o un gruppo di variabili complementate o non complementata, come nello schema sopra mostrato . Ogni cella sarà identificata da coordinate di riga e colonna come nell'esempio.
2. Ad ogni cella corrisponderà un mintermine della tabella di verità relativa alla funzione logica
3. Si compila quindi la mappa inserendo gli uno della colonna y della tabella di verità nelle relative celle, e lasciando vuote le celle che corrisponderanno agli zeri della stessa colonna .

Dopo la compilazione si raggruppano le celle adiacenti per riga o per colonna , riportando per ciascun raggruppamento la variabile o il gruppo di variabili che non si modifica ossia l'intestazione di riga se il raggruppamento viene fatto in riga, o l'intestazione di colonna se il raggruppamento è fatto in colonna. Le colonne più esterne sono considerate adiacenti in quanto si modifica una sola variabile.

	$\bar{A}\bar{B}$	$\bar{A}B$	AB	$A\bar{B}$	
\bar{C}	1			1	→ \bar{B}
C	1	1	1	1	

La funzione minima sarà: $Y_{min} = \bar{B} + C$

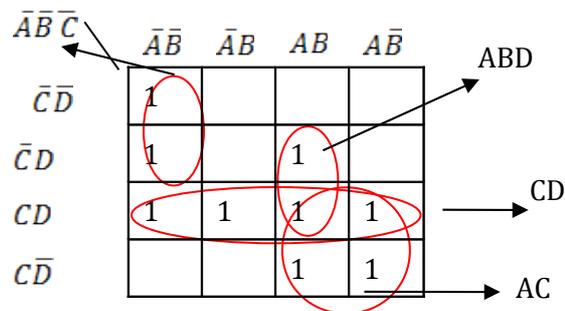
Es. nel caso di funzione a quattro variabili

	$\bar{A}\bar{B}$	$\bar{A}B$	AB	$A\bar{B}$
$\bar{C}\bar{D}$	$\bar{A}\bar{B}\bar{C}\bar{D}$	$\bar{A}B\bar{C}\bar{D}$	$AB\bar{C}\bar{D}$	$A\bar{B}\bar{C}\bar{D}$
$\bar{C}D$	$\bar{A}\bar{B}\bar{C}D$	$\bar{A}B\bar{C}D$	$AB\bar{C}D$	$A\bar{B}\bar{C}D$
CD	$\bar{A}\bar{B}CD$	$\bar{A}BCD$	$ABCD$	$A\bar{B}CD$
$C\bar{D}$	$\bar{A}\bar{B}C\bar{D}$	$\bar{A}BC\bar{D}$	$ABC\bar{D}$	$A\bar{B}C\bar{D}$

1. Si costruisce una mappa di quattro righe e quattro colonne , a ciascuna riga e colonna si assegna un gruppo di variabili complementate o non complementate, come nello schema sopra mostrato . Ogni cella sarà identificata da coordinate di riga e colonna come nell'esempio.
2. Ad ogni cella corrisponderà un mintermine della tabella di verità relativa alla funzione logica

3. Si compila quindi la mappa inserendo gli uno della colonna y della tabella di verità nelle relative celle, e lasciando vuote le celle che corrisponderanno agli zeri della stessa colonna .

Dopo la compilazione si raggruppano le celle adiacenti per riga o per colonna , riportando per ciascun raggruppamento la variabile o il gruppo di variabili che non si modifica ossia l'intestazione di riga se il raggruppamento viene fatto in riga, o l'intestazione di colonna se il raggruppamento è fatto in colonna. Le colonne più esterne e le righe più esterne sono considerate adiacenti in quanto si modifica una sola variabile.



La funzione minima sarà: $Y_{min} = \bar{A}\bar{B}\bar{C} + ABD + CD + AC$

N.B. Nella costruzione della mappa la sola regola da rispettare è quella di far cambiare una sola variabile per volta e nei raggruppamenti di raccogliere il maggior numero di celle purché siano in numero pari alle potenze del due (1,2,4,8,...).

1.9 ESERCIZI SULLE MAPPE DI KARNAUGH

- i) Semplificare con le mappe di Karnaugh le seguenti funzioni logiche:

(a) $\bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + \bar{A}B\bar{C} + ABC$

(b) $BC\bar{D} + CD + \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}C$

(c) $\bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}C\bar{D}$

(d) $A\bar{B}D + BCD + C\bar{D}$

- ii) Costruire la mappa relativa alle seguenti funzioni

(1) $Y = A + C + BD$

(2) $Y = AC + BD + A\bar{D}$

- iii) Date le seguenti mappe ricavare le funzioni minime corrispondenti

	$\bar{A}\bar{B}$	$\bar{A}B$	AB	$A\bar{B}$
$\bar{C}\bar{D}$			1	
$C\bar{D}$	1	1	1	

CD		1	1	1
$C\bar{D}$		1		

	$\bar{A}\bar{B}$	$\bar{A}B$	AB	$A\bar{B}$
$\bar{C}\bar{D}$	1		1	1
$\bar{C}D$	1	1	1	1

CD	1	1	1	1
$C\bar{D}$	1	1		1

I CIRCUITI INTEGRATI

1.10 CLASSIFICAZIONE DEI CIRCUITI INTEGRATI

1.10.1 FAMIGLIE LOGICHE

L'evoluzione tecnologica ha prodotto circuiti via via più complessi, ma che utilizzano spazi sempre più ridotti. Da queste necessità si è originata la classificazione dei circuiti integrati a seconda del numero di porte integrate in un singolo chip.

- ✓ piccola scala di integrazione SSI con meno di 12 porte logiche
- ✓ media scala di integrazione MSI con meno di 100 porte logiche
- ✓ larga scala di integrazione LSI con meno di 1000 porte logiche
- ✓ grandissima scala di integrazione VLSI con più di 1000 porte logiche

E' possibile inoltre classificare i circuiti integrati a seconda della tecnologia usata per la loro realizzazione.

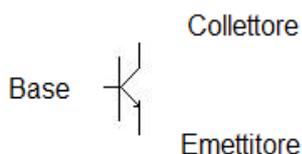
Ne esaminiamo due, le più comuni:

- TTL
- CMOS

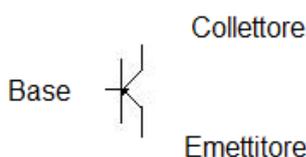
Le prime utilizzano **transistor a giunzione bipolare BJT**, le seconde usano **transistor unipolari MOS**.

Il transistor BJT è un dispositivo a **semiconduttore**¹ in cui sono presenti 3 differenti zone di drogaggio. A seconda della disposizione delle diverse zone di drogaggio si ottengono due configurazioni PNP ed NPN, dove N e P sono rispettivamente zone a drogaggio negativo o positivo:

BJT NPN



BJT PNP



Il drogaggio connota elettricamente un materiale altrimenti elettricamente neutro.

¹ Semiconduttore = elemento caratterizzato dall'aver proprietà intermedie tra conduttori e isolanti. Può presentare nella struttura atomica cariche mobili sia positive che negative.

Il funzionamento del transistor pur essendo molto complesso può essere così semplificato:

- la base controlla il passaggio delle cariche dal collettore all'emettitore
- se la base è polarizzata direttamente, circola corrente (trascurabile) sia dalla base verso l'emettitore che dal collettore all'emettitore
- se la base è polarizzata inversamente, non circola alcuna corrente.

Pertanto in un BJT si hanno tre correnti:

- ✓ I_B corrente di base
- ✓ I_C corrente di collettore
- ✓ I_E corrente di emettitore.

Queste correnti sono tra loro legate dalla seguente relazione $I_E = I_C + I_B$.

Si definisce **guadagno statico di corrente** il rapporto tra I_C e I_B : $h_{fe} = I_C / I_B$.

Perché un BJT possa funzionare deve essere polarizzato, ossia bisogna imporre, alle giunzioni che compongono il dispositivo, delle tensioni che garantiscano il passaggio di cariche e quindi le correnti I_C e I_B

Il transistor può funzionare in due modalità diverse:

- in modo lineare, ossia con proporzionalità diretta tra le tensioni di ingresso e di uscita; si dice che si comporta come **amplificatore**.
- in modo non lineare ossia senza proporzionalità diretta tra le tensioni di ingresso e di uscita che assumono solo due valori basso e alto; il circuito si comporta come un **interruttore chiuso o aperto**.

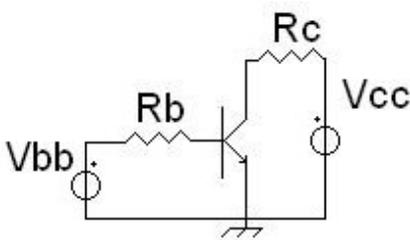
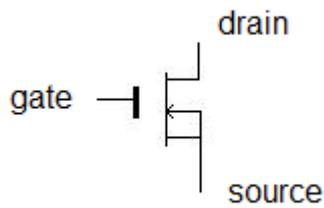


Fig. 12 Circuito di polarizzazione del BJT

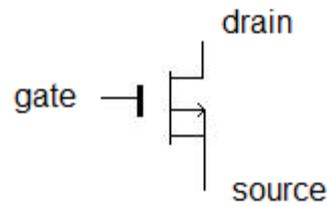
I MOS sono invece componenti unipolari, comandati in tensione, in cui il passaggio di corrente viene gestito da un canale di tipo N o P che può essere chiuso o aperto a seconda della tensione imposta ai morsetti.

I morsetti si chiamano rispettivamente **gate, drain, source**.

MOS a canale N



MOS a canale P



Per i circuiti integrati si usano entrambi i tipi di MOS collegati in modo complementare; pertanto si parla di CMOS.

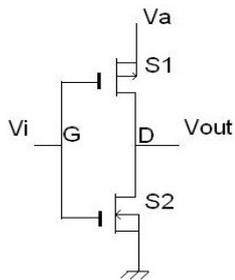


Fig. 13 Circuito di polarizzazione di un MOS

Mentre i BJT possono essere alimentati solo con una tensione di 5 V , i CMOS sono più versatili in quanto l'alimentazione può variare in un campo ampio, da 3 a 18 V.

1.10.2 FAMIGLIA LOGICA TTL

La famiglia logica TTL comprende due serie di circuiti integrati : la serie 54 (militare) e la serie 74 (commerciale). Entrambe vanno alimentate a 5 V, anche se hanno parametri che spaziano in range differenti e che sono ricavabili dai *data sheets*.

Le due serie sono disponibili in diverse versioni, riconoscibili per le seguenti sigle:

Standard	--
Alta velocità	L
Bassa potenza	LP
Schottky	S
Schottky a bassa potenza	LS
Advanced Schottky	AS
Advanced Schottky a bassa potenza	ALS

Un circuito integrato appartenente alla famiglia 74 si può quindi riconoscere da un codice così fatto:



1.10.3 LA FAMIGLIA CMOS

la famiglia CMOS ha una serie commerciale la 4000 ed è disponibile in tre versioni

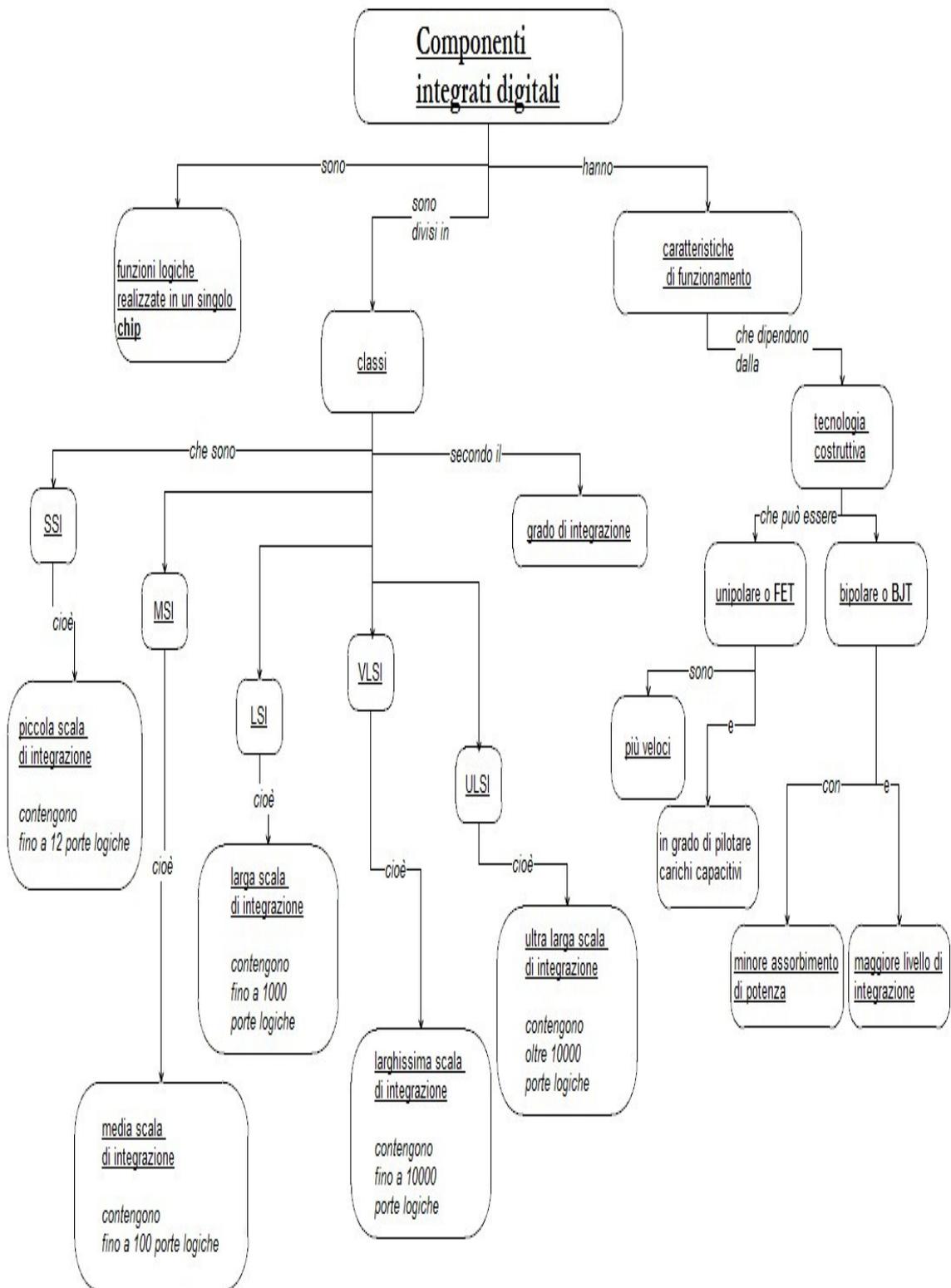
A	In disuso
B	Con Buffer in uscita
UB	UnBuffered (non bufferata) la migliore

Esistono anche le serie 54/74 CMOS con la piedinatura simile alle TTL nelle versioni:

54/74	C
54/74	HC
54/74	HCT

Vanno alimentate con tensione variabile tra 3 e 18 Volt; si considera livello basso una tensione inferiore a $\frac{1}{3}V_{CC}$ e livello alto una tensione superiore a $\frac{2}{3}V_{CC}$, dove per V_{CC} si intende il valore della tensione di alimentazione scelto.

Nella pagina successiva è riportato una mappa riassuntiva sia delle classificazioni che della tecnologia usata per costruire i circuiti digitali



1.11 PARAMETRI DEI CIRCUITI INTEGRATI

Nello studio dei circuiti digitali molti sono i parametri che vengono presi in considerazione: il **margine di rumore**, le **correnti di ingresso ed uscita**, le **caratteristiche di trasferimento**, i **tempi di risposta** all'applicazione di segnali.

Poiché un circuito integrato per funzionare correttamente viene alimentato e le informazioni sono di tipo digitale, si considerano sempre due valori di tensione riferibili al livello logico alto (uno logico) e livello logico basso (zero logico). Inoltre tale distinzione viene applicata sia se il segnale viene rilevato in ingresso che in uscita.

Definizione 1.2. Si definisce margine di rumore l'intervallo di tensione misurabile, sia a livello alto che basso, tra l'uscita di una data porta logica (out) e l'ingresso (in) della porta successiva ad essa collegata.

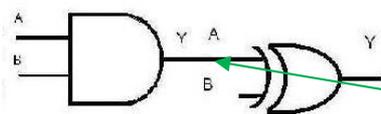


Fig. 14 Esempio di circuito logico

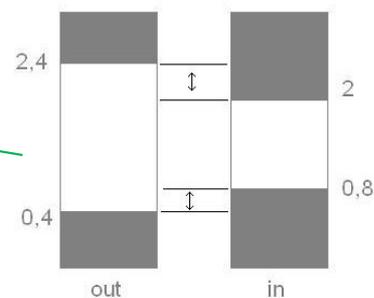


Fig. 15 Rappresentazione del margine di rumore con il grafico delle tensioni misurato nel punto indicato

1

Definizione 1.3. Si definisce caratteristica di trasferimento o caratteristica ingresso/ uscita il grafico che esprime la tensione in uscita ad una porta logica in funzione di quella di ingresso, supposti gli altri ingressi a livello fisso.

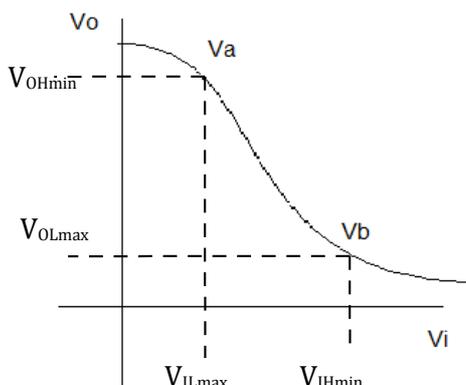


Fig. 16 Caratteristica di trasferimento di una NOT

Legenda

- V_{OHmin} minima tensione a livello alto che la porta fornisce in uscita (2,4V TTL)
- V_{OLmax} massima tensione a livello basso che la porta fornisce in uscita (0,4V TTL)
- V_{ILmax} massima tensione a livello basso che la porta fornisce in ingresso (0,8V TTL)
- V_{IHmin} minima tensione a livello alto che la porta fornisce in ingresso (2V TTL)

Tutti i valori compresi tra V_a e V_b sono considerati di **transizione** ossia né a livello basso, né a livello alto e vengono riconosciuti come errori se minori di V_{OHmin} o maggiori di V_{OLmax} .

Definizione 1.4. Si definiscono **correnti di ingresso e di uscita** le correnti che fluiscono attraverso le porte. Esse possono essere **erogate**(di *source*) o **assorbite**(di *sink*).

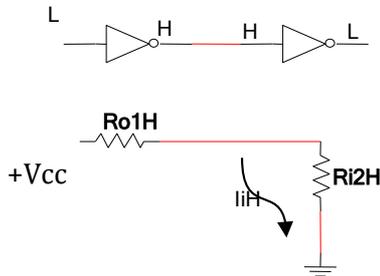


Fig. 17 L'uscita di una porta a livello H eroga corrente all'ingresso successivo

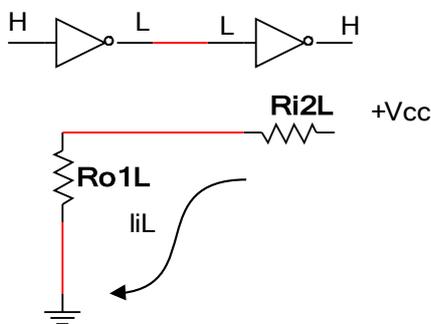


Fig. 18 l'uscita di una porta a livello L assorbe corrente dall'ingresso successivo.2

Se un'uscita a livello H viene collegata a più ingressi, eroga una corrente pari alla somma delle correnti assorbite dagli ingressi successivi. Questo impone dei limiti al numero degli ingressi collegabili ad una sola uscita.

Se un'uscita a livello L viene collegata a più ingressi, assorbe una corrente pari alla somma delle correnti erogate dagli ingressi successivi. Ci sono dei limiti anche in questi casi.

Definizione 1.5. Si definisce **FAN-OUT** il numero massimo degli ingressi collegabili ad una sola uscita.

Fan-out a livello H = $\frac{I_{OHmax}}{I_{IHmax}}$ per le TTL $\frac{400 \mu A}{40 \mu A}$ (quindi massimo 10 ingressi collegabili ad una uscita)

Fan-out a livello L = $\frac{I_{OLmax}}{I_{ILmax}}$ per le TTL $\frac{16mA}{1,6mA}$ (quindi massimo 10 ingressi collegabili ad una uscita)

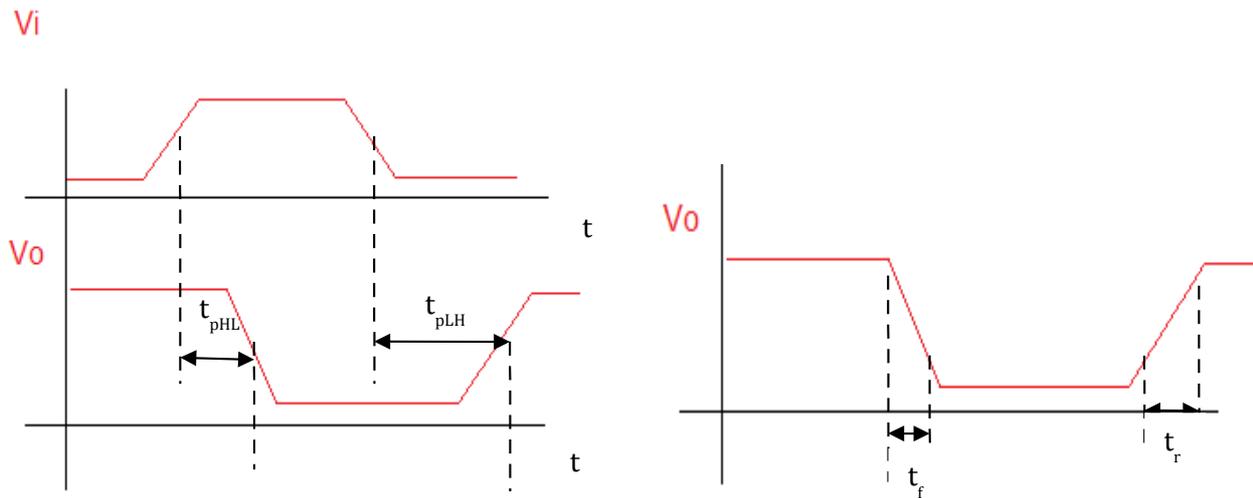
Ogni porta logica fornisce in uscita una risposta in un tempo che può essere misurato e che dipende da fenomeni intrinseci e dal circuito stesso.

Questo tempo può essere così classificato:

- Tempo di propagazione t_{pLH} e t_{pHL}

- Tempo di salita t_r (rise time)
- Tempo di discesa t_f (fall time)

Definizione 1.6. Si definisce tempo di propagazione il tempo che impiega il segnale ad attraversare la porta. Si misura tra il 50% della variazione del segnale in ingresso e il 50% della corrispondente variazione in uscita.



Il tempo di **salita** e di **discesa** misurano il tempo che il segnale impiega a commutare tra un livello e l'altro e si misurano tra il 10% e il 90% della variazione.

1.12 UN TEST PER VERIFICARE LA CONOSCENZA

1. Una porta a due ingressi che fornisce un livello basso solo quando entrambi gli ingressi sono alti è una

- | | |
|---------|----------|
| A. NOR | B. OR |
| C. NAND | D. EX-OR |

2. Una porta in grado di rilevare con un livello alto l'uguaglianza dei livelli sugli ingressi è la

- | | |
|----------|-----------|
| A. NOR | B. NAND |
| C. EX-OR | D. EX-NOR |

3. La famiglia TTL deve essere alimentata con

- | | |
|-----------------------|---------------------|
| A. 3,3 V stabilizzati | B. 5 V stabilizzati |
| C. 5 ÷ 18 V | D. 2 ÷ 6 V |

4. Un segnale analogico può assumere:

- | | |
|-----------------------------|---|
| A. numerosi valori distinti | B. tutti gli infiniti valori compresi in un intervallo continuo |
| C. due soli valori | D. un valore infinito |

5. Vengono chiamati terminali di ingresso di un circuito quelli da cui:

- A. si entra nel circuito
B. la corrente entra nel circuito
C. il circuito riceve i segnali da
D. il circuito riceve energia rielaborare

6. La tabella di verità permette di rappresentare :

- A. tutte le informazioni relative ad un circuito
B. la relazione tra i segnali di ingresso e di uscita di un qualsiasi circuito
C. la successione nel tempo degli stati di una rete digitale
D. la relazione tra i segnali di ingresso e di uscita di una rete combinatoria

7. Quante combinazioni di livelli di ingresso sono rappresentate nella tabella di verità di una rete combinatoria a 5 ingressi?

- A. 32
B. 5
C. 10
D. 25
E. 16
F. 64

8. La sigla TTL è utilizzata in elettronica per indicare

- A. un particolare componente
B. un tipo di porta logica
C. una famiglia di circuiti integrati digitali
D. un contenitore per circuiti integrati

9. La funzione di un integrato digitale è identificata

- A. dalla forma del contenitore
B. dal codice alfanumerico stampato sul contenitore
C. dalle barre colorate poste sul contenitore
D. dal numero dei pin

10. Quale sigla identifica comunemente il terminale di un integrato da collegare all'alimentazione?

- A. GND
B. DIP
C. Vcc
D. A

11. Si desidera ottenere un'uscita ALTA quando i due ingressi sono allo stesso livello. In tal caso bisogna usare una porta:

- A. AND
B. EX-OR
C. EX-NOR
D. NOR

12. La minimizzazione è utile per:

- A. effettuare un disegno più piccolo
B. evitare l'uso di una forma canonica
C. realizzare un circuito con meno componenti possibili
D. evitare complicazioni matematiche

13. Se una tensione è a livello H vuol dire che:

- A. siamo in logica positiva
B. siamo in logica negativa
C. è a livello basso
D. è a livello alto

14. Una tensione a livello L è:

- A. una tensione di 0 V
B. una tensione di 5 V
C. una tensione il cui valore massimo consentito è inferiore al valore minimo consentito a livello H
D. una tensione che normalmente è più piccola di una a livello H

15. Una forma canonica è:

- A. una espressione logica in forma minima
B. una espressione logica che soddisfa la funzione a cui si riferisce

C. una espressione che non contiene in tutti i termini tutte le variabili

D. una espressione logica, sempre possibile per ogni funzione, i cui termini contengono ognuno tutte le variabili

16. Il tempo di salita di una porta si può definire come

A. il tempo necessario all'uscita per portarsi dallo 0% al 50% della variazione prevista

B. il tempo necessario all'ingresso per portarsi dal 10% al 90% della variazione prevista

C. il tempo necessario all'uscita per portarsi dal 10% al 90% della variazione prevista

D. il tempo necessario all'ingresso per portarsi dallo 0% al 50% della variazione prevista

20. Un half adder ha questo nome perchè:

A. fornisce il risultato della somma diviso per due

B. effettua la somma tra due bit e ne fornisce anche il riporto

C. effettua la somma tra due bit, fornisce anche il riporto ma non tiene conto dell'eventuale precedente riporto

D. effettua la somma solo tra due bit

21. La serie TTL 74 è la serie [1] _____ e la TTL 54 è la serie [2] _____ .

22. la corrente di sink [1] _____ nell'uscita e [2] _____ dall'entrata.

23. Si definisce [1] _____ la massima tensione di [2] _____ sovrapponibile al segnale in uscita ad una porta senza che [3] _____ della successiva lavori con un [4] _____ logico non corretto.

24. Si definisce [1] _____ il numero massimo di [2] _____ collegabili ad una singola [3] _____ .

25. Le operazioni che si possono effettuare tra le variabili logiche sono [1] _____: la [2] _____ o NOT, la somma logica o [3] _____ e il prodotto logico o [4] _____ .

26. I dispositivi realizzati con tecnologia bipolare sono [1] _____ e possono erogare [2] _____ in uscita, quelli realizzati con tecnologia unipolare necessitano di [3] _____ minori e realizzano [4] _____ molto maggiori.

CIRCUITI IN MEDIA SCALA DI INTEGRAZIONE

Tra i circuiti in media scala di integrazione (MSI) si considerano il multiplexer, il demultiplexer, l'encoder, il decoder, il display.

1.13 MULTIPLEXER

Il multiplexer è un dispositivo che permette di selezionare in uscita uno degli n segnali digitali presenti agli n ingressi. Per poter effettuare questa operazione occorre selezionare l'ingresso che, di volta in volta si ritiene di dover trasferire in uscita. Questo viene fatto mediante **ingressi di selezione**.

Il multiplexer infatti funziona come un commutatore digitale in cui viene impostata la posizione di commutazione attraverso un codice digitale.

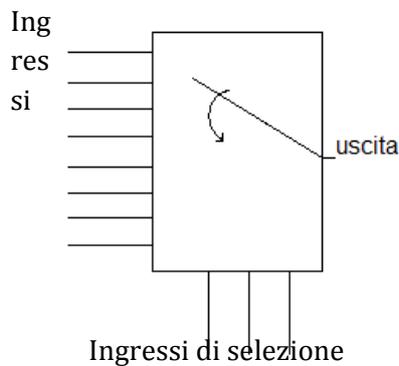
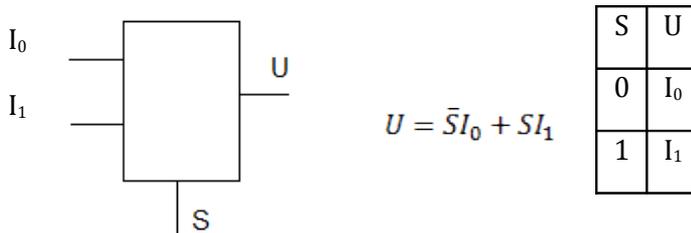


Fig. 19 Multiplexer

1.13.1 MULTIPLEXER A 2 INGRESSI

Un multiplexer a 2 ingressi necessita di un solo ingresso di selezione. Se l'ingresso di selezione S è a 0 si trasferisce in uscita l'ingresso I_0 , se l'ingresso di selezione S è a 1 si trasferisce in uscita l'ingresso I_1 .



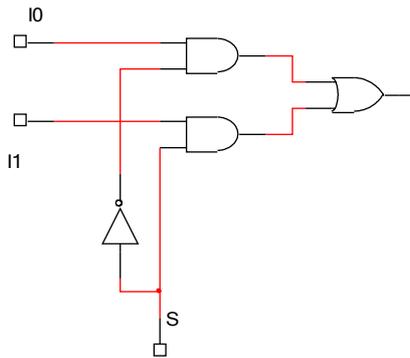
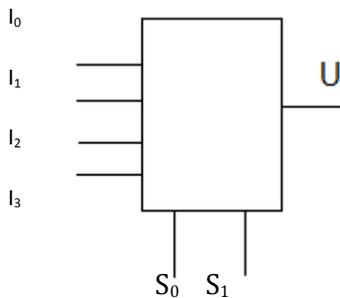


Fig. 20 Schema logico multiplexer a 2 ingressi

1.13.2 MULTIPLEXER A 4 INGRESSI

In un MUX a 4 ingressi sono necessari 2 ingressi di selezione; infatti il numero degli ingressi è uguale al numero ottenuto elevando la base 2 al numero degli ingressi di selezione $I_n = 2^{S_n}$ (nel nostro caso $4=2^2$).



S1	S0	U
0	0	I ₀
0	1	I ₁
1	0	I ₂
1	1	I ₃

$$U = \bar{S}_0\bar{S}_1I_0 + \bar{S}_0S_1I_1 + S_0\bar{S}_1I_2 + S_0S_1I_3$$

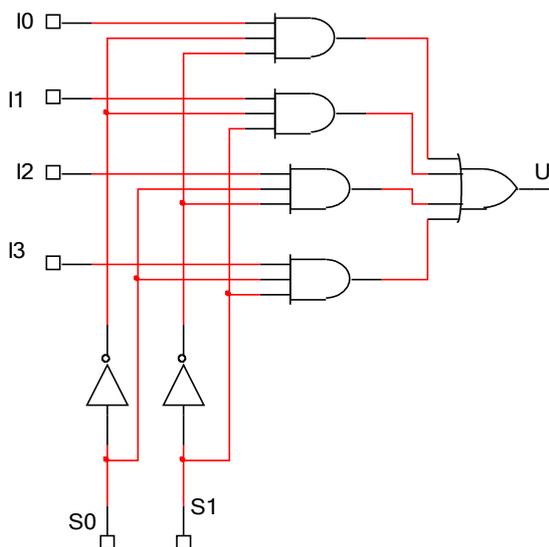


Fig. 21 Schema logico multiplexer a 4 ingressi

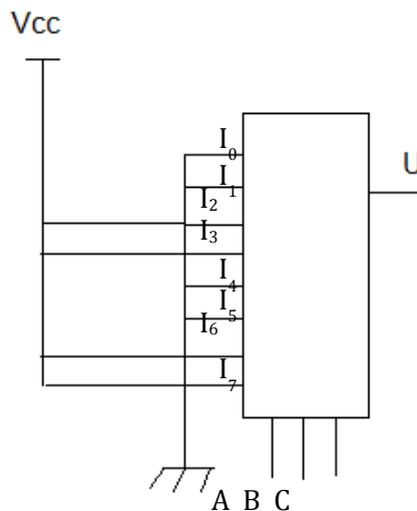
1.13.3 USI DEL MUX

Guardando lo schema di un Mux si nota immediatamente come il dispositivo presenti tante linee in ingresso e una sola linea in uscita quindi il sistema si presenta utile nel trasferimento di informazioni da parallelo a serie.

Tuttavia non è il solo utilizzo , infatti viene usato anche per implementare ²funzioni logiche . Data una funzione logica , se si fa corrispondere ad ogni ingresso dati del MUX uno dei mintermini ricavati dalla tabella di verità e agli ingressi di selezione le variabili logiche relative , a seconda della combinazione delle variabili logiche, si porta in uscita l'ingresso cui corrisponde il mintermine ottenuto dalla combinazione delle variabili.

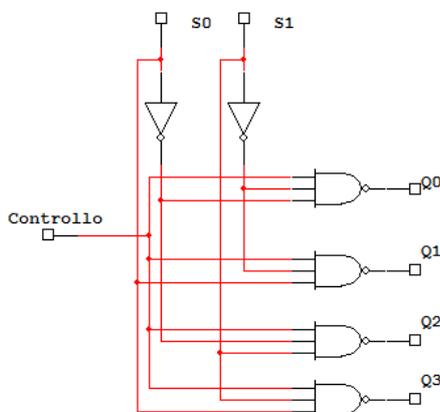
ES.: La funzione $Y = \bar{A}\bar{B}\bar{C} + \bar{A}BC + A\bar{B}\bar{C} + ABC$ dà luogo alla tabella sottoriportata

A	B	C	Y	Ingressi del MUX
0	0	0	0	I ₀
0	0	1	0	I ₁
0	1	0	1	I ₂
0	1	1	1	I ₃
1	0	0	0	I ₄
1	0	1	0	I ₅
1	1	0	1	I ₆
1	1	1	1	I ₇



1.14 DEMULTIPLEXER

Il demultiplexer è un dispositivo che svolge la funzione opposta al multiplexer: preleva il dato dall'unico ingresso dati e lo smista su una delle possibili uscite, che viene selezionata attraverso una linea di controllo.



ingressi		uscite			
S ₀	S ₁	Q ₀	Q ₁	Q ₂	Q ₃
0	0	Controllo	0	0	0
0	1	0	Controllo	0	0
1	0	0	0	Controllo	0
1	1	0	0	0	Controllo

Fig. 22 Demultiplexer da 2 a 4

² Implementare = realizzare, costruire.

1.15 DECODER

Il decoder è un dispositivo costituito da n ingressi e 2^n uscite che vengono attivate una alla volta solo in corrispondenza di una configurazione degli ingressi.

Spesso il decoder viene usato per realizzare un demultiplexer.

ingressi		uscite			
I_0	I_1	Q_0	Q_1	Q_2	Q_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

Tabella 6 Tabella di verità

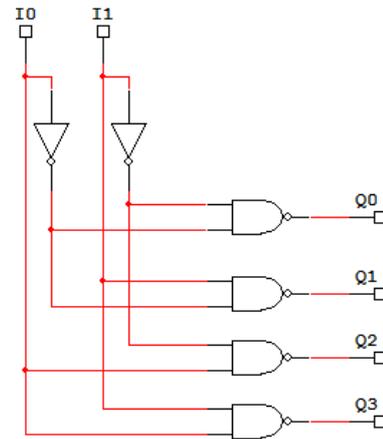


Fig. 23 decoder a 2 ingressi e 4 uscite

Si può notare come due o più uscite non possono essere contemporaneamente alte e che ad una uscita alta corrisponde una precisa configurazione degli ingressi.

1.16 ENCODER

L'encoder esegue la funzione inversa di quella del decoder. E' un dispositivo dotato di 2^n ingressi ed n uscite. In corrispondenza di ciascuna linea attiva di ingresso si ha in uscita una configurazione di bit secondo un opportuno codice. In questo dispositivo non si deve mai verificare che siano attivi due o più ingressi; se ciò dovesse verificarsi, il codice di uscita è errato.

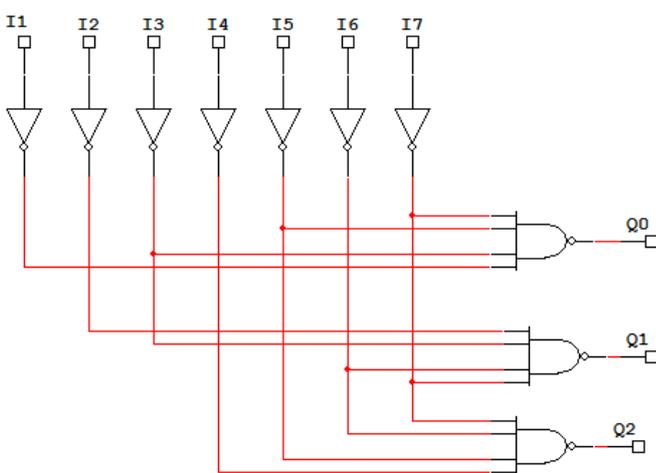


Fig. 24 Encoder a 8 ingressi e 3 uscite

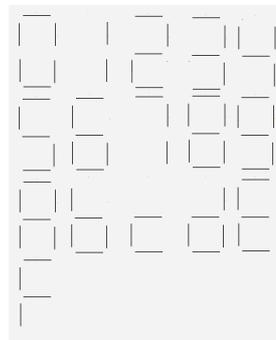
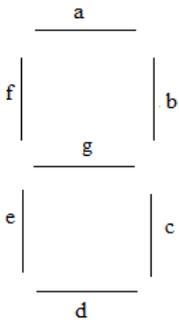
Quando tutti gli ingressi sono alti, le uscite sono basse. Se l'ingresso I_1 diventa alto, solo l'uscita Q_0 diventa alta; se dall'ingresso I_5 diventa alto, le uscite Q_0 e Q_2 diventano alte (101= corrispondente al numero 5 convertito in binario)

I **priority encoder** sono dispositivi che generano un codice corrispondente all'ingresso attivo con priorità più elevata, che viene assegnata all'ingresso con numero massimo e va gradualmente decrescendo con il numero associato a ciascun ingresso.

1.17 DISPLAY

Il display è un sistema di visualizzazione dei dati che utilizza la tecnologia LED o quella a cristalli liquidi LCD.

Un display a LED può essere a segmenti o a punti. Il più comune è il **display a 7 segmenti** ed è di tipo alfanumerico. E' costituito da 7 LED disposti nella seguente maniera



che permettono di combinare i simboli a fianco.

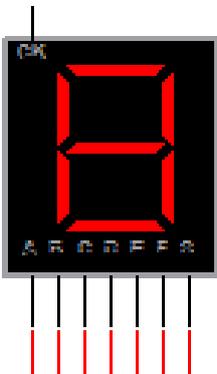


Fig. 25 display a 7 seg.

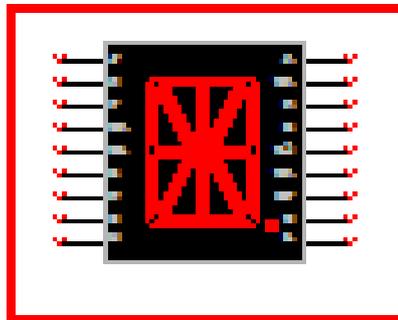


Fig. 26 display a 14 seg.

I led che costituiscono i singoli segmenti possono essere collegati tra loro in due modi possibili: a **catodo comune** e ad **anodo comune**.

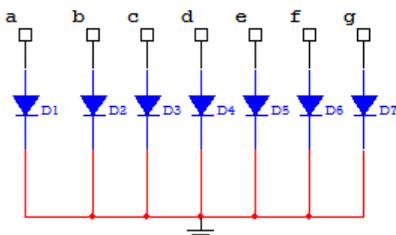


Fig. 27 Display a catodo comune

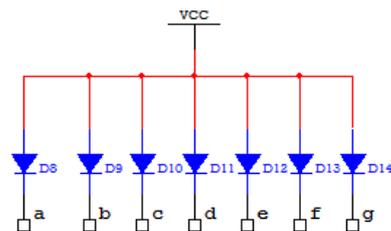


Fig. 28 Display ad anodo comune

Un display a LCD usa la tecnologia dei cristalli liquidi termotropici, che modificano il proprio comportamento, cioè la struttura cristallina, con il variare della temperatura. Se un fascio di luce incide su tali cristalli, riuscirà ad attraversarli solo se questi sono orientati (polarizzati) in una direzione opportuna, cosa che viene ottenuta mediante un campo elettrico.

Un display, per visualizzare informazioni provenienti da circuiti logici deve essere pilotato da un decoder.

LA LOGICA SEQUENZIALE

CIRCUITI DI MEMORIA O SEQUENZIALI

Un circuito si dice **sequenziale** se le sue uscite dipendono dallo stato degli ingressi nell'istante in cui si effettua la misura, ma anche dalla condizione in cui si trovavano le stesse uscite nell'istante precedente quello considerato.

I circuiti più semplici che realizzano tale condizione si chiamano **latch** (= chiavistello). Il significato del loro nome sta nel fatto che bloccano l'informazione che arriva agli ingressi e aprono il chiavistello solo in certe condizioni.

1.18 I LATCH

Il primo dispositivo realizzato è stato il **latch SR**. Esso può essere ad ingressi attivi bassi o alti.

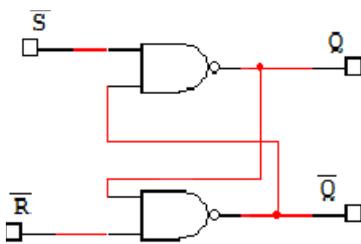


Fig. 29 latch ad ingressi attivi bassi e relativa tabella

\bar{S}	\bar{R}	Q_i	Q_{i+1}	$\overline{Q_{i+1}}$	Condizione dell'uscita
0	0	0	1	1	Non valida
		1	1	1	
0	1	0	1	0	set
		1	1	0	
1	0	0	0	1	Reset
		1	0	1	
1	1	0	0	1	memoria
		1	1	0	

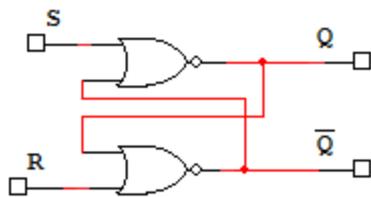
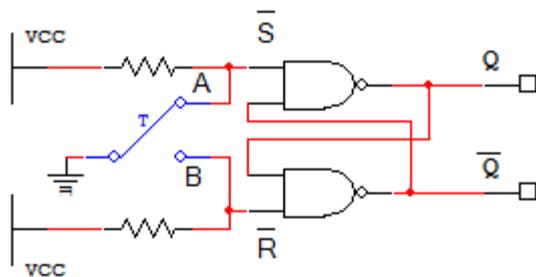


Fig. 30 latch ad ingressi attivi alti e relativa tabella

S	R	Q_i	Q_{i+1}	$\overline{Q_{i+1}}$	Condizione dell'uscita
0	0	0	1	1	Memoria
		1	1	0	
0	1	0	0	1	Reset
		1	0	1	
1	0	0	1	0	Set
		1	1	0	
1	1	0	1	1	Non valida
		1	1	1	

Uno degli usi più frequenti del latch è quello del **circuito antirimbalo** (*anti bounce*) quando la presenza di interruttori e deviatori meccanici comporta una rapida sequenza di rimbalzi indesiderati.

La rapida sequenza di commutazioni da un livello all'altro provoca un anomalo funzionamento dei circuiti digitali che possono interpretare in modo errato il dato in arrivo. Tutto ciò va evitato, appunto, con il seguente circuito.

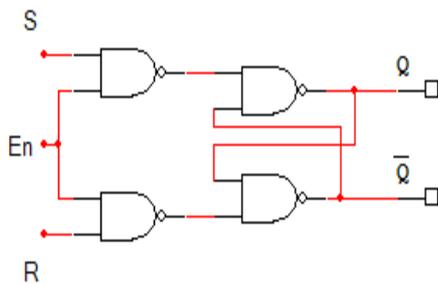


l'interruttore T può toccare A portandolo a zero, può toccare B, può essere in posizione intermedia tra i due, ma non può portare entrambi a zero contemporaneamente.

Il che equivale a eliminare la condizione non valida della tabella di figura 24.

Fig. 31 Circuito antibounce

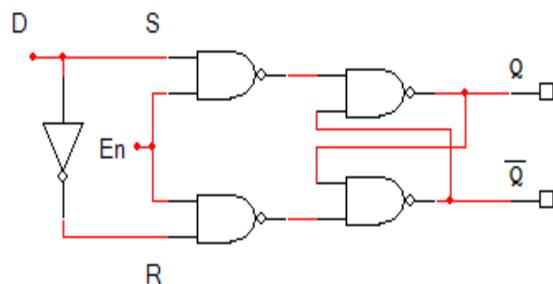
I circuiti di memoria (latch) possono essere dotati di un ulteriore ingresso(**enable**) che, se posto a livello basso, blocca il funzionamento alla condizione di memoria, se a livello alto, non influenza il comportamento del latch.



En	S	R	Q_{i+1}
0	X	X	Q_i
1	0	0	Q_i
1	0	1	0
1	1	0	1
1	1	1	Non valida

Fig. 32 latch SR con Enable e tabella relativa

Se si collegano insieme gli ingressi SR attraverso una porta NOT, come nella figura sotto riportata, si ottiene un latch D che fornisce uscita 1 se l'ingresso comune D è a 1, darà uscita zero se D=0



En	D	Q_{i+1}
0	X	Q_i
1	0	0
1	1	1

Fig. 33 latch D con Enable

1.19 I FLIP-FLOP

I latch privi di Enable sono sì elementi di memoria, ma sentono tutte le variazioni che subiscono i segnali in ingresso; se si usano i latch con enable si migliora la risposta ma anche in questo caso esiste un inconveniente: l'enable non può essere tenuto attivo per molto tempo. Anzi sarebbe preferibile che l'enable fosse attivo per tempi brevissimi, in modo da cogliere solo ciò che avviene in un determinato istante. Per questo motivo i circuiti sequenziali sono dotati di un ingresso di temporizzazione che si chiama **clock**.

Definizione 3.1. Si definisce FF (flip-flop) un elemento di memoria con ingresso di clock.

Il clock, a differenza dell'enable che agisce su tutto il livello, interviene solo in corrispondenza dei fronti.

Apriamo una parentesi per analizzare il segnale ad onda quadra.

Ecco come si presenta un segnale ad onda quadra ideale visualizzato all'oscilloscopio.

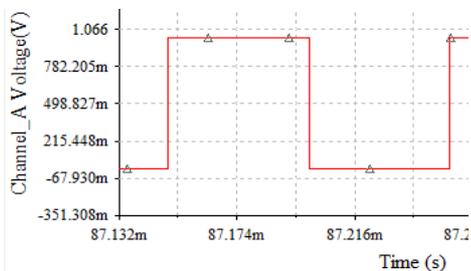


Fig. 34 Segnale ad onda quadra ideale

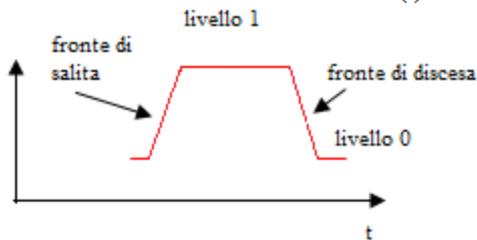


Fig. 35 segnale reale

Bisogna tener presente che il clock non deve avere tempi di discesa o di salita troppo lunghi perché ciò potrebbe pregiudicare il funzionamento corretto del FF.

Il flip-flop può essere attivo sul fronte di salita  o sul fronte di discesa 

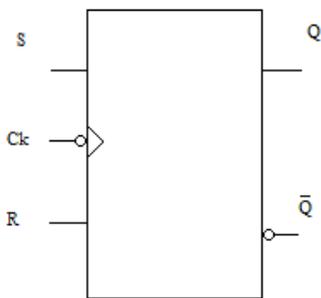
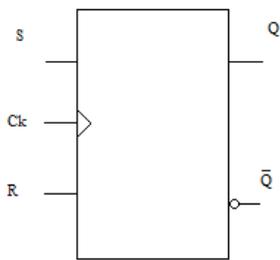


Fig. 36 FFSR con clock attivo sul fronte di discesa e relativa tabella

Ck	S	R	Q_{i+1}
0	X	X	Q_i
1	X	X	Q_i
	X	X	Q_i
	0	0	Q_i
	0	1	0
	1	0	1

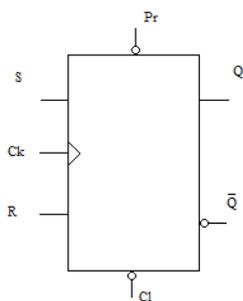


Ck	S	R	Q_{i+1}
0	X	X	Q_i
1	X	X	Q_i
	X	X	Q_i
	0	0	Q_i
	0	1	0
	1	0	1

Fig. 37 FFSR con clock attivo sul fronte di salita e relativa tabella

Gli ingressi SR vengono chiamati **sincroni** perché sono attivi solo quando il clock è attivo.

Talvolta è necessario fissare lo stato iniziale del FF; in questo caso è necessario fare uso di altri due ingressi detti **asincroni** che permettono di portare a 1 l'uscita Q (**preset**) o portare a 0 l'uscita Q (**clear**).



Sia il preset (Pr) che il clear (Cl) sono attivi bassi

Fig. 38 FFSR con ingressi asincroni

1.20 TIPOLOGIE DI FLIP-FLOP

Ck	J	K	Q_{i+1}
	X	X	Q_i
	0	0	Q_i
	0	1	0
	1	0	1
	1	1	\bar{Q}_i

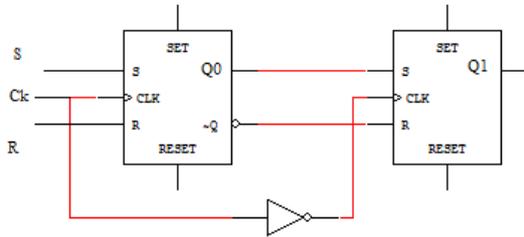
I Flip-flop, oltre che SR, possono essere anche di tipo **D**, che si ottengono, come nel caso dei latch collegando insieme l'ingresso S con quello R attraverso una porta NOT.

Poiché nel FFSR non può essere presente la condizione 11, perché fornirebbe una condizione di uscita non valida, il circuito è stato modificato in modo da superare il problema. La soluzione si chiama **FFJK** in cui la condizione 11 porta

l'uscita al complementare dell'uscita precedente (se l'uscita nell'istante precedente era a 1, viene portata a 0, se l'uscita era a 0 viene portata a 1). Questa condizione si chiama di **Toggle**.

Se nel FFJK si collegano insieme l'ingresso J con il K, si ottiene il **FFT** in cui, a seconda che l'ingresso comune sia a 0 o a 1 l'uscita commuta continuamente tra la precedente e la sua complementata.

Analizziamo adesso il seguente circuito:



Si tratta di due FF SR collegati tra di loro e su cui agisce lo stesso comando di clock.

Il clock agisce sul primo FF con il fronte di salita e sul secondo FF con il fronte di discesa.

Se si compila la tabella degli stati si ricava

Ck	S	R	Q_i	Q_{i+1}	
0	0	0	0	0	Memoria
			1	1	
0	0	1	0	0	Reset
			1	0	
1	0	0	0	1	Set
			1	1	
1	1	1	0	X	Non valida
			1	X	

La tabella è valida per entrambi i FF ma cambia il fronte attivo del clock. Considerando il sistema completo, esso funziona quindi sull'intero impulso di clock. **Si tratta di un FF master-slave.**

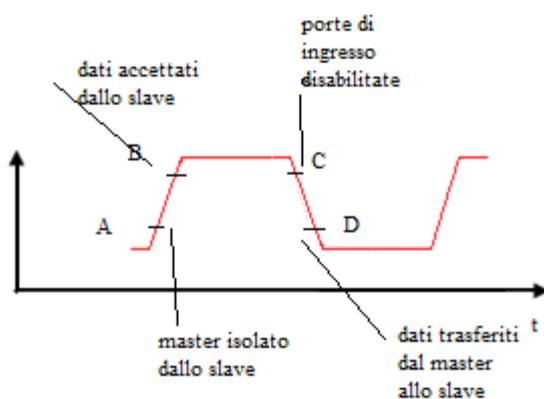
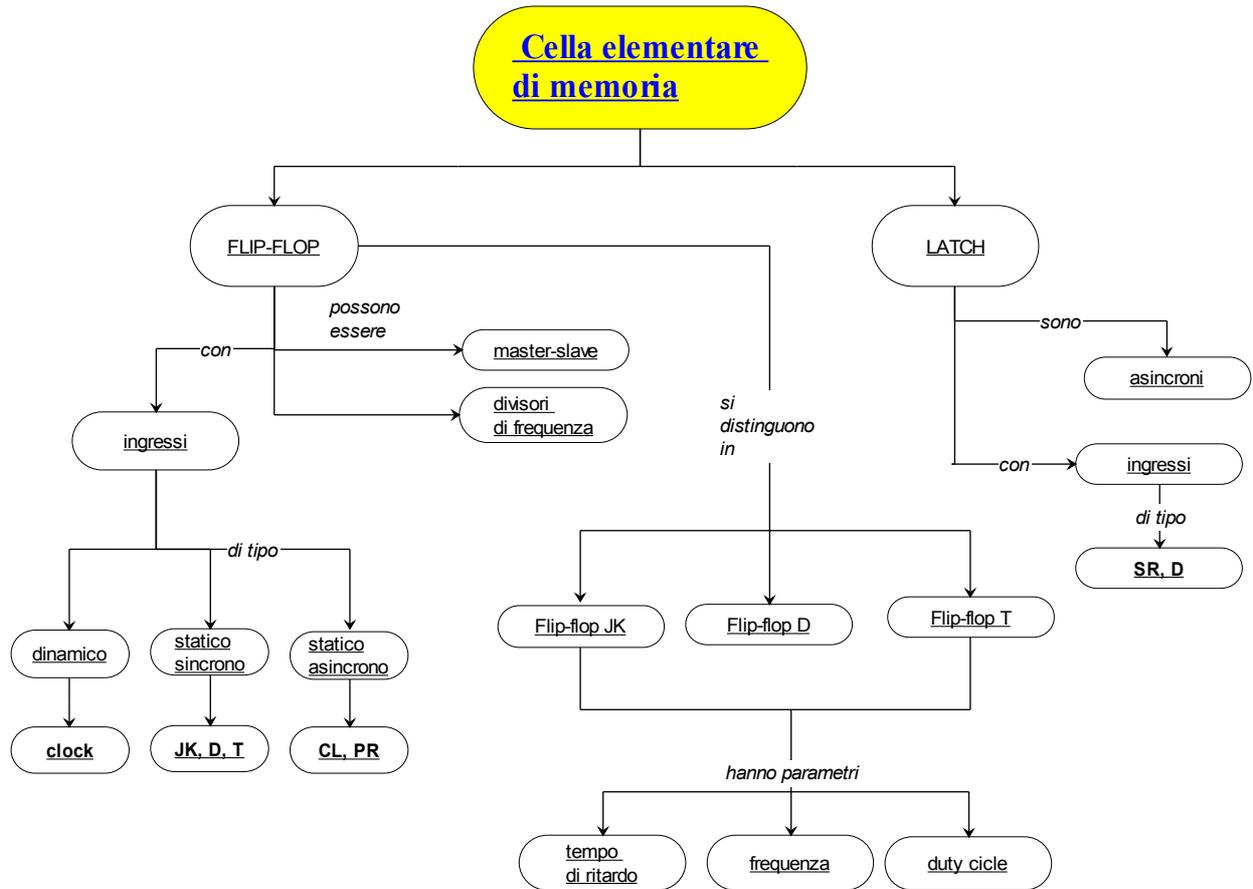


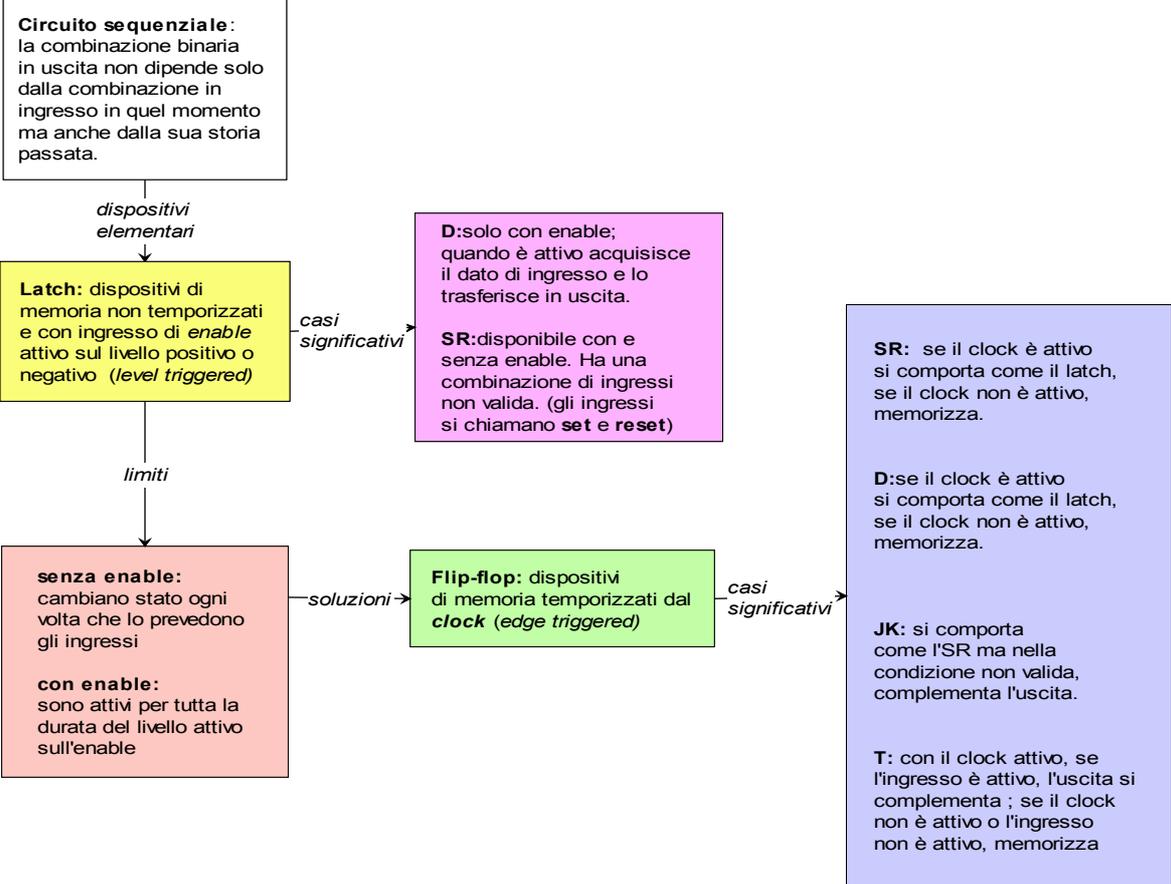
Fig. 39 livelli di sincronizzazione di un master-slave

SLIDE CU CIRCUITI APPLICATIVI DELLE CELLE DI MEMORIA

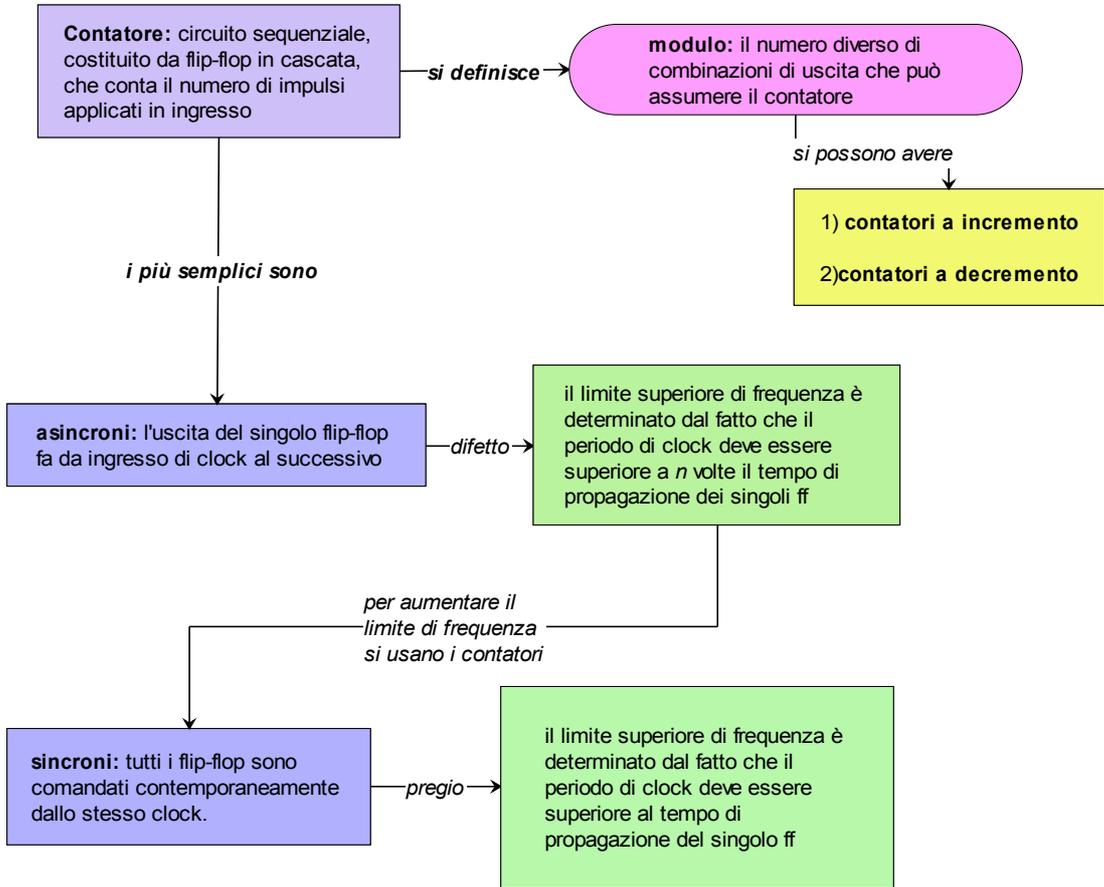


Slide 2

Latch e flip-flop



I contatori



I registri

Registri: sono dispositivi di memoria per parole a più bit realizzati con flip-flop

si raggruppano in quattro casi

- SISO
(Serial Input
Serial Output)
- SIPO
(Serial Input
Parallel Output)
- PISO
(Parallel Input
Serial Output)
- PIPO
(Parallel Input
Parallel Output)

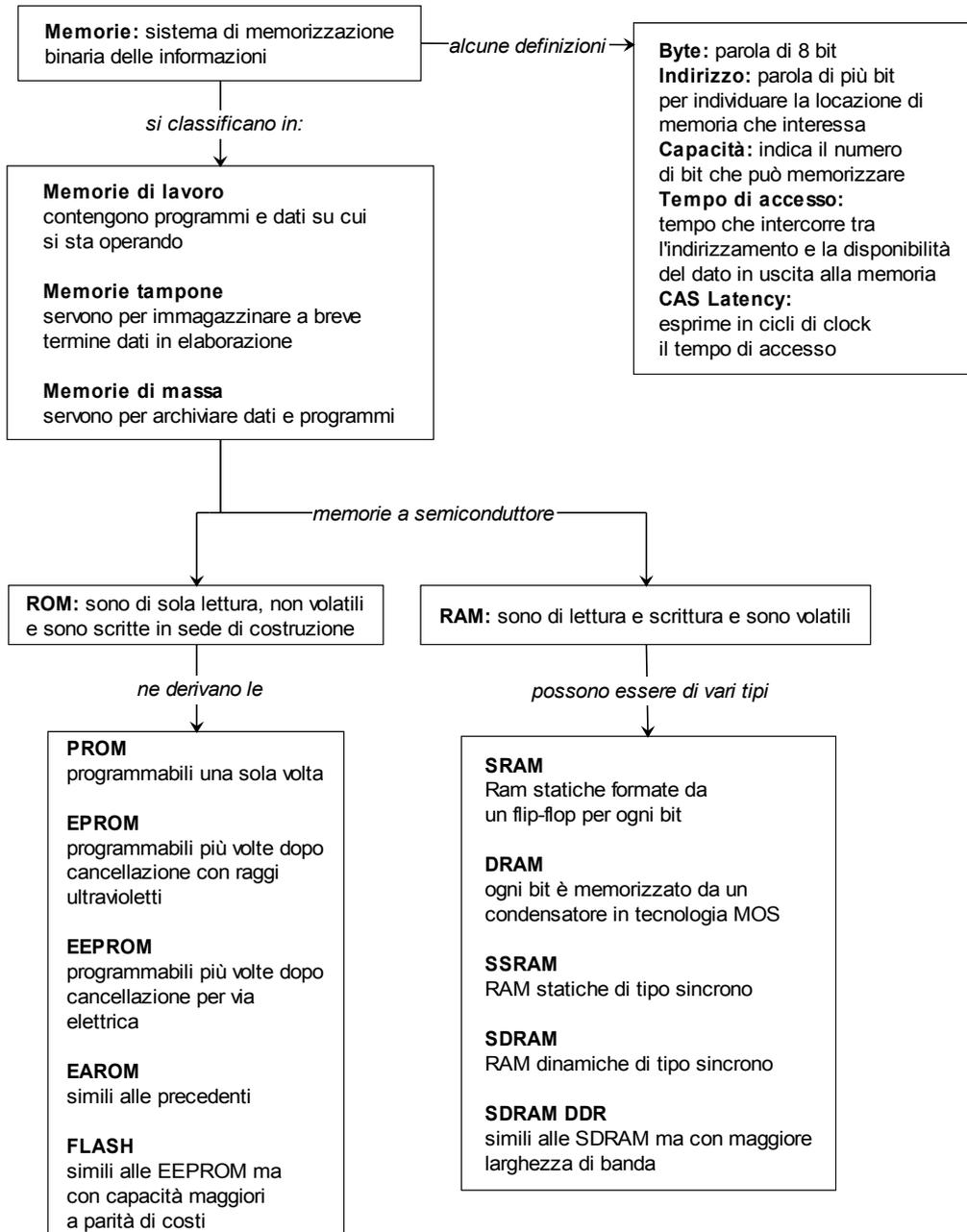
I singoli bit vengono acquisiti serialmente in modo sincrono e trasferiti serialmente in uscita sempre in modo sincrono. Ad ogni impulso di clock i bit si spostano di una posizione verso destra

I singoli bit vengono acquisiti serialmente in modo sincrono e trasferiti parallelamente in uscita in modo asincrono. Se i FF sono n, solo dopo n impulsi di clock una parola di n bit è presente in uscita

I singoli bit vengono acquisiti contemporaneamente in modo parallelo e asincrono e trasferiti serialmente in uscita in modo sincrono, un bit alla volta, a ogni impulso di clock

I singoli bit vengono acquisiti contemporaneamente in modo parallelo sincrono e contemporaneamente trasferiti in modo parallelo sincrono in uscita

Le memorie



Indice delle slide

Indice delle figure

FIG. 1 CIRCUITO LOGICO AND.....	6
FIG. 2 PORTA LOGICA AND.....	6
FIG. 3 CIRCUITO LOGICO OR.....	7
FIG. 4 PORTA LOGICA OR.....	7
FIG. 5 CIRCUITO LOGICO NOT.....	7
FIG. 6 PORTA LOGICA NOT.....	8
FIG. 7 PORTA LOGICA NAND.....	8
FIG. 8 PORTA LOGICA NOR.....	9
FIG. 9 PORTA LOGICA XOR.....	9
FIG. 10 PORTA LOGICA XOR.....	10
FIG. 11 GATING COMPLETO CON PORTE NAND.....	11
FIG. 12 CIRCUITO DI POLARIZZAZIONE DEL BJT.....	18
FIG. 13 CIRCUITO DI POLARIZZAZIONE DI UN MOS.....	19
FIG. 14 ESEMPIO DI CIRCUITO LOGICO.....	22
FIG. 15 RAPPRESENTAZIONE DEL MARGINE DI RUMORE CON IL GRAFICO DELLE TENSIONI MISURATO NEL PUNTO INDICATO.....	22
FIG. 16 CARATTERISTICA DI TRASFERIMENTO DI UNA NOT.....	22
FIG. 17 L'USCITA DI UNA PORTA A LIVELLO H EROGA CORRENTE ALL'INGRESSO SUCCESSIVO.....	23
FIG. 18 L'USCITA DI UNA PORTA A LIVELLO L ASSORBE CORRENTE DALL'INGRESSO SUCCESSIVO.2.....	23
FIG. 19 MULTIPLEXER.....	27
FIG. 20 SCHEMA LOGICO MULTIPLEXER A 2 INGRESSI.....	28
FIG. 21 SCHEMA LOGICO MULTIPLEXER A 4 INGRESSI.....	28
FIG. 22 DEMULTIPLEXER DA 2 A 4.....	29
FIG. 23 DECODER A 2 INGRESSI E 4 USCITE.....	30
FIG. 24 ENCODER A 8 INGRESSI E 3 USCITE.....	30
FIG. 25 DISPLAY A 7 SEG.	31
FIG. 26 DISPLAY A 14 SEG.	31
FIG. 27 DISPLAY A CATODO COMUNE.....	31
FIG. 28 DISPLAY AD ANODO COMUNE.....	31
FIG. 29 LATCH AD INGRESSI ATTIVI BASSI E RELATIVA TABELLA.....	33
FIG. 30 LATCH AD INGRESSI ATTIVI ALTI E RELATIVA TABELLA.....	33
FIG. 31 CIRCUITO ANTIBOUNCE.....	34
FIG. 32 LATCH SR CON ENABLE E TABELLA RELATIVA.....	34
FIG. 33 LATCH D CON ENABLE.....	34
FIG. 34 SEGNALE AD ONDA QUADRA IDEALE.....	35
FIG. 35 SEGNALE REALE.....	35
FIG. 36 FFSR CON CLOCK ATTIVO SUL FRONTE DI DISCESA E RELATIVA TABELLA.....	35
FIG. 37 FFSR CON CLOCK ATTIVO SUL FRONTE DI SALITA E RELATIVA TABELLA.....	36
FIG. 38 FFSR CON INGRESSI ASINCRONI.....	36
FIG. 39 LIVELLI DI SINCRONIZZAZIONE DI UN MASTER-SLAVE.....	37