

Raffaele Ilardo

**ELEMENTI DI
ELETTRONICA DIGITALE**

CAPITOLO 1

L'elettronica digitale è quella branca dell'elettronica che tratta di particolari circuiti progettati per assumere due soli valori di tensione, ciascuno avente un diverso significato logico.

Il nome *digitale* deriva dall'inglese "digit", che vuol dire cifra o numero; in effetti, tutto il funzionamento dei circuiti digitali si basa sul conteggio, a differenza dell'elettronica lineare, che si basa sulla variazione continua di grandezze fisiche, quali una tensione o una resistenza. L'elettronica digitale è alla base dello sviluppo dei computers che, nel bene e nel male, accompagnano la nostra vita quotidiana; si può oggi dire che praticamente tutte le nostre attività sono soggette, in una fase o nell'altra, ad un processo di digitalizzazione. Ma cosa s'intende, allora, per elaborazione digitale? Cercheremo di chiarirlo nei capitoli che seguono.

Nei circuiti digitali una qualsiasi tensione può avere solo due significati: sono considerate a livello basso, ovvero come "zero logico", tutte le tensioni di valore prossimo a zero, mentre sono considerate a livello alto, ovvero come "uno logico", tutte le tensioni prossime alla tensione di alimentazione: ad esempio, per un circuito digitale alimentato a 5 volt, una tensione di 0,2 volt e una tensione di 0,9 volt sono esattamente uguali: equivalgono entrambe ad uno **zero logico**; analogamente, una tensione di 4,2 volt e una di 4,9 volt sono entrambe equivalenti ad un **uno logico**.

Cercheremo adesso di chiarire la differenza sostanziale esistente fra valore analogico e valore digitale. Supponiamo di voler pesare un oggetto, usando due tipi diversi di bilance (figure a lato).

Nel primo caso pogeremo l'oggetto sul piatto di una bilancia a quadrante: a seconda del peso, la molla interna verrà più o meno compressa, mentre la lancetta si sposterà sul quadrante indicando il peso del nostro oggetto.

Nel secondo caso, useremo una bilancia di vecchio tipo, di quelle a due piatti: poseremo l'oggetto sul piatto di sinistra, quindi prenderemo dei pesi dalla cassetta e li pogeremo sull'altro piatto, fino a equilibrare il tutto.

Qual' è la differenza fra i due metodi? Nel primo caso, la lancetta può ruotare ed assumere sul quadrante qualsiasi posizione, in una gamma infinita di valori: una misurazione di questo tipo viene definita "**analogica**". Nel secondo caso, per equilibrare la bilancia, dobbiamo scegliere, tra i vari pesi, quelli giusti che permettano di eguagliare il peso poggiato sull'altro piatto. Osserviamo, però, che non abbiamo un numero infinito di possibilità; scegliendo i pesi giusti possiamo cercare di avvicinarci il più possibile al peso da misurare, e non è detto che si riesca a farlo in modo esatto.

Cosa abbiamo fatto in questo secondo caso? Abbiamo cercato di eguagliare un peso (cioè di avvicinarci il più possibile ad esso) usando dei pesi di riferimento, di valore già definito; ebbene, un tale modo di valutare una grandezza viene definito "**digitale**".

Naturalmente, se disponiamo di tantissimi pesi, di valore via via più piccolo, possiamo approssimare sempre meglio il peso da misurare, e commettiamo un errore più piccolo.

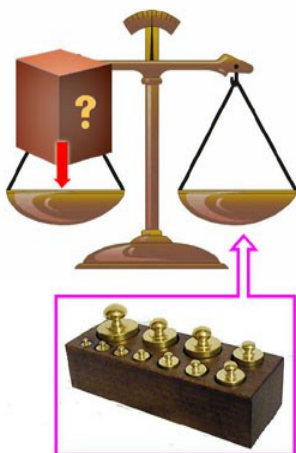
La tecnica di misurazione digitale, quindi, prevede che le grandezze misurate possano assumere soltanto un determinato numero di valori, corrispondenti alle combinazioni possibili dei valori utilizzati come campioni.

Ciò è diverso da quanto avviene in natura, dove i fenomeni variano in modo continuo e la loro consistenza può assumere una serie infinita di valori; d'altra parte, il fatto di limitare i possibili valori ad un numero finito, offre notevoli vantaggi e consente operazioni che altrimenti non sarebbero realizzabili.

La tecnica digitale è ormai presente in tutti i campi, persino in quelli che sembravano per loro natura i più lontani: la musica e le immagini.



Letture analogica



Letture digitale

La maggior parte della musica che ascoltiamo nelle nostre case proviene ormai da sorgenti digitali; ascoltando da un Compact Disk i nostri brani preferiti, mai penseremmo che il suono viene fuori da una successione interminabile di *uni* e di *zeri* registrati sulla superficie argentea del disco!

E chi oggi non possiede una macchina fotografica o una videocamera digitale? Proprio in tali settori la nuova tecnologia ha consentito di ottenere risultati sorprendenti, semplificando la costruzione meccanica degli apparecchi: non più pellicole o cassette magnetiche, ma solo memorie in grado di immagazzinare dati in quantità sempre maggiore.

Una differenza fondamentale fra l'elettronica digitale e quella, per così dire, "classica", sta nel fatto che cambiano i criteri di progettazione. Per realizzare un circuito a funzionamento lineare occorre tenere presente l'aspetto elettrico; ogni componente deve funzionare con le giuste tensioni, si devono introdurre resistenze e condensatori di valore adeguato, ogni stadio di amplificazione deve essere correttamente polarizzato, ecc. ecc.. Nell'elettronica digitale questo problema non esiste: ogni elemento logico, che sia una porta, un contatore o altro, è già di per se funzionante; quello che occorre è saperli disporre secondo schemi funzionalmente corretti.

Esamineremo adesso alcune applicazioni dell'elettronica digitale, cominciando dalle tecniche di calcolo con sistema binario, che sono alla base del funzionamento dei calcolatori.

Anche se la trattazione dei sistemi numerici esula dallo scopo di questo manuale, un sia pur breve accenno è indispensabile per comprendere i meccanismi di base.

Il sistema binario

Noi tutti indichiamo la quantità delle cose con i numeri: diciamo "ho 26 anni", "ho speso 340 euro", e così via. Per comporre i nostri numeri usiamo sempre e solo 10 cifre, ovvero lo zero, l'uno, il due, il tre ecc., fino al nove. Siamo talmente abituati a questo modo di contare che non ci sembra ne possano esistere altri. Invece il nostro modo di contare è solo uno dei tanti, e si basa su quello che viene chiamato "sistema decimale", proprio perché utilizza 10 cifre.

Potremmo però usare un numero maggiore di cifre o segni, per esempio sedici (un tale sistema sarebbe detto "esadecimale"), oppure usarne di meno, addirittura solo due. Il sistema basato su due sole cifre, che sono lo zero e l'uno, viene detto "binario", ed è quello che permette il funzionamento di tutti i calcolatori, dai primi prototipi fini ai potentissimi elaboratori attuali.

In tutti i sistemi numerici, il valore di ogni cifra è dato dalla sua posizione.

Nel sistema decimale, per esempio, il valore delle cifre corrisponde alle potenze di 10: leggendole sempre da destra verso sinistra, la prima rappresenta le unità, perché vale 10^0 (cioè 10 elevato a zero, ovvero 1), la seconda vale 10^1 (cioè 10) e quindi rappresenta le decine, la terza vale 10^2 e quindi rappresenta le centinaia, e così via; il numero 127 significa quindi che la quantità da esprimere è formata da 7 unità, aggiunte a 2 decine ed aggiunte ad 1 centinaia.

Nel sistema binario, invece, le cifre corrispondono alle potenze di 2; sempre leggendo da destra verso sinistra, la prima cifra vale 2^0 e pertanto indica le unità, la seconda vale 2^1 e pertanto indica le *duine*, la terza vale 2^2 ed indica i *quartetti*, la quarta vale 2^3 ed indica gli *ottetti*, ecc. In pratica questo vuol dire che ogni cifra raddoppia di valore (o di peso) spostandosi verso sinistra.

Il meccanismo risulterà più chiaro facendo un paragone tra sistema decimale e sistema binario; useremo un numero che, essendo formato solo da cifre uno e zero, può essere sia decimale che binario: 10110

Si vede, nella tabella che segue, che la stessa sequenza di cifre 10110 può assumere significati diversi, e cioè esprimere differenti quantità, in funzione del sistema numerico che si adotta.

Il valore del numero dipende quindi dal "peso" che assume ogni singola cifra in funzione della sua posizione e del sistema numerico:

- nel sistema decimale, ogni cifra assume un valore pari a 10 volte quella che la precede;
- nel sistema binario, ogni cifra assume un valore pari a 2 volte quella che la precede (naturalmente le cifre del numero vanno lette da destra verso sinistra)

Si capisce così come sia possibile, anche utilizzando solo una sequenza di zeri e uni, rappresentare qualunque valore, purché si disponga di un numero sufficiente di cifre.

VALORE DEL NUMERO 10110 NEI SISTEMI DECIMALE E BINARIO									
SISTEMA DECIMALE					SISTEMA BINARIO				
decine di migliaia	migliaia	centinaia	decine	unità	sedicetti	ottetti	quartetti	duetti	unità
1	0	1	1	0	1	0	1	1	0
10000	0	100	10	0	16	0	4	2	0
Valore totale: 10000+100+10 = 10110 (diecimilacentodieci)					Valore totale: 16+4+2 = 22 (ventidue)				

Il sistema binario, che sarebbe decisamente scomodo per noi umani, va però molto bene per i circuiti digitali che, come abbiamo detto, possono assumere solo due stati logici: lo zero e l'uno. Se consideriamo un solo circuito, il cui stato può cambiare fra livello "L" (cioè basso = 0) e livello "H" (cioè alto = 1), abbiamo poche possibilità: possiamo rappresentare solo 2 valori!

primo circuito	secondo circuito	valore decimale corrispondente
0	0	0
0	1	1
1	0	2
1	1	3

Con due circuiti si possono rappresentare 4 diversi valori

Se però usiamo due circuiti, le due possibilità del primo si combinano con le due possibilità del secondo, dando così origine a quattro possibili situazioni o valori (come si vede nella prima delle due tabelle a lato).

Se poi i circuiti sono tre (seconda tabella), le combinazioni possibili diventano $2 \times 2 \times 2$, ovvero otto, e si possono in tal modo rappresentare otto diversi valori (da zero a sette).

Si vede allora che il numero di valori rappresentabili dipende da quanti sono i circuiti utilizzati; per l'esattezza, diremo che il numero di combinazioni è uguale alla potenza di 2 avente per esponente il

numero di circuiti; infatti, nel caso dei tre circuiti, abbiamo:

$$2^3 = 8$$

primo circuito	secondo circuito	terzo circuito	Valore decimale corrispondente
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

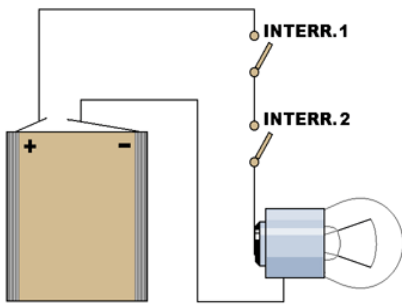
Con tre circuiti si possono rappresentare 8 diversi valori

Anche il calcolatore più complesso, è formato da un numero enorme di circuiti elementari; ognuno di questi circuiti elementari può essere considerato come una cella in grado di assumere uno dei due stati logici: zero o uno. Questa unità elementare di informazione viene denominata **bit**; quando si parla di dati, un'importante caratteristica è il numero di bit che li compongono.

Per *dato a 8 bit*, per esempio, si intende un valore rappresentato da otto circuiti od unità elementari. In base alla formula che abbiamo visto prima, otto cifre in binario possono rappresentare in tutto 256 diversi valori. Se per determinate applicazioni una gamma di 256 valori non risulta sufficiente, occorre rappresentare i dati servendosi di un maggior

numero di bit. Ad esempio, usando 16 bit, si possono rappresentare 65536 diversi valori, che sono già una quantità notevole.

Tanto per passare ad un'applicazione pratica, parliamo delle immagini fotografiche: per fornire una rappresentazione accettabile dei vari colori, si usano abitualmente dati a 24 bit, il che vuol dire 16.777.216 colori. Questo valore corrisponde alla combinazione di diverse quantità di rosso, verde e blu; per l'esattezza si parte da 256 possibili gradazioni di rosso, che si combinano con altrettante gradazioni di verde ed ancora con altrettante gradazioni di blu. Non sono poche, anche se in realtà un occhio esperto e sensibile è in grado di apprezzare un maggior numero di sfumature!

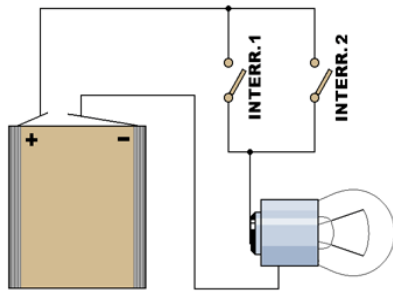


Purtroppo questa volta occorre un po' di pazienza; prima di poter passare a qualche applicazione pratica è necessario trattare ancora un argomento: le porte logiche.

In questo capitolo abbiamo cominciato parlando di sistemi numerici, ovvero di matematica; adesso affronteremo, senza troppo farci coinvolgere, quella che viene chiamata "Algebra di Boole". In realtà si tratta di concetti logici piuttosto semplici, acquisibili facilmente con un minimo di ragionamento. Esamineremo due casi: la condizione **AND** e la condizione **OR**.

Riallacciamoci per un attimo ai circuiti elettrici e immaginiamo di voler accendere la lampadina della figura 9-1. Poiché ci sono due interruttori uno appresso all'altro, la corrente attraverserà prima uno

e poi l'altro: per accendere la lampadina occorre chiudere entrambi gli interruttori; occorre, cioè, chiudere



La presenza di quella **e** identifica la condizione AND (la parola inglese AND, infatti, equivale alla nostra congiunzione "e"). La condizione AND esprime quindi la necessità della concomitanza di più azioni per ottenere un risultato.

Consideriamo ora il caso della figura 9-2: ci sono due interruttori in parallelo, ovvero affiancati, che permettono di far arrivare la corrente alla lampadina; è possibile chiudere indifferentemente uno o l'altro degli interruttori: la corrente, potendo passare da una delle due strade, passerà dove trova l'interruttore chiuso e la lampadina si accenderà. Per accendere la lampadina occorre quindi chiudere



La presenza della **o** identifica la condizione OR (la parola inglese OR corrisponde alla nostra congiunzione "o" ed esprime la possibilità di un'alternativa: per ottenere un risultato è possibile compiere un'azione oppure un'altra)

Queste implicazioni logiche si ritrovano pari pari nel *funzionamento delle porte logiche* usate nei circuiti digitali. Una porta logica è un semplice circuito elettronico realizzato con un certo numero di transistori, diodi, ecc; ma quello che a noi interessa, non è sapere come è costruita una porta logica, bensì come si comporta e, quindi, come può essere utilizzata.

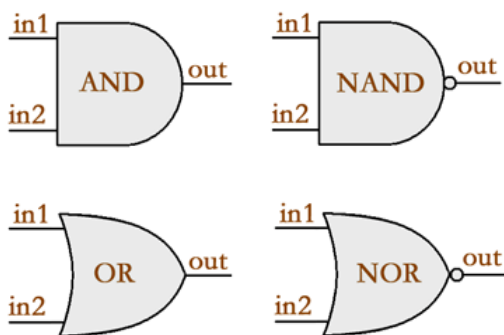


figura 9-3

Le porte logiche vengono indicate negli schemi con appositi simboli; in figura 9-3 vediamo quattro tipi di porte: la AND, la NAND, la OR e la NOR. In ogni caso, una porta possiede due o più ingressi (IN 1, IN 2, ...) ed un'uscita (OUT). Coerentemente a quanto detto fin qui per l'elettronica digitale, le porte logiche non funzionano con tensioni qualsiasi, ma solo con due livelli di tensione esattamente definiti: il livello basso (indicato anche con **L**, dall'inglese LOW) ed il livello alto (indicato anche con **H**, dall'inglese HIGH). Il livello presente sull'uscita è alto o basso in funzione dei livelli che sono presenti sugli ingressi.

Nel caso di una porta AND avente due ingressi, in accordo con i criteri logici visti prima, l'uscita sarà a livello alto solo quando entrambi gli ingressi sono a livello alto.

Il funzionamento di una qualsiasi porta è rappresentato in modo schematico in una tabella, che viene detta "tavola della verità" o, in inglese, "truth table"

La nostra porta AND sarebbe rappresentata da una tabella come quella seguente:

ingresso 1	Ingresso 2	Uscita
L	L	L
L	H	L
H	L	L
H	H	H

La tabella è composta da tre colonne; nella prima e nella seconda sono indicati i livelli presenti sui due ingressi, mentre nella terza colonna si trova il livello corrispondente in uscita. Si nota come l'uscita sia a livello H solo nell'ultima riga, ovvero solo quando entrambi gli ingressi sono a livello H.

Nel caso di una porta OR, sempre a due ingressi, la tavola della verità sarà quella che segue:

Ingresso 1	Ingresso 2	Uscita
L	L	L
L	H	H
H	L	H
H	H	H

Vediamo cioè che se l'uno o l'altro degli ingressi è a livello H (alto), l'uscita si trova a livello H (alto).

C'è un caso particolare che va un attimo considerato: abbiamo detto che la porta OR rappresenta la possibilità di alternativa, cioè di scegliere fra l'uno e l'altro; e nel caso che entrambi gli ingressi siano a livello alto? In effetti questa rappresenta una situazione particolare, che può originare qualche perplessità; proprio in relazione a questa eventualità, esistono due tipi di porte OR: quelle esclusive e quelle inclusive.

Quando sono alti tutti e due gli ingressi:

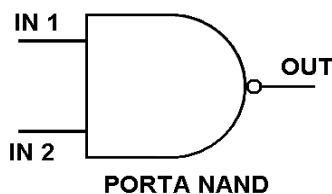
- se la porta è del tipo OR inclusivo l'uscita è anch'essa alta
- se la porta è del tipo OR esclusivo, l'uscita è a livello basso.

Ci sono altri tipi di porte logiche (porte AND a 4 o ad 8 ingressi, porte NAND, porte NOR ed altre ancora), ma il loro funzionamento è riconducibile ai due modelli base che abbiamo visto; osserviamo inoltre che è possibile ricostruire il funzionamento di una qualsiasi porta abbinando due o più porte semplici di diverso tipo. L'invertitore (inverter in inglese) è una porta che si limita ad invertire lo stato presente al suo ingresso: se l'entrata è a livello basso, l'uscita è a livello alto, e viceversa; inviando, per esempio, l'uscita di una porta AND ad un invertitore, si realizza il funzionamento di una porta NAND. Ma vedremo meglio questi meccanismi durante la loro applicazione pratica.

Tutte queste porte sono contenute all'interno dei circuiti integrati logici; utilizzando questi integrati in modo da combinare il funzionamento di diverse porte, è possibile realizzare circuiti in grado di svolgere funzioni logiche e di calcolo complesse quanto si vuole. Dal punto di vista dei procedimenti tecnologici di realizzazione, i circuiti logici si dividono principalmente in due grandi famiglie: i TTL ed i CMOS.

ingresso 1	Ingresso 2	Uscita
L	L	H
L	H	H
H	L	H
H	H	L

Figura 9-3b



La prima, il cui nome sta per "Transistor-Transistor Logic", utilizza i classici transistor bipolari; richiede una tensione di alimentazione molto precisa, esattamente di 5 volt, e presenta un consumo di corrente abbastanza elevato (circa 3 o 4 mA per ogni porta).

La seconda, il cui nome sta per "Complementary Metal Oxide Semiconductor", è invece realizzata con dei semiconduttori chiamati "Mosfet", utilizzati in configurazione complementare; funziona con qualsiasi tensione di alimentazione compresa tra 3 e 18 V, ed è caratterizzata da un consumo di corrente notevolmente minore (meno di 1 mA per ogni porta).

Ciascuna delle due famiglie presenta vantaggi e svantaggi; noi useremo in ogni caso gli integrati del tipo CMOS, che, come abbiamo visto, possono funzionare con diverse tensioni di alimentazione ed assorbono poca corrente.

Esaminiamo adesso uno dei circuiti integrati più semplici: il CD4011, che contiene al suo interno quattro porte NAND a due ingressi.

La porta NAND è esattamente uguale alla porta AND, tranne che i livelli sono invertiti (cioè quando l'uscita di una porta AND sarebbe H, l'uscita della porta NAND è L, e viceversa). Vediamo in figura 9-3b la tavola della verità di una porta NAND, e il suo simbolo:

Il simbolo della porta NAND differisce da quello della porta AND per la presenza di un pallino sull'uscita; quel pallino sta appunto ad indicare una inversione, ovvero che i livelli di uscita sono invertiti rispetto al funzionamento della porta AND.

Per meglio comprendere come con l'uso delle porte logiche sia possibile effettuare dei calcoli, realizzeremo un semplice circuito sommatore; precisiamo che tale circuito non ha alcuna utilità pratica, se non quella di servire a scopo didattico, come introduzione alle tecniche dei calcolatori.

Usando pochissimi componenti, cercheremo di ottenere una struttura logica in grado di realizzare i meccanismi necessari per sommare fra di loro due bit. Come si è visto in precedenza, il bit costituisce l'unità elementare di informazione nel mondo dei calcolatori.

Un bit può avere solo due valori: uno o zero, a seconda che la tensione che lo rappresenta sia a livello alto o basso.

Volendo sommare fra loro due bit, si possono presentare i seguenti casi:

- 1- i due bit sono entrambi a zero
- 2- uno dei due è a livello uno
- 3- entrambi i bit sono a livello 1

Nel primo caso non ci sono dubbi: il risultato deve essere un bit uguale a zero.

Nel secondo caso, poiché $1+0$ fa 1, il risultato sarà rappresentato da un bit uguale a uno.

Nel terzo caso, poiché $1+1$ fa 2, e nel sistema binario la cifra 2 non esiste, dovremo esprimere il risultato aggiungendo un bit a sinistra del primo, vale a dire un bit di peso doppio.

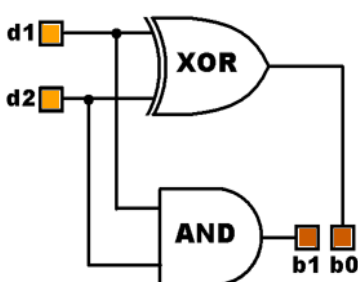
E' bene ricordare che il valore di una cifra dipende dalla sua posizione nell'ambito del numero; nei numeri binari le cifre hanno i seguenti valori:

512	256	128	64	32	16	8	4	2	1
-----	-----	-----	----	----	----	---	---	---	---

- il primo bit a destra, che viene detto "bit meno significativo", o LSB (dall'inglese *Last Significant Bit*) corrisponde a 2^0 e vale 1
- il secondo da destra corrisponde a 2^1 e vale 2
- il terzo da destra corrisponde a 2^2 e vale 4
- il quarto da destra corrisponde a 2^3 e vale 8
- il quinto da destra corrisponde a 2^4 e vale 16
- ... e così via...

Pertanto, quando sommiamo due bit che valgono 1, il risultato, che è 2, sarà espresso da due cifre: esattamente 1 0

dove lo zero indica zero unità, e l'uno indica la presenza del valore 2.



In figura 9-4 vediamo il circuito logico del sommatore. I due bit da sommare sono presenti ai due ingressi d1 e d2, mentre il risultato appare sulle uscite b1 e b0. Il circuito utilizza solo due porte: una porta XOR e una porta AND. Come detto in precedenza, la porta XOR realizza la funzione di OR esclusivo, vale a dire che fornisce 1 in uscita quando uno degli ingressi è ad 1 uno, e soltanto in quel caso; se entrambi gli ingressi sono a 1, l'uscita va a 0. La porta XOR realizza quindi la somma dei due dati presenti in ingresso, facendo apparire il risultato su b0; nel caso che d1 e d2 valgano entrambi 1, l'uscita va a zero, e ciò è giusto, perché in effetti la prima cifra del risultato (la meno significativa) deve essere zero. Sarà poi il meccanismo del riporto a fornire il giusto risultato. Al riporto (in inglese carry) provvede la porta AND; quando i due dati d1 e d2,

figura 9-4

Al riporto (in inglese carry) provvede la porta AND; quando i due dati d1 e d2,

che sono presenti anche all'ingresso della porta AND, sono entrambi uno, la porta fa scattare a livello alto (ovvero livello logico 1) la cifra più significativa (quella di sinistra).

Se volete divertirvi a costruire realmente questo circuito, qui di seguito trovate le indicazioni necessarie.

Occorre procurarsi due circuiti integrati:

- un MM74C86, contenente al suo interno 4 porte XOR a due ingressi (va bene anche l'integrato CD4070, che vi corrisponde come funzioni e come piedinatura)
- un MM74C08, contenente 4 porte AND a due ingressi; vanno bene anche i tipi HEF4081 (Philips) e MC14081 (Motorola)
- quattro LED di qualsiasi colore
- due transistor NPN in configurazione Darlington (vanno bene i seguenti tipi: BC517 - BC618 - MPSA14 o simili)
- due switch (commutatori per elettronica) di qualsiasi tipo
- quattro resistenze da 22 ohm
- due resistenze da 56 kohm

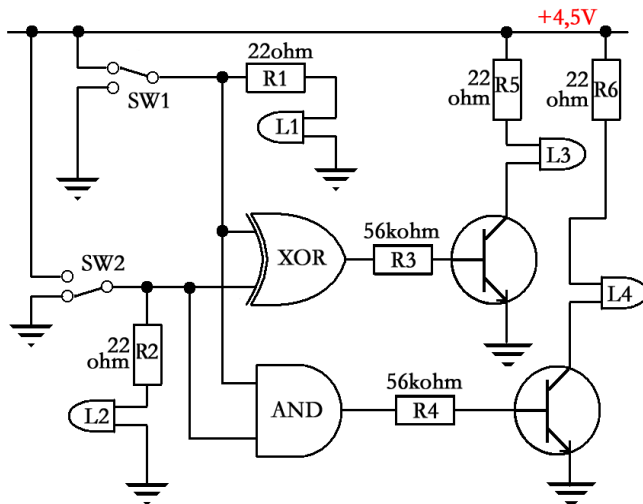


figura 9-5

Come si vede nella figura 9-5, sugli ingressi sono stati disposti due switch (ovvero commutatori) ed altrettanti LED.

Lo switch SW1 serve per simulare il primo dato: quando (come nel caso raffigurato) è collegato all'alimentazione positiva, il diodo led L1, alimentato attraverso la resistenza R1, è acceso, ad indicare che sull'ingresso è presente un 1; quando lo switch è collegato a massa, il diodo L1 è spento, indicando che sull'ingresso è presente uno 0.

Lo stesso discorso vale per lo switch SW2 (che appare collegato a massa), e serve ad impostare il secondo dato, il cui valore viene segnalato dal diodo L2.

I diodi led L3 ed L4 indicano lo stato logico delle uscite delle due porte, ovvero il risultato. Quando il led L3 è acceso, vuol dire che l'uscita della porta XOR è a livello alto, ovvero a 1;

quando L3 è spento, vuol dire che l'uscita è a zero.

Ricordiamo che L3 indica la cifra di destra, la meno significativa, mentre L4, collegato all'uscita della porta AND, indica il valore della cifra di sinistra, la più significativa.

DT1 e DT2 sono due transistor di tipo darlington; si tratta di transistor ad alto guadagno, ovvero in grado di controllare la corrente di collettore con una corrente di base debolissima, quale è quella che possono fornire in uscita le porte dei circuiti che abbiamo usato.

All'interno dei circuiti integrati, le porte sono collegate come risulta dalla figura che segue:

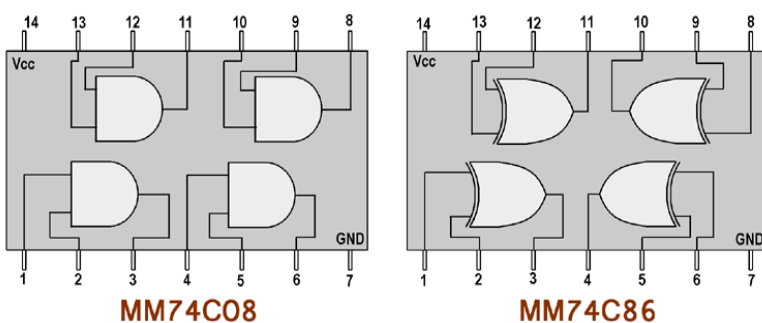


figura 9-7

Come si vede, ogni integrato contiene 4 porte uguali; per la nostra realizzazione noi ne useremo una sola, per esempio quella i cui ingressi corrispondono ai piedini 1 e 2 e l'uscita al piedino 3.

Nella pratica, il circuito può essere realizzato in tanti modi. Il più comune sarebbe quello di usare una basetta preforata di vetronite, i

cui fori corrispondono al passo dei piedini dei circuiti integrati; dal di sopra si inseriscono i componenti, e dal di sotto, ove ad ogni foro corrisponde un cerchio di rame, si effettuano le saldature (figure 9-8 e 9-9).

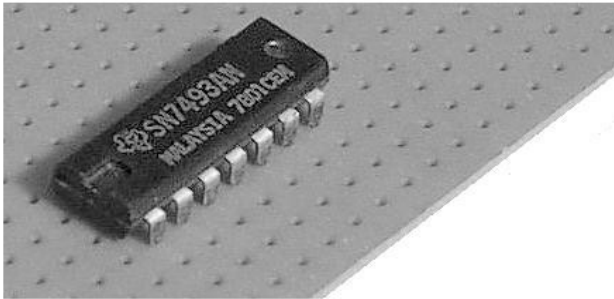


Figura 9-8

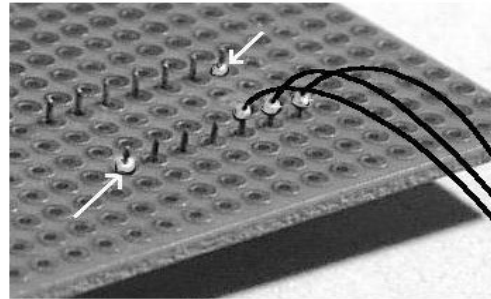


figura 9-9

Occorre stare attenti a non depositare quantità eccessive di stagno, per non rischiare di creare ponti metallici fra cerchietti di rame vicini, e a non insistere troppo col saldatore, perché i cerchietti di rame possono staccarsi. Un buon metodo è quello di fermare l'integrato alla bassetta con due soli punti di saldatura (come indicato dalle frecce); per gli altri collegamenti, i fili si salderanno sull'estremità dei piedini, senza toccare la bassetta.

Un secondo sistema, più grezzo, ma ugualmente valido, è quello di usare come supporto una tavoletta di legno e di incollarvi sopra, rovesciati, i due integrati (con i piedini per aria); gli stessi piedini serviranno come supporto per il fissaggio degli altri componenti. Altri punti di ancoraggio potranno ottenersi usando, come già spiegato nei capitoli precedenti, delle piccole viti di ottone inserite dove occorre.

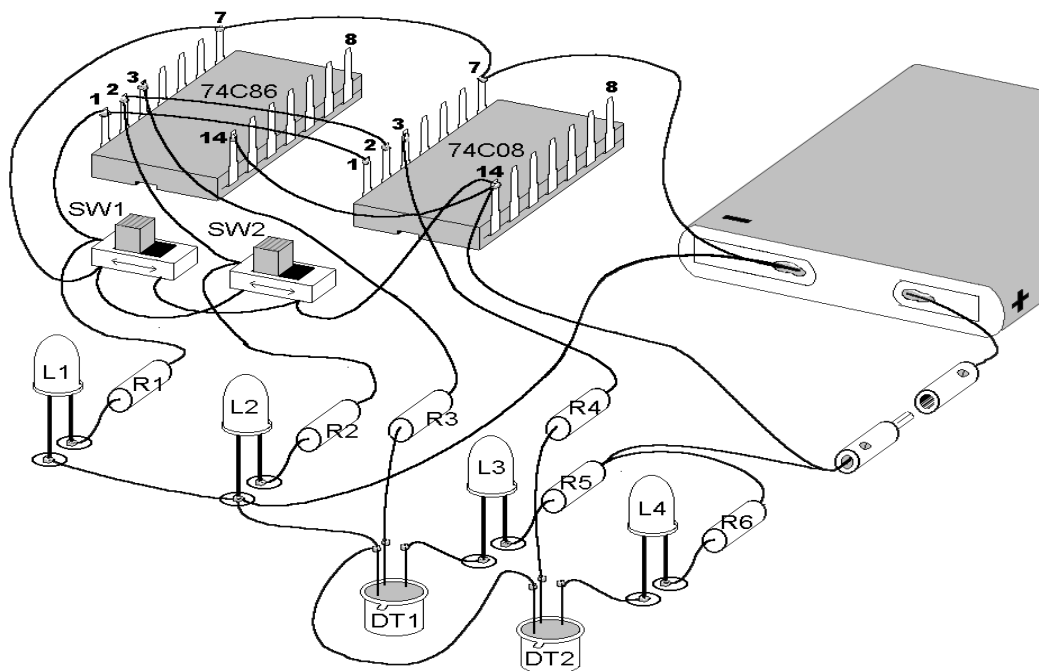


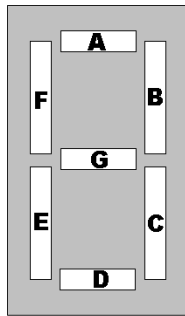
figura 9-10

In figura 9-10 si vedono i collegamenti da realizzare. Naturalmente è possibile disporre i vari pezzi in modo diverso, purché i collegamenti siano rispettati.

Gli switch SW1 e SW2 si collegano tramite tre contatti: in pratica c'è un contatto centrale che, a seconda della posizione della levetta, viene commutato da una parte o dall'altra; per il corretto funzionamento è importante individuare quale è il contatto centrale.

Display a sette segmenti e rappresentazione dei numeri

In molti casi è necessario che i valori calcolati dai circuiti siano visualizzati, ovvero comunicati in una forma che risulti leggibile da parte dell'utilizzatore umano. A tale scopo provvedono i cosiddetti "display", ovvero dei dispositivi in grado di mostrare cifre numeriche e lettere, tramite l'accensione di



- 0 = A+B+C+D+E+F**
- 1 = B+C**
- 2 = A+B+G+E+D**
- 3 = A+B+G+C+D**
- 4 = B+F+G+C**
- 5 = A+B+G+E+D**
- 6 = F+E+G+C+D**
- 7 = A+B+C**
- 8 = A+B+C+D+E+F+G**
- 9 = A+B+G+F+C**

Figura 9-13

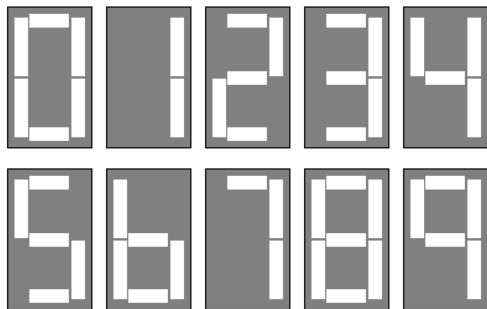
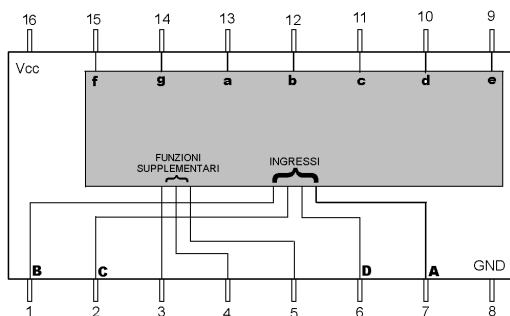


Figura 9-14

Le cifre da 0 a 9 rappresentate su un display a sette segmenti



DECODIFICATORE 74C48

Figura 9-15

segmenti luminosi. Ogni segmento è in pratica un diodo Led, di forma rettangolare, e può essere acceso o spento; i sette segmenti luminosi vengono combinati secondo lo schema indicato in figura 9-13. Ogni segmento risulta individuato da una lettera (da A a G):

- accendendo i segmenti a,b,c,d,e,f si ottiene la cifra 0
- accendendo i segmenti b e c si ottiene la cifra 1
- accendendo i segmenti a,b,g,e,d si ottiene il 2 e così via.

Ad accendere i segmenti giusti provvedono appositi circuiti integrati, chiamati "decoder", ovvero *decodificatori*.

Tali circuiti contengono una logica di decodifica che, in base al valore in ingresso, accende i segmenti del display che corrispondono alla cifra voluta.

Il dato deve essere presente in ingresso in forma binaria, o meglio in codice BCD (dall'inglese "Binary Coded Decimal", ovvero decimale codificato in binario). Abbiamo detto che il display può rappresentare le cifre da 0 a 9; poiché il numero più alto, ovvero il 9, in binario è rappresentato da 1001, vuol dire che saranno necessarie quattro entrate, ognuna delle quali potrà avere valore di 1 o di 0. Il circuito integrato decodificatore possiede infatti 4 entrate, denominate A, B, C e D, mentre le uscite sono in numero di sette, e vanno collegate nel giusto ordine a ciascuno dei sette segmenti del display.

In figura 9-15 vede lo schema del decodificatore MM74C48. L'ingresso dei dati si trova sui piedini 7-1-2-6, mentre le uscite da collegare ai sette segmenti del display sono sui piedini 13-12-11-10-9-15-14. I piedini 3-4-5 servono per funzioni supplementari che adesso non ci interessano.

In luogo dell'integrato MM74C48 può essere usato il CD4511, che è esattamente corrispondente.

Alle uscite del MM74C48 può essere collegato direttamente un display a catodo comune; vediamo questo cosa significa.

Se consideriamo i sette segmenti del display come sette diodi led separati, per il loro collegamento dovremmo disporre di 14 terminali, vale a dire 2 per ogni led. Possiamo però mettere insieme i terminali di tutti e sette i led da una sola parte, e lasciare l'altra

disponibile per il collegamento che dovrà accendere o meno il segmento. Le possibilità sono due: se mettiamo insieme tutti i terminali positivi, realizziamo un collegamento ad anodo comune; se mettiamo insieme tutti i terminali negativi, realizziamo un collegamento a catodo comune. La figura 9-16 illustra meglio questo concetto.

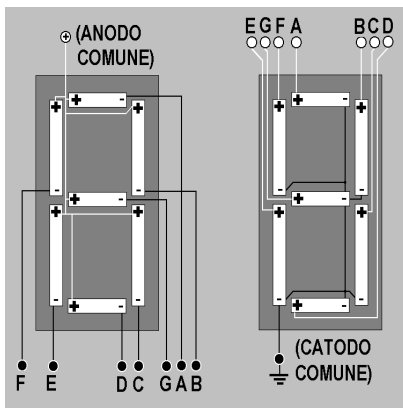


Figura 9-16

Nel primo caso, ovvero nel display ad anodo comune, questo va collegato all'alimentazione positiva; i terminali corrispondenti a ciascuno dei sette segmenti dovranno essere collegati ad altrettanti circuiti in grado di ricevere la corrente proveniente dai segmenti luminosi e chiuderla verso massa.

Nel secondo caso, ovvero nel display a catodo comune, sette circuiti collegati all'alimentazione positiva invieranno corrente nei segmenti che dovranno risultare accesi; le correnti in entrata, attraversati i vari segmenti luminosi, confluiranno poi al catodo comune e quindi a massa.

Chi desidera esercitarsi con i numeri binari ed approfondire il funzionamento di questi componenti, può costruire il semplice circuito illustrato in figura 9-17.

Come per il circuito del sommatore, anche in questo caso useremo dei commutatori o switch, per impostare i valori binari in ingresso. I quattro switch, indicati con le sigle da SW1 a SW4, commutano ciascuno dei quattro ingressi dell'integrato sulla tensione positiva oppure a zero;

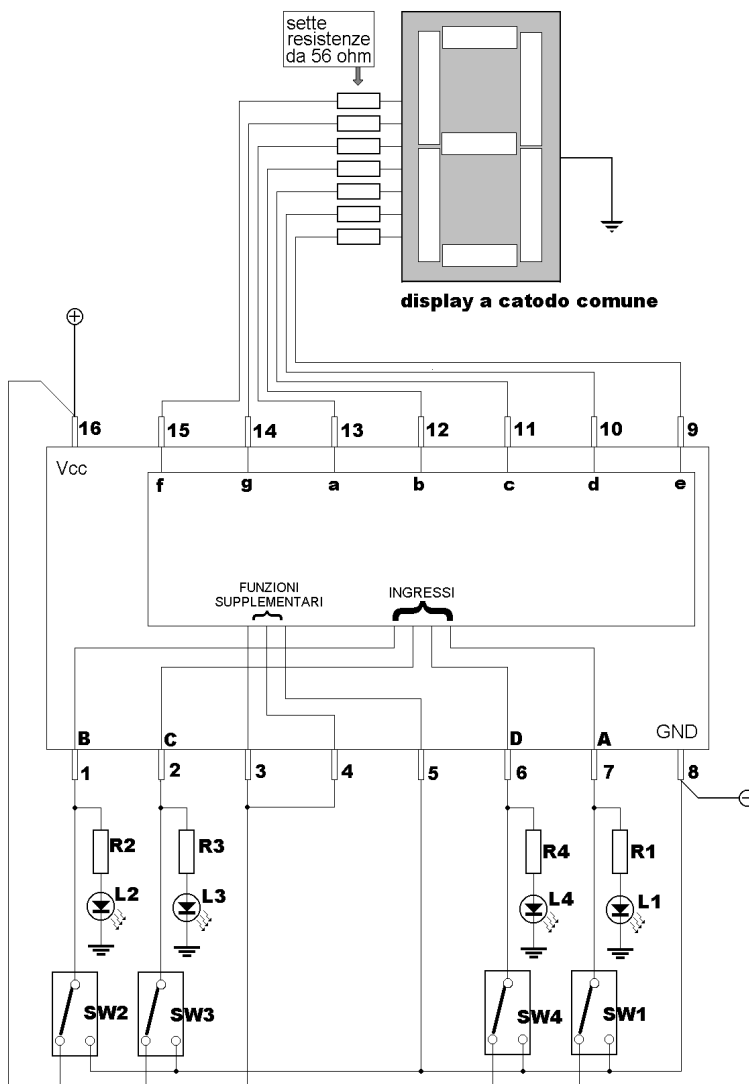


Figura 9-17

quando uno switch è collegato alla tensione positiva, ovvero è in posizione di 1, risulta acceso il LED corrispondente (ci sono naturalmente quattro LED, indicati come L1, L2 ecc.)

Il numero di ogni switch indica anche il suo valore come cifra binaria: lo switch 1 rappresenta la cifra meno significativa, cioè di minor peso; lo switch 4 rappresenta la cifra più significativa.

Le uscite del circuito integrato decodificatore vanno collegate ai relativi ingressi sul display, facendo attenzione ad individuare i segmenti corrispondenti. Come già detto, il display deve essere del tipo a catodo comune; può andare bene, ad esempio, il tipo FND357, illustrato in figura 9-18. Nel disegno a destra, che raffigura la vista posteriore, sono riportate per ogni piedino delle lettere che indicano il segmento luminoso corrispondente. I due piedini segnati k sono internamente collegati insieme e corrispondono al catodo comune; il piedino dp indica il punto decimale, che nel nostro caso non viene utilizzato.

Fra il decodificatore ed il display vanno inserite sette resistenze da 56 ohm, una per ogni segmento luminoso. Per il corretto funzionamento del decodificatore, non dimenticate di collegare alla tensione positiva i piedini 3 e 4 ed alla massa (cioè alla tensione zero) il piedino numero 5. Tutti i collegamenti necessari sono comunque indicati nello schema.

Anche questo circuito può essere alimentato con una pila da 4,5 volt.

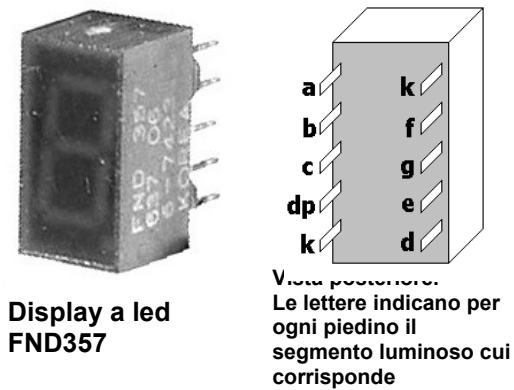


Figura 9-18

CAPITOLO 2

I flip-flop

Con la loro innata predisposizione a chiamare le cose con nomi spicciativi, spesso non privi di una certa ironia, gli americani hanno appioppato il curioso nome di **Flip-Flop** ad un particolare circuito digitale, forse proprio per la sua caratteristica di commutarsi fra due stati logici, una volta da un lato,

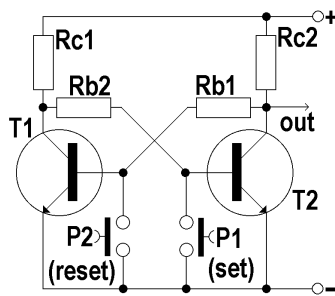


figura 10-1 A

INP 1	INP 2	USCITA PORTA AND	USCITA PORTA NAND
H	H	H	L
L	H	L	H
H	L	L	H
L	L	L	H

Tavola della verità per porta AND e per porta NAND

collettore di T1 va a zero, e così anche la tensione di base di

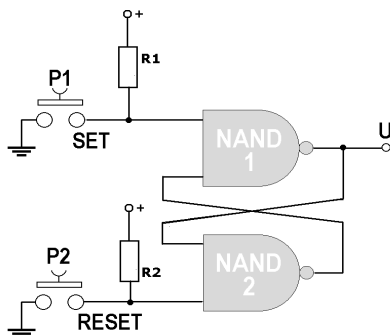


figura 10-1 B

Supponiamo che inizialmente l'uscita U sia a livello basso. Lo stesso livello basso si troverà allora anche sull'ingresso della porta NAND2 che è collegato all'uscita U: avendo un ingresso alto e uno basso, l'uscita della porta NAND2 dovrà essere per forza di cose essere a livello alto. Questo livello alto sarà presente anche sull'ingresso della porta NAND1 ad essa collegato; poiché l'altro ingresso di NAND1 era già alto, l'uscita di tale porta sarà a livello basso. La situazione descritta è una situazione assolutamente stabile.

una volta dall'altro. Il flip-flop può essere considerato il circuito base in grado di memorizzare un'informazione. Esistono tipi diversi di flip flop: cominceremo esaminando il funzionamento del tipo detto "set-reset", illustrato in figura 10-1 A, nella versione a transistor.

Schematicamente esso può essere rappresentato come un blocco funzionale dotato di due ingressi o, meglio, punti di comando, e di un'uscita (out); i due ingressi vengono chiamati con nomi diversi: uno è l'ingresso di **set** (che sta per "attivare, predisporre"), l'altro è l'ingresso di **reset** (termine che sta per "riazzerare"). Supponiamo che ad un certo istante il transistor T2 sia in conduzione; la tensione sul suo collettore è praticamente zero, e quindi anche la base di T1, ad esso collegata tramite Rb1, risulta a tensione zero. Ma se la base del transistor T1 è a zero, ciò significa che esso non conduce, e quindi sul suo collettore c'è tensione; tale tensione risulta applicata, tramite Rb2, anche alla base di T2, che, appunto come specificato in partenza, risulta in conduzione. Tale situazione risulta assolutamente stabile e, se consideriamo come uscita del flip-flop il collettore di T2 (punto out), possiamo dire che siamo nella condizione di reset, cioè di "uscita a zero". Per modificare lo stato dell'uscita occorre premere il pulsante di set P1; la base di T2 va così a zero, bloccando la conduzione del transistor, e di conseguenza la tensione di collettore sale. Tale tensione, che risulta applicata alla base di T1, porta in conduzione questo secondo transistor; la tensione di collettore di T1 va a zero, e così anche la tensione di base di

collettore di T1 va a zero, e così anche la tensione di base di T2, che, a maggior ragione, smette di condurre. A questo punto l'uscita out è a livello alto, ovvero è "settata". Per tornare alla condizione iniziale occorre premere il pulsante di reset (P2).

In figura 10-1 B vediamo lo schema di un flip-flop Set-Reset realizzato con due porte logiche di tipo NAND. Come si vede nella tabella in alto, la porta NAND si comporta in modo opposto alla porta AND: l'uscita va a livello basso **soltanto** quando entrambi gli ingressi (INP1 e INP2) sono a livello alto.

Nello schema di figura 10-1 B, si vede che ciascuna delle due porte ha un ingresso collegato a livello alto (resistenze R1 ed R2), mentre l'altro ingresso è collegato in modo incrociato all'uscita dell'altra porta.

Proviamo adesso a mandare per un attimo l'ingresso Set a livello basso; non essendo più entrambi gli ingressi a livello alto, l'uscita U andrà a livello alto. Il livello alto arriverà anche all'ingresso della NAND2, la cui uscita scatterà a livello basso (poiché l'ingresso Reset è già a livello alto). Tale livello basso giungerà anche all'ingresso della NAND1, confermando il livello alto sull'uscita U. Come per lo schema di fig. 1A., si dice che l'uscita del flip flop è stata "settata", cioè portata ad UNO. Come è facile constatare, anche questa situazione è stabile. Se vogliamo riportare le cose alla situazione precedente, dobbiamo "resettare", ovvero azzerare l'uscita, e questo lo si fa applicando un livello basso sull'ingresso di Reset.

E' utile osservare che questo semplice circuito (non importa se realizzato con transistor o con porte logiche) rappresenta una "cella di memoria": avendo la capacità di mantenersi indefinitamente in uno stato o nell'altro, esso è in grado di immagazzinare un'informazione elementare, ovvero minima, che viene chiamata "bit". Lo stato dell'uscita del flip flop ci dice se quel bit vale uno oppure zero. Con quattro circuiti flip-flop uguali a quelli descritti, e collegati in modo da lavorare secondo una logica opportuna, è possibile memorizzare un numero compreso tra zero e quindici.

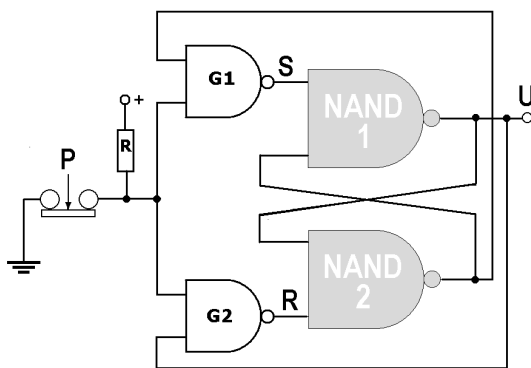


figura 10-2

Dal tipo di flip-flop appena visto passiamo adesso ad un circuito leggermente più complesso, ma di grande utilità pratica: si tratta del flip-flop detto JK. Quello che si vede in figura 10-2 è una versione alquanto semplificata di tale flip-flop, ma utile per comprenderne il principio di funzionamento. La differenza fondamentale tra questo circuito ed il precedente, come capiremo meglio in seguito, consiste nell'aver sostituito i due pulsanti con uno solo: questo unico pulsante, ogni volta che viene premuto, fa commutare lo stato logico dell'uscita (da livello basso a livello alto e viceversa).

Come è possibile comandare con un unico pulsante entrambi gli ingressi, ovvero quello di set e quello di reset? Il compito di inviare l'impulso all'ingresso

giusto viene svolto dalle due porte NAND aggiunte in entrata (G1 e G2). Come si è visto, una porta NAND ha l'uscita a livello basso solo quando entrambi i suoi ingressi sono a livello alto; osservando lo schema, si nota che la porta G1 ha un ingresso collegato all'uscita di NAND2, mentre la porta G2 ha un ingresso collegato all'uscita NAND1: in tal modo, delle due porte G1 e G2, soltanto una si troverà col livello alto in ingresso, e sarà quindi in grado di applicare l'impulso di set o di reset. Le due porte G1 e G2, hanno l'altro ingresso collegato a massa attraverso il pulsante P: quando il pulsante si apre, la resistenza R porta a livello alto gli ingressi di G1 e di G2, ma solo una delle due (che ha già un livello alto anche sull'altro ingresso) potrà far commutare l'uscita.

Il flip-flop come divisore di frequenza

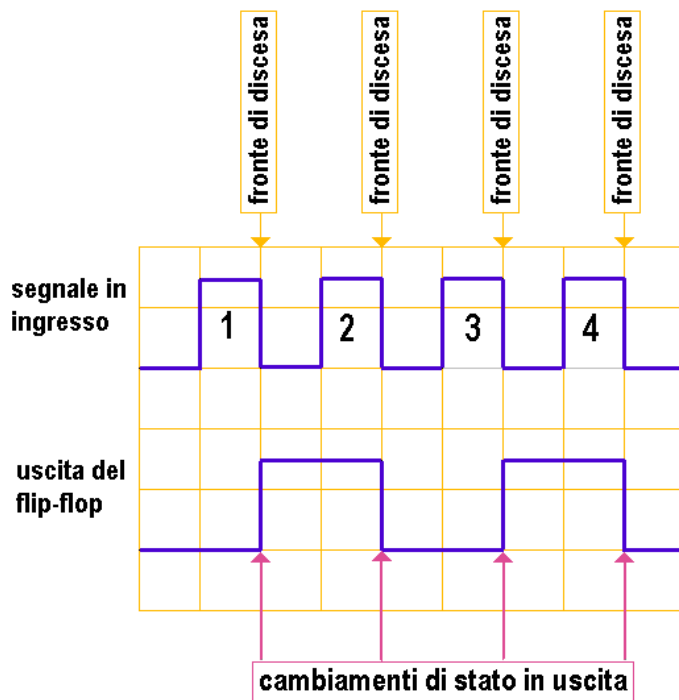


Figura 10-3

Proviamo adesso a sostituire il pulsante con un segnale, esattamente una sequenza di impulsi rettangolari (che costituisce quello che viene chiamato “clock”). Questa sequenza di impulsi, applicata all’ingresso comune fra G1 e G2, ha la stessa funzione del pulsante: quando è presente un impulso, e cioè la tensione in ingresso è a livello alto, è come se il pulsante fosse aperto; quando l’impulso termina, nel momento in cui la tensione torna a zero (detto “fronte di discesa del segnale”), è come se il pulsante venisse premuto, collegando a massa l’ingresso. Succede allora che, in corrispondenza della fine di ogni impulso, ovvero in corrispondenza del “fronte di discesa”, l’uscita del nostro flip-flop cambia di stato. Quello che si verifica è illustrato in figura 10-3: si vede che, in corrispondenza del “fronte di discesa” dell’impulso 1, l’uscita del flip-flop commuta, ovvero cambia di stato, passando a livello alto. Quindi arriva l’impulso 2: in corrispondenza del suo

fronte di discesa, l’uscita commuta di nuovo, tornando questa volta a livello basso, e così via. Quanto abbiamo visto è della massima importanza: basta osservare che, per avere in uscita dal flip-flop un impulso completo, occorre attendere che all’ingresso siano arrivati due impulsi; in pratica si forma un impulso in uscita per ogni due in ingresso!

Il nostro circuito può allora essere usato come un divisore di frequenza. Supponiamo che ogni quadretto della griglia di figura 10-3 rappresenti un tempo di 10 millisecondi: il segnale in ingresso presenta un impulso ogni quadretto, cioè ogni 10 ms; vuol dire che la sua frequenza è di 100 Hertz.

Guardiamo adesso il segnale in uscita: esso presenta un impulso ogni due quadretti; la sua frequenza è allora di 50 hz, esattamente la metà di quella in ingresso!

Un circuito come questo è di grandissima utilità; se inviamo l’uscita del flip-flop ad un altro circuito uguale, otteniamo un segnale in uscita la cui frequenza risulta nuovamente dimezzata, e così via; collegando in cascata quanti circuiti vogliamo, possiamo ottenere in uscita qualunque frequenza, partendo da una frequenza elevata quanto si vuole.

Facciamo un esempio pratico: negli orologi, la scansione dei secondi richiede una frequenza di 1 hz; si tratta di una frequenza bassissima, che nessun oscillatore potrebbe produrre con adeguata precisione ed affidabilità. Esistono invece dei dispositivi come i quarzi, che sono in grado di oscillare con estrema precisione, ma a frequenze molto elevate: alcuni a 32.768 hz, altri ad 1 megahertz, ecc.; ecco che allora si costruisce un oscillatore a quarzo e poi, con un numero adeguato di divisori, si divide la frequenza più volte fino ad ottenere quella di 1 hz, ovvero di un battito al secondo.

Il flip-flop come contatore

Lo stesso circuito che, come si è visto, funziona da divisore di frequenza, può agevolmente essere impiegato per realizzare gli utilissimi “contatori”. Tale nome, che in inglese diventa “counter”, indica un insieme di circuiti in grado di conteggiare una sequenza di impulsi, restituendo in uscita il totale sotto forma di codice binario. Supponiamo di collegare in cascata tre flip-flop come quello di figura 10-2; otterremo il circuito di figura 10-4, composto da un ingresso INP e tre uscite, una per ciascuno dei tre flip-flop: OUT1, OUT2 e OUT3.

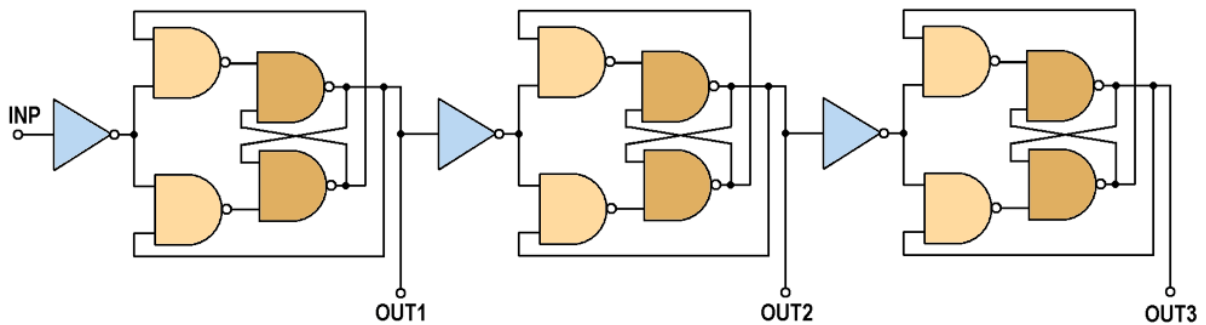


figura 10-4

Come si vede, l'uscita di ogni stadio è applicata all'ingresso del successivo. Il funzionamento dei tre stadi è quello mostrato nella figura 10-5:

alla fine dell'impulso 1 l'uscita del 1° stadio passa a livello alto

alla fine dell'impulso 2 l'uscita del 1° stadio torna a zero, facendo scattare a livello alto l'uscita del 2° stadio

alla fine dell'impulso 3 l'uscita del 1° stadio torna a livello alto

alla fine dell'impulso 4 l'uscita del 1° stadio torna a livello basso, facendo scattare a livello basso quella del 2° stadio; a sua volta, l'uscita del 2° stadio, passando a livello basso, fa scattare l'uscita del 3° a livello alto...

...la sequenza si ripete con lo stesso meccanismo per tutti i vari stadi in cascata.

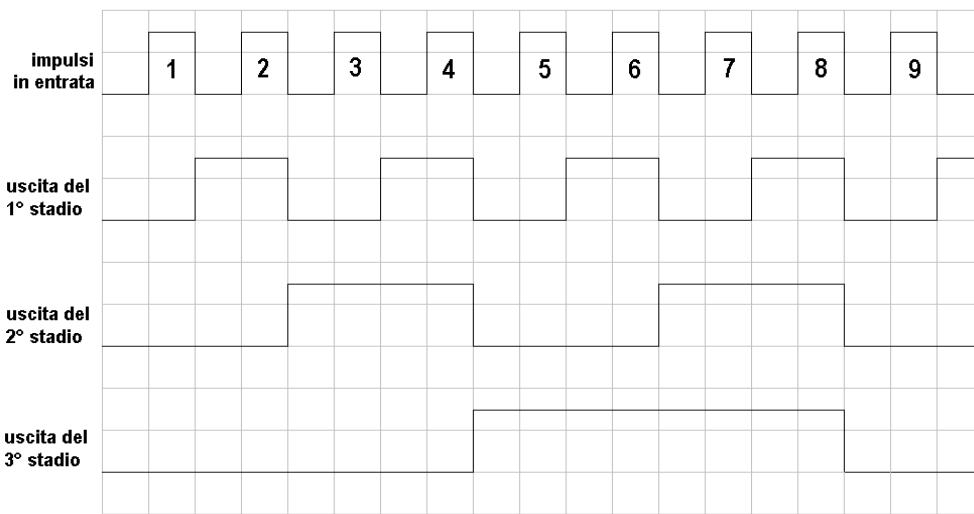


Figura 10-5

Riportiamo adesso in una tabella lo stato delle tre uscite in funzione degli impulsi in ingresso, indicando le uscite a livello basso con L (low) e quelle a livello alto con H (high).

Num. impulsi	OUT 1	OUT 2	OUT 3
0	L	L	L
1	H	L	L
2	L	H	L
3	H	H	L
4	L	L	H
5	H	L	H
6	L	H	H
7	H	H	H
8	L	L	L

Vediamo che all'inizio le uscite sono tutte L. Dopo il primo impulso l'uscita OUT 1 diventa H; in corrispondenza del secondo impulso OUT1 torna ad L mentre OUT2 diventa H, ecc. Osserviamo che dopo il settimo impulso le uscite sono tutte su H; con l'impulso successivo, invece, tutto si riazzera: le uscite tornano a livello L e la sequenza ricomincia come dall'impulso 1. Questo significa che il nostro circuito può contare fino ad 8 (o, meglio, da zero a sette), e ciò per il fatto che è composto da tre flip-flop: infatti 2^3 è uguale a 8. Per ogni flip-flop che si aggiunge, si raddoppia il massimo numero raggiungibile: con 4 circuiti si possono contare fino a 16 impulsi, con 5 circuiti fino a 32 impulsi, ecc.

Come si fa a sapere quando gli impulsi hanno raggiunto un certo numero? Supponiamo di voler fare in modo che succeda qualcosa dopo 6 impulsi: per esempio far suonare un allarme. Quello che

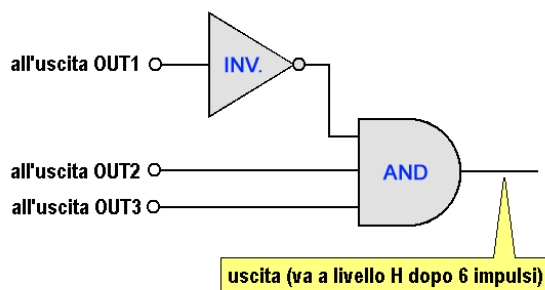


Figura 10-6

dobiamo fare è semplicemente “leggere” lo stato delle tre uscite del nostro circuito: dalla tabella vediamo che in corrispondenza dell'impulso 6 le uscite sono L - H - H; tale condizione deve essere rilevata da un adatto circuito logico: per esempio possiamo usare una porta AND a tre ingressi, inserendo un invertitore sull'ingresso che viene collegato alla OUT1, che risulta a livello L. Anche la porta AND a tre ingressi fornisce un livello alto in uscita solo quando sono alti tutti i suoi tre ingressi; se allora colleghiamo la porta come si vede in figura 10-6, otteniamo che in corrispondenza dell'impulso 6, e solo in quel caso, i tre ingressi della porta AND saranno tutti a livello alto, per cui

la nostra porta ci fornirà un livello alto in uscita, indicando che il numero di sei impulsi è stato raggiunto.

Oscillatori realizzati con porte logiche

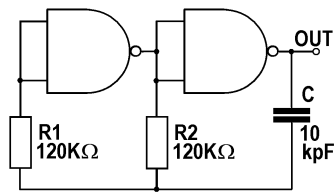


Figura 10-7

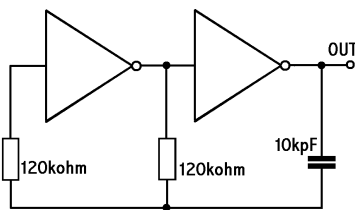


Figura 10-8

Con le porte logiche si possono anche realizzare circuiti simili a quelli visti nell'elettronica lineare. Per esempio, l'oscillatore della figura 10-7 è realizzato con due porte nand; come si vede, i due ingressi di ogni porta sono collegati insieme per formare un ingresso unico. I valori delle due resistenze ($120\text{ K}\Omega$) e del condensatore (10 kF) sono indicativi; se si aumentano, si ottengono frequenze più basse, e viceversa. Le due resistenze possono anche avere valori diversi. Il segnale in uscita (out) è un'onda quadra, naturalmente a livello logico, cioè che varia fra lo zero e la tensione di alimentazione.

Lo stesso oscillatore può essere realizzato anche con altri componenti, in funzione degli elementi che di volta in volta risultano disponibili in un circuito. Poiché i circuiti integrati contengono, a seconda dei tipi, due, quattro, sei o più porte, può capitare che, nel realizzare un determinato circuito, alcune porte non vengano usate: ecco che allora, se occorre un oscillatore, risulta conveniente costruirlo con le porte rimaste inutilizzate. In figura 10-8 vediamo lo stesso oscillatore realizzato con due invertitori (in inglese "inverter").

Anche in questo caso valgono le stesse considerazioni fatte precedentemente in merito al valore dei componenti.

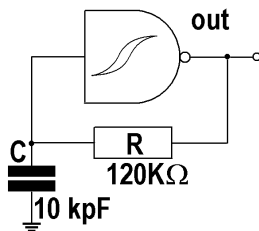


Figura 10-9

Quello di figura 10-9 è un oscillatore molto semplice, realizzato però con un componente un po' particolare: come indica il simbolo, si tratta di un "trigger di Schmitt". Questo componente è caratterizzato in ingresso da due soglie, ovvero due livelli di tensione, che possiamo chiamare V_{t+} e V_{t-} . Quando la tensione applicata in ingresso raggiunge il valore di soglia V_{t+} l'uscita del trigger commuta e scatta a livello basso; per tornare a livello alto, occorre che il segnale in ingresso scenda fino a raggiungere il valore della soglia V_{t-} . Il trigger di Schmitt viene usato per esempio negli stadi di ingresso, per rendere il circuito di conteggio insensibile ai disturbi che possono accompagnarsi ai segnali da elaborare.

Il circuito di figura 10-9 funziona in modo analogo ad altri oscillatori: con l'uscita a livello alto, il condensatore C comincia a caricarsi attraverso la resistenza R; quando la tensione sul condensatore raggiunge il livello V_{t+} l'uscita commuta e va a zero. A questo punto il condensatore comincia a scaricarsi, sempre attraverso la resistenza R, collegata all'uscita, che adesso è a livello zero. Quando la tensione sul condensatore è scesa fino al livello V_{t-} , il trigger commuta nuovamente e l'uscita torna a livello alto; il condensatore ricomincia a caricarsi ed il ciclo si ripete.

In figura 10-10 si vede lo schema interno di un integrato che contiene sei trigger di Schmitt: il tipo MC14584 prodotto dalla Motorola (equivalente "pin to pin" all'MM74C914 ed al CD40106 della National).

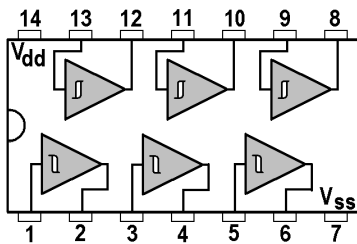
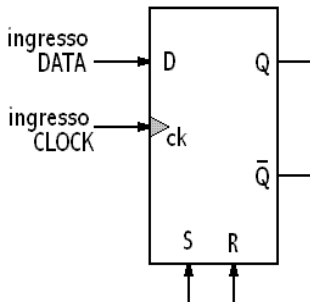


Figura 10-10

Tastiera a combinazione

Vedremo adesso la realizzazione di un circuito "logico" che permette di attivare un dispositivo tramite la pressione di determinati tasti di una tastiera, e nel giusto ordine. Il circuito descritto costituisce anche un'occasione per meglio comprendere l'uso dei flip-flop e delle porte logiche.

Il circuito utilizza quattro flip-flop del tipo "DATA", comunemente chiamati D-Flip Flop.



Il D-flip-flop è un sottoinsieme logico caratterizzato da due stati, entrambi stabili, che si manifestano sulle due uscite Q e Q invertita (una Q con sopra un trattino); ha un ingresso "data", un ingresso "clock", e due terminali, uno per il SET e uno per il RESET.

I due terminali "Set" e "Reset" devono essere generalmente a livello "L" (low, cioè zero logico). Applicando un livello "H" (high, cioè uno logico) al terminale SET le uscite commutano nello stato corrispondente (stato di "set"), assumendo l'uscita Q un livello H e l'uscita Q invertita un livello L; applicando un livello "H" al terminale RESET le uscite commutano nello stato corrispondente (stato di "reset"), assumendo l'uscita Q un livello L e l'uscita Q invertita un livello H.

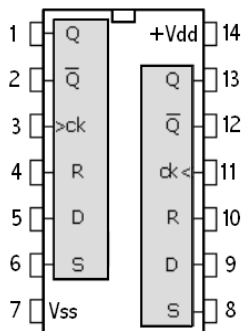
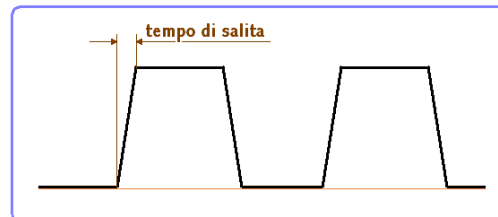
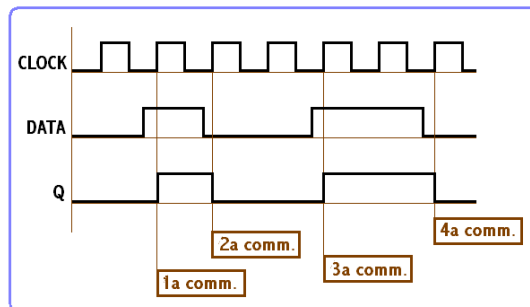
Ma cosa fa un D-Flip Flop? Consideriamo che all'ingresso "data" può essere presente, a seconda degli istanti, uno stato logico alto o uno stato logico basso: ebbene, questo stato viene trasferito all'uscita Q in corrispondenza del fronte di salita del segnale di "clock" (ricordiamo che contemporaneamente

l'uscita Q invertita assumerà un livello logico opposto).

Con il termine "clock" si intende un segnale periodico di riferimento temporale, in genere avente forma di onda quadra. In breve, esso consiste in una tensione il cui livello passa alternativamente da L (basso, uguale a zero) ad H (alto, uno logico, che coincide con la tensione di alimentazione); questi cambiamenti di stato devono avvenire in modo rapido, per garantire la corretta commutazione dei circuiti logici.

In figura è illustrato il funzionamento di un flip-flop tipo "data": in alto si vede il segnale di clock e, al centro, uno stato logico casuale che risulta presente sull'ingresso "D"; l'ultimo tracciato indica lo stato dell'uscita Q del flip-flop, dove si osserva che lo stato logico presente sul terminale D viene trasferito all'uscita Q solo nell'istante in cui il clock passa da livello L a livello H, e cioè in corrispondenza del "fronte di salita".

Nella seconda figura è illustrato in particolare un segnale di clock: nella realtà il passaggio da uno stato L a uno stato H non è mai immediato, ma richiede un certo tempo, sia pure brevissimo, che viene definito "tempo di salita" (in inglese "rise time"); è il tempo di salita che deve essere sufficientemente breve perché il flip-flop possa commutare con sicurezza.



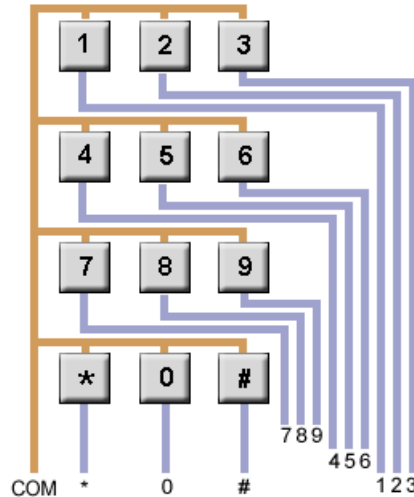
CD4013

Una curiosità: il piccolo triangolo che si osserva vicino all'ingresso "clock" indica che si tratta di un flip-flop di tipo "edge triggered", ovvero che risponde solo a cambi di stato repentini (il segnale di clock deve quindi avere un fronte ripido); nel caso specifico, si tratta di un "rising-edge triggered", cioè di un tipo di flip-flop che risponde solo al passaggio dal livello L al livello H.

Il circuito integrato CD4013 (figura a lato) contiene due flip-flop del tipo descritto; il primo utilizza i piedini da 1 a 6, mentre il secondo fa capo ai piedini da 8 a 13. Il piedino 7 va collegato alla massa, ovvero al polo negativo, mentre la tensione di alimentazione, che può essere compresa fra 3 e 15 V, va collegata al piedino 14.

Per realizzare il circuito che stiamo descrivendo occorre una tastiera, tipo quelle telefoniche, che può avere 9 o più cifre. In breve essa consiste in tanti interruttori che si chiudono ogni volta che un tasto viene premuto. Nel procurarsi la tastiera, bisogna evitare quelle cablate a matrice, che non potrebbero funzionare nel circuito che stiamo descrivendo;

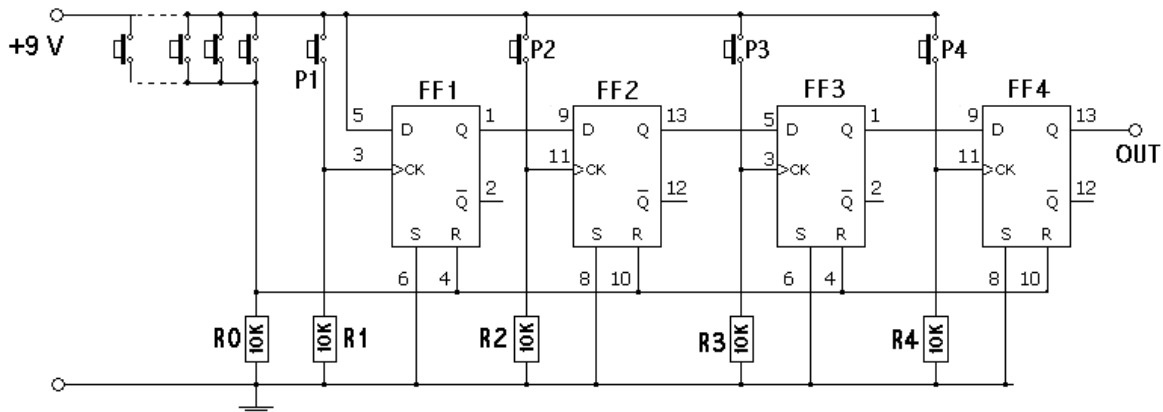
bisogna usare una tastiera dove i due siano liberamente accessibili (figura a come ci occorre. In alternativa, è tastini sfusi, ciascuno con il suo suoi contatti. Non occorre che la bastano anche i soli numeri da 0 a 9. Poiché il nostro circuito utilizza scegliere a nostro piacere 4 numeri combinazione in grado di sbloccare i tasti corrispondenti, ciascuno con altri tasti non saranno collegati, ma confondere le idee a chi non conosce Osserviamo ancora che i tasti nel giusto ordine: premendo anche una giusta sequenza, i vari flip-flop Introdotta l'esatta combinazione, livello logico "H": noi sfrutteremo tale qualsiasi altro dispositivo che ci allarme, ecc.)



terminali di ciascun tasto lato), per poterli collegare possibile procurarsi tanti numero stampigliato ed i tastiera abbia 12 tasti;

quattro flip-flop, dovremo che formeranno la l'uscita: abbineremo quindi uno dei quattro flip-flop. Gli serviranno soltanto a la combinazione. dovranno essere premuti solo numero non nella saranno azzerati. l'uscita "OUT" passerà a tensione per attivare un interessa (serratura,

Nella figura che segue vediamo il circuito completo; esaminiamo il suo funzionamento.

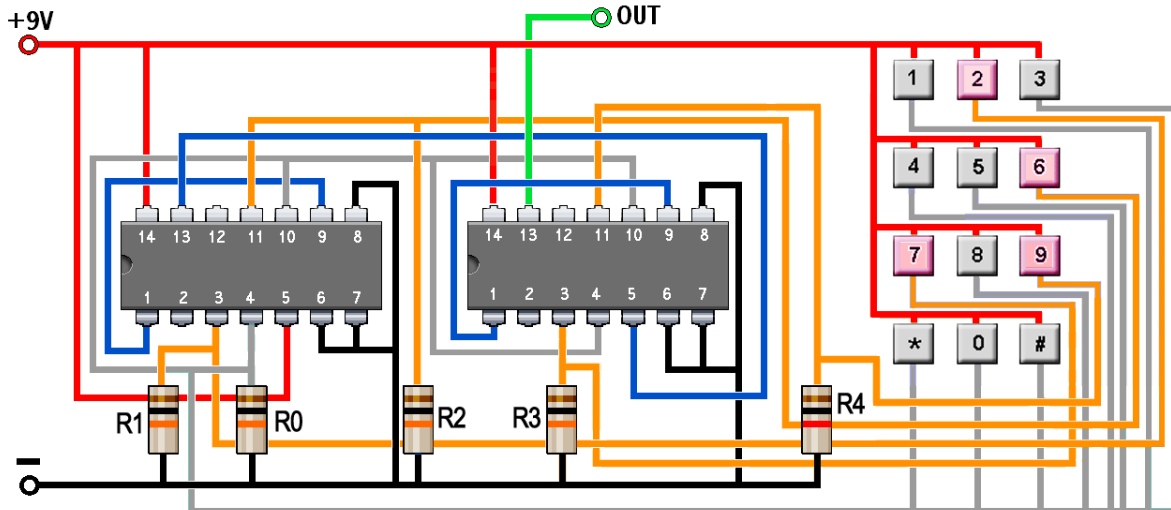


Notiamo subito i vari pulsanti, precisando che a noi interessano solo P1, P2, P3 e P4, ovvero quelli che abbiamo deciso di inserire nella combinazione di apertura. Ognuno di questi pulsanti è collegato all'ingresso "clock" di uno dei quattro flip-flop (da FF1 a FF4).

L'alimentazione, in questo caso di 9 V (ma poteva essere anche un'altra tensione) entra dal lato sinistro: il positivo su "+9V" ed il negativo alla massa. I terminali di SET di tutti i flip-flop sono collegati a massa direttamente, mentre fra i terminali RESET e la massa si trovano delle resistenze da 10 kohm; in effetti anche questi terminali è come se fossero collegati a massa, con la possibilità però di inviare una tensione sulla resistenza e cambiare quindi lo stato logico da L ad H.

L'ingresso D del primo flip-flop è a livello H; quando si preme il primo tasto della combinazione (pulsante P1) si fa transitare l'ingresso clock da L ad H (l'equivalente del fronte di salita), per cui il livello H presente su D viene trasferito all'uscita Q. Ma l'uscita Q di FF1 porta a livello H anche l'ingresso D di FF2, per cui, se adesso si preme il secondo tasto della combinazione (P2), anche il livello H presente sul piedino 9 di FF2 viene trasferito all'uscita Q (piedino 13). Il meccanismo si ripete

procedendo a catena fino a FF4. Ma se, nell'introdurre la combinazione, viene premuto un tasto non previsto (quelli a sinistra nella figura), il tasto fa arrivare un livello H ai piedini di RESET dei vari flip-flop, che vengono pertanto azzerati (resettati, cioè con l'uscita Q a livello L).



Per favorire chi non ha troppa pratica di componenti elettronici, nella figura sopra vengono illustrati tutti i collegamenti da realizzare fra i due integrati e la tastiera.

Osserviamo subito che tutti i tasti della tastiera sono collegati da un lato al positivo della tensione di alimentazione.

Nell'esempio raffigurato è stata scelta, come combinazione, la successione dei tasti 2, 6, 7, 9, che nell'immagine risultano in colore; gli altri tasti, di colore grigio, sono quelli indifferenti, quelli cioè che, se premuti, determinano il reset del sistema.

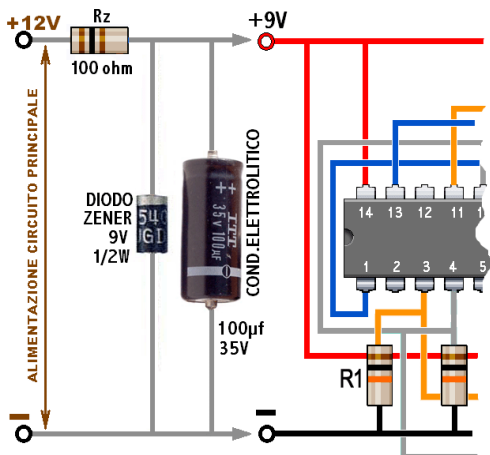
Si vede chiaramente che le uscite di tutti i tasti in grigio risultano collegate insieme e fanno capo alla resistenza R0: premendo uno di tali tasti, sulla R0 arriva la tensione positiva; questo equivale a dire che su tutti i terminali collegati a R0, e cioè i pin 4 e 10 di ciascun integrato, arriva un livello "H" che determina il reset.

I pulsanti che fanno parte della combinazione, P1, P2, P3 e P4 arrivano ciascuno alla propria resistenza, rispettivamente R1, R2, R3 ed R4, e, se premuti, portano a livello "H" l'ingresso di clock del rispettivo flip-flop. In base al funzionamento descritto in precedenza, quando il clock passa da livello "L" a livello "H", lo stato logico presente all'ingresso D viene trasferito all'uscita Q. Siccome D del primo integrato (piedino 5) è collegato al positivo (e cioè è a livello "H"), quando si preme P1 il livello "H" viene trasferito al piedino 1. Il meccanismo prosegue a catena fino all'ultimo flip-flop, la cui uscita Q andrà a livello "H" quando viene premuto P4 (l'ultimo tasto della combinazione).

Poichè questo circuito è destinato a funzionare in abbinamento ad altri dispositivi, la sua alimentazione verrà prelevata da questi ultimi; nel caso che l'alimentazione dell'altro dispositivo sia di 12 V, useremo lo schema della figura a lato (parte tracciata in grigio) per derivare l'alimentazione del nostro circuito.

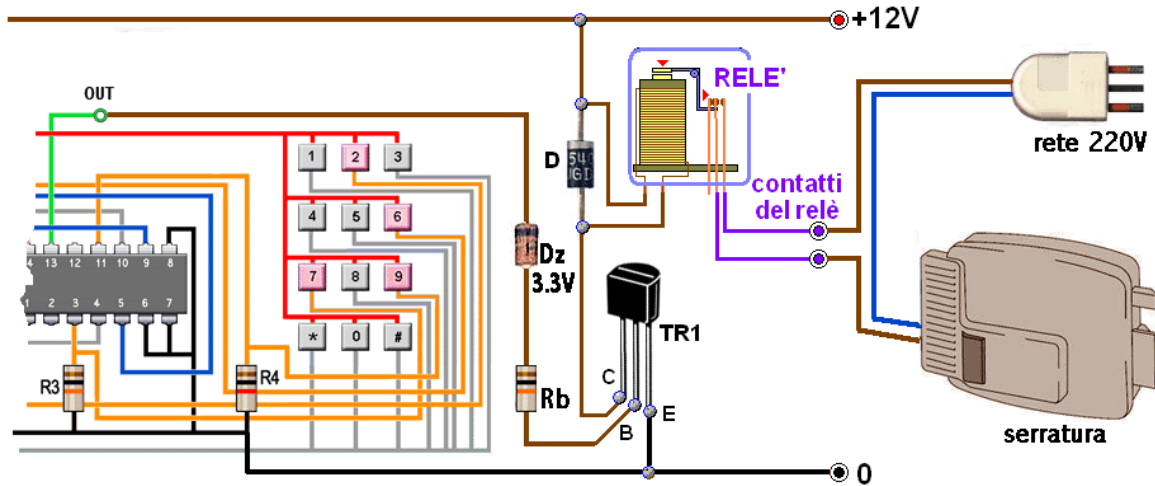
Dal positivo dell'alimentazione a 12V preleviamo la corrente per il nostro circuito attraverso la resistenza da Rz 100 ohm; tale resistenza determina una caduta di tensione di 1 o 2 volt, e consente al diodo zener da 9V di espletare la sua funzione stabilizzatrice.

Il condensatore elettrolitico da 100 µF sopprime eventuali oscillazioni locali.



Serratura elettrica a combinazione

Per fornire un esempio di impiego pratico, vediamo nella figura che segue come il circuito a combinazione possa essere usato per comandare l'apertura di una serratura elettrica.



L'uscita OUT del circuito a combinazione comanda, tramite una resistenza da 10 kohm, la base di un transistor NPN (TR1); quando l'uscita OUT va a livello "H", il transistor passa in conduzione, facendo chiudere il relè che, a sua volta, tramite i suoi contatti, fa arrivare corrente alla serratura, provocandone l'apertura.

Il diodo D, come spiegato altrove, protegge il circuito dalle extracorrenti causate dalla bobina del relè. Dz è invece un diodo zener da 3,3 V, ed ha lo scopo di evitare che il relè possa attivarsi a causa della tensione di qualche centinaio di millivolt che potrebbe comunque essere presente sull'uscita OUT anche quando questa è nominalmente a livello "L"

In funzione del transistor utilizzato, il valore della resistenza di base Rb può avere un valore leggermente diverso da quello di 10 kohm indicato nello schema.

Il diodo D in parallelo al relè può essere un generico 1N4001 o equivalenti; il relè deve funzionare con bobina a 12V ed i suoi contatti devono poter interrompere una corrente di alcuni ampere alla tensione di 220 V alternata.

CAPITOLO 3

Realizzazione di un contasecondi

Allo scopo di prendere confidenza con le tecniche esposte, costruiremo un conta secondi, ovvero un circuito che, dopo un numero prestabilito di secondi, ci fornirà per esempio un segnale o sarà in grado di comandare un evento (praticamente si tratta di un “timer”).

Il nostro circuito si compone di varie sezioni distinte:

- un generatore di clock, ovvero un oscillatore con relativi divisori di frequenza
- un circuito di gating, ovvero di controllo, che abilita o meno il passaggio degli impulsi
- un circuito di conteggio degli impulsi
- un circuito di fine conteggio, che controlla il numero raggiunto ed abilita un segnale in uscita.

Il generatore di clock

Considerato lo scopo prevalentemente didattico di questa realizzazione, è parso opportuno utilizzare, in funzione di generatore di clock, un semplice oscillatore basato sul circuito integrato 555 (ampiamente descritto nella sezione dedicata a tale integrato), evitando quindi l’uso di oscillatori a

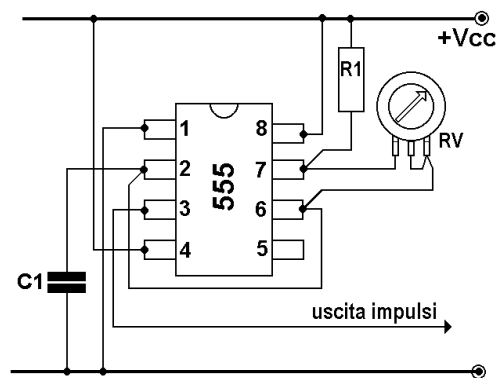
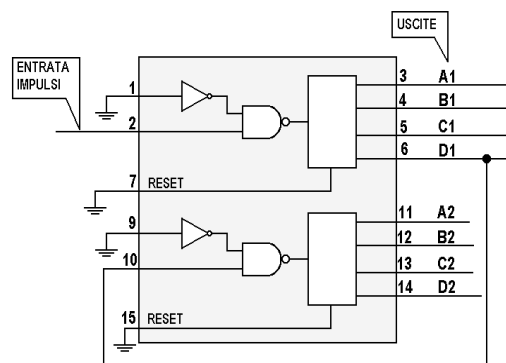


Figura 12-1

quarzo. Scegliendo valori opportuni per C1, R1 ed RV, l’oscillatore di figura 12-1 è in grado di generare una frequenza di 1 hz, e pertanto di fornire in uscita impulsi distanziati esattamente di un secondo uno dall’altro; sarà sufficiente contare tali impulsi per ottenere il contasecondi.

Contatore binario CD4520



DOPPIO CONTATORE BINARIO CD4520

figura 12-2

Il circuito di conteggio può essere realizzato in vari modi, anche in funzione del numero massimo che si intende contare. Nel nostro caso, useremo un circuito integrato CD4520, che contiene al suo interno due contatori binari, ciascuno a 4 bit, ovvero a quattro stadi, ed il cui schema è riportato in figura 12-2. Sul piedino 2 entrano gli impulsi generati dall'oscillatore della figura precedente; tali impulsi vengono quindi conteggiati dai quattro stadi del primo dei due contatori, mentre i risultati del conteggio appaiono sulle relative uscite 1A, 2A, 3A e 4A (corrispondenti ai piedini 3-4-5-6). Osserviamo che lo stato della prima uscita (1A) cambia ad ogni impulso in arrivo, da L ad H e viceversa, mentre lo stato della seconda uscita (2A) cambia ogni volta che l'uscita 1A va a livello L; di conseguenza cambia ogni due impulsi in entrata. A sua volta, l'uscita 3A cambia ogni volta che la 2A va a livello L, e quindi cambia ogni quattro impulsi in entrata; l'ultima, l'uscita 4A, cambia ogni volta che va a livello L l'uscita 3A, e quindi cambia ogni otto impulsi in entrata. E' importante notare che quest'ultima uscita viene anche collegata all'entrata del secondo contatore, esattamente sul piedino 10; in questo modo il conteggio continua sulle altre quattro uscite 1B, 2B, 3B e 4B. In seguito al meccanismo appena illustrato, succede che l'uscita 1B cambia di stato ogni 16 impulsi in ingresso, l'uscita 2B ogni 32 impulsi, l'uscita 3B ogni 64 impulsi e l'uscita 4B ogni 128 impulsi.

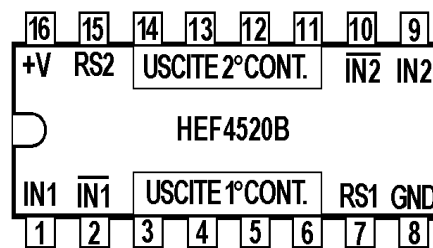


figura 12-3

In figura 12-3 è riportato l'integrato con la relativa piedinatura (fare sempre riferimento allo schema di figura 12-2). Come si vede, ciascuna delle due sezioni ha i suoi piedini di collegamento. Il primo dei due contatori ha come ingressi IN1 (piedino 1) e IN1 invertito (quello con la riga sopra, piedino 2); si può usare un ingresso o l'altro:

- se gli impulsi entrano su IN1, i cambiamenti di stato avvengono in corrispondenza del fronte di salita del segnale (passaggio da low ad high); in questo caso l'ingresso IN1 invertito deve essere collegato a livello alto
- se gli impulsi entrano su IN1 invertito, i cambiamenti avvengono in corrispondenza del fronte di discesa (passaggio da high a low); in questo caso l'ingresso IN1 deve essere collegato a livello basso. Applicando un livello alto sul piedino 7, si ottiene il reset (cioè il riposizionamento a zero) del contatore; nel funzionamento normale tale piedino deve quindi essere collegato a massa.

Le uscite del primo contatore corrispondono ai piedini dal 3 al 6.

Per il secondo contatore valgono le stesse considerazioni fatte per il primo; vi sono ugualmente i due ingressi (IN2 e IN2 invertito), il piedino di reset (piedino 15) e le quattro uscite (piedini da 11 a 14).

Naturalmente il circuito integrato deve essere collegato all'alimentazione: il piedino 8 (GND) alla massa, ovvero a zero volt; il piedino 16 ad una tensione compresa fra 5 e 15 volt.

Circuito di verifica di raggiunto conteggio

Le otto uscite dell'integrato 4520 (schema di figura 12-2) devono essere collegate al circuito verificatore illustrato in figura 12-4; si tratta del circuito che ha lo scopo di leggere lo stato delle otto uscite, e, per così dire, dare l'allarme quando il numero previsto è stato raggiunto.

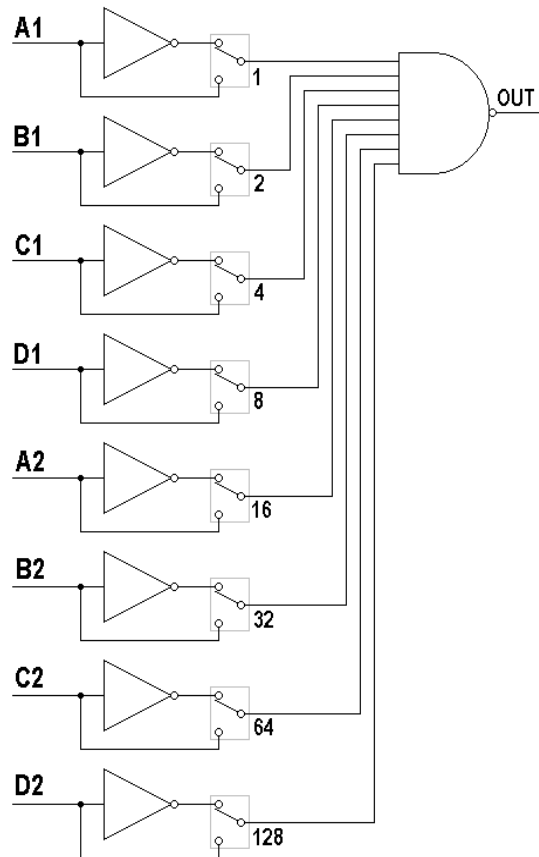


figura 12-4

Lo schema si compone di otto invertitori e di una porta NAND a otto ingressi; si utilizzeranno allo scopo due integrati CD4069 oppure MM74C04 (si chiamano “esainvertitori” perché contengono ciascuno sei invertitori) ed un integrato MM74C30, i cui schemi di collegamento sono riportati in figura 12-5.

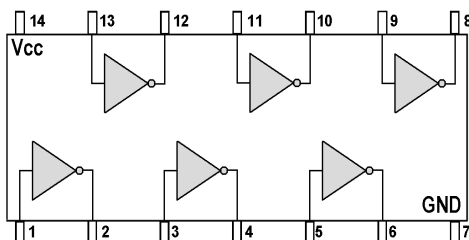
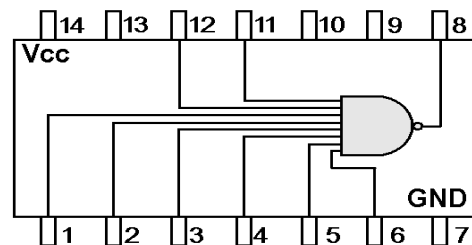


Figura 12-5 Esainvertitore 4069



Porta NAND a 8 ingressi

La porta NAND a otto ingressi fornisce in uscita un livello basso solo quando tutti gli ingressi sono a livello alto. Dovremo allora fare in modo che, quando il contatore raggiunge la condizione che ci interessa, con i relativi livelli L ed H presenti sulle otto uscite, alla porta NAND arrivino in realtà otto livelli H. Per tale motivo è stato previsto l'impiego di otto invertitori ed otto commutatori o switch; come

Nu- mero im- puls i	1 A	2 A	3 A	4 A	1 B	2 B	3 B	4B
0	L	L	L	L	L	L	L	L
1	H	L	L	L	L	L	L	L
2	L	H	L	L	L	L	L	L
3	H	H	L	L	L	L	L	L
4	L	L	H	L	L	L	L	L
5	H	L	H	L	L	L	L	L
6	L	H	H	L	L	L	L	L
7	H	H	H	L	L	L	L	L
8	L	L	L	H	L	L	L	L
9	H	L	L	H	L	L	L	L
10	L	H	L	H	L	L	L	L
11	H	H	L	H	L	L	L	L
12	L	L	H	H	L	L	L	L
13	H	L	H	H	L	L	L	L
14	L	H	H	H	L	L	L	L
15	H	H	H	H	L	L	L	L
16	L	L	L	L	H	L	L	L
...
60	L	L	H	H	H	H	L	L
...
120	L	L	L	H	H	H	H	L
...
155	H	H	L	H	H	L	L	H
...
250	L	H	L	H	H	H	H	H

si vede, ciascuna delle otto uscite (da A1 a D2) può arrivare alla porta NAND direttamente oppure tramite l'invertitore: dipende dalla posizione del commutatore.

A lato è riportata la tabella che mostra come cambiano i livelli delle varie uscite per ogni impulso che viene conteggiato (naturalmente non ci sono tutte le 256 possibili combinazioni, ma solo una parte). Per comprendere meglio il meccanismo, sarà sufficiente qualche esempio:

Supponiamo di voler contare fino a 120 secondi; guardando la tabella, vediamo che a 120 impulsi le uscite hanno i seguenti livelli: L-L-L-H-H-H-H-L. Noi vogliamo che la porta NAND commuti una volta raggiunta questa situazione: dobbiamo allora fare in modo che tutti gli ingressi della porta siano a livello alto. Le uscite 4-5-6-7 sono già a livello H, per cui le collegheremo alla porta direttamente, scegliendo la giusta posizione dei relativi switch. Le uscite 1-2-3-8 sono a livello L, per cui devono essere invertite: sposteremo allora i relativi commutatori (o switch) nella posizione in cui il collegamento passa tramite l'invertitore.

Secondo esempio: vogliamo che il conteggio si arresti al numero 155, situazione in cui le uscite si presentano con la combinazione H-H-L-H-H-L-L-H; dovremo allora far passare tramite invertitore le uscite 3, 6 e 7, per cui sposteremo nella giusta posizione i tre switch corrispondenti.

Se osserviamo bene, si tratta in pratica di collegare le uscite del circuito di conteggio alla porta NAND tutte tramite invertitore, tranne quelle che, nel sistema binario, corrispondono al numero prescelto.

In seguito, comunque, parlando dell'uso pratico del contasecondi, torneremo ancora su questo concetto.

Per l'alimentazione dei vari circuiti che costituiscono il contasecondi può essere utilizzata una pila piatta da 4,5 volt, oppure una qualsiasi altra tensione proveniente da un alimentatore, purché compresa fra 4,5 e 12 volt.

Il materiale occorrente per la realizzazione è elencato nella tabella a lato. Come già detto, un esainvertitore contiene al suo interno sei invertitori; poiché a noi ne occorrono otto, ed un solo integrato non è sufficiente, dei dodici invertitori, quattro resteranno inutilizzati. Sarebbe stato possibile utilizzare parte di questi invertitori per realizzare l'oscillatore, senza ricorrere all'integrato NE555; in realtà, considerato che quest'ultimo integrato è stato progettato proprio in funzione dell'uso come generatore di intervalli di tempo, esso garantisce una miglior precisione e stabilità, anche nel caso di variazioni della tensione di alimentazione. Chi, comunque, costruisce il contasecondi a scopo di esercitazione, e non è interessato ad una particolare precisione, può evitare di procurarsi l'integrato NE555 e costruire l'oscillatore riportato in figura 12-6. I valori da usare sono:

0,68 μ F per il condensatore C

1,8 M Ω per R1

330 K Ω per R2

1 M Ω per RV

MATERIALE OCCORRENTE PER LA REALIZZAZIONE DEL CONTASECONDI	
Circuito oscillatore	Quantità
Timer NE555	1
C1: condensatore da 3,3 μ F	1
R1: resistenza da 1 K Ω	1
RV: trimmer da 470 K Ω	1
Circuito di conteggio	
Doppio contatore binario a 4 bit: CD4520	1
Circuito di verifica	
Esainvertitore: CD4069 oppure MM74C04	2
Porta nand a 8 ingressi MM74C30	1
Commutatori o switch	8
Circuito di gating	
4 porte NAND a 2 ingressi: MM74C00 oppure CD4011	1
R1 e R2 = 10 K Ω	2
C1 = 2 μ F	1
Pulsante di start	1

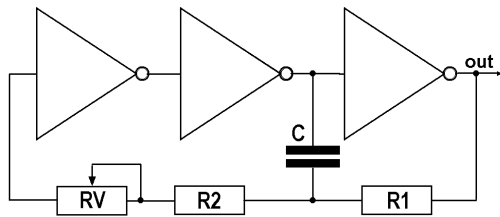


figura 12-6

In ogni caso, sia con tale circuito, sia con il 555, occorre procedere ad una paziente taratura dell'oscillatore, provando a variare il valore della resistenza RV, fino ad ottenere l'esatta corrispondenza degli impulsi generati con il battito dei secondi. Tenete sempre presente che i valori indicati sono teorici, per cui nel funzionamento reale potrebbe rendersi necessaria qualche piccola modifica dovuta alla tolleranza dei componenti. Supponiamo che nel circuito di figura 12-6, anche variando la RV il battito

risulti sempre troppo veloce; provate allora ad aumentare il valore di R1 (per esempio a 2,2 MΩ) o di R2 (per esempio a 560 KΩ).

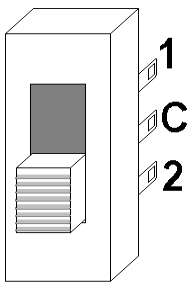


figura 12-7

Per finire, si collegano gli otto commutatori (in inglese *switch*) che permettono di selezionare il valore a cui deve terminare il conteggio. Potete usare qualsiasi tipo di switch in grado di commutare un contatto centrale in una posizione o nell'altra, come quello illustrato in figura 12-7, dove il polo centrale C può essere commutato in posizione 1 o 2; ne monterete otto tutti in fila su di un pannello o sulla scatola che userete come contenitore del contasecondi, e scriverete (magari con i trasferibili) in corrispondenza di ogni switch il valore di una potenza di 2 (come si vede in figura 12-8). Per impostare il numero di secondi a cui vogliamo che il conteggio si arresti, e quindi che venga dato il segnale, basta alzare le levette dei commutatori: volendo realizzare per esempio un intervallo di tempo di 60 secondi, occorrerà alzare le levette 32, 16, 8 e 4 perché 32+16+8+4 fa 60; per ottenere 100 secondi si alzeranno le levette 64, 32 e 4, e così via. L'importante è collegare i commutatori correttamente,

ovvero facendo in modo che, con la levetta in basso, il segnale passi attraverso gli invertitori, mentre con la levetta in alto il segnale sia collegato direttamente.

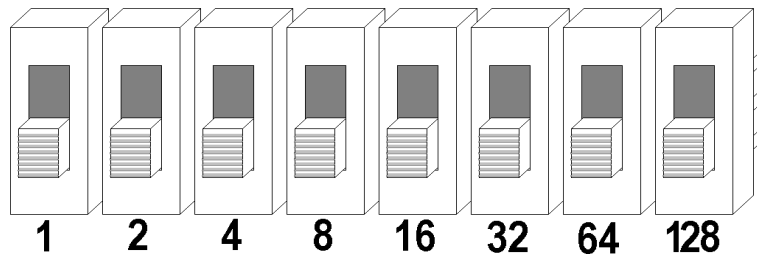


figura 12-8

C'è ancora un particolare da sistemare: col circuito che abbiamo visto, succede che appena si collega l'alimentazione ha inizio il conteggio; questo non va bene, per cui occorre aggiungere un semplice dispositivo per realizzare un controllo (gating) del segnale. Possiamo per esempio intervenire sull'ingresso del primo contatore, esattamente sul piedino 1; avevamo visto infatti che per abilitare l'ingresso del segnale occorre collegare tale piedino a livello L (cioè a massa). Con due porte NAND di un integrato tipo MM74C00 oppure CD4011 (vedere piedinatura rispettivamente nelle figure 12-10 e 12-11) costruiamo allora un flip-flop come quello di figura 10-1, attuando però qualche piccolo accorgimento, come si vede in figura 12-9. La resistenza R1 e il condensatore C1 fanno in modo che all'accensione, essendo C1 scarico, sull'ingresso della porta venga applicato un livello L; questo garantisce che il flip-flop si posizioni con l'uscita U a livello H.

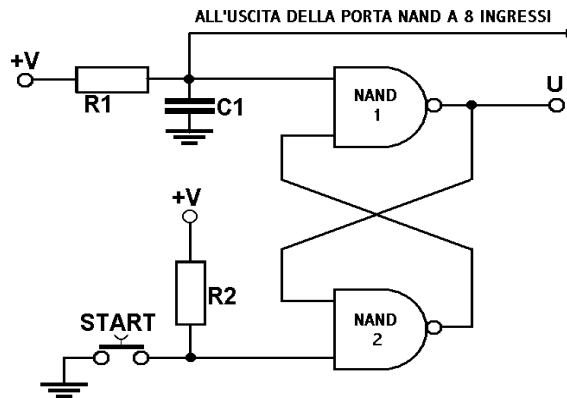
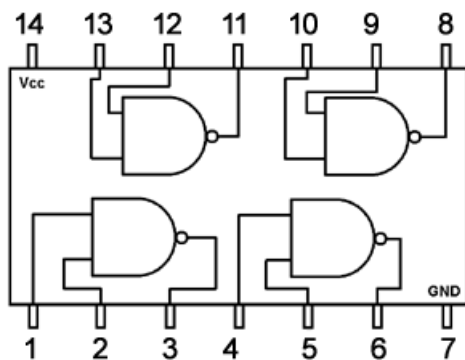


figura 12-9

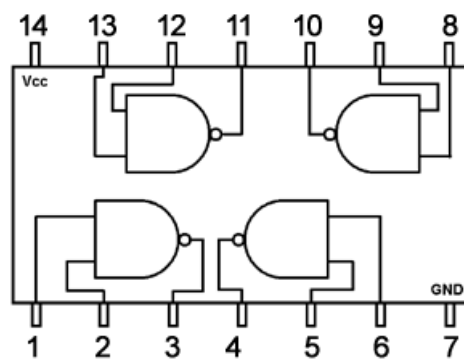
Tale uscita deve essere collegata al famoso piedino 1 del contatore, ed in tal modo gli impulsi in arrivo risultano bloccati. Quando vogliamo far partire il contasecondi, premiamo il pulsante START: il livello L così applicato fa commutare l'uscita di nand2, che invia un livello H all'ingresso di nand1. Ma nand1 ha a livello alto anche l'altro ingresso (poiché nel frattempo C1 si è caricato), ed allora l'uscita U passa a livello L, abilitando l'entrata degli impulsi in arrivo sul contatore. Quando il conteggio termina e l'uscita della porta nand a 8 ingressi passa a livello basso, tale livello torna anche indietro all'ingresso di nand1, commutandola nuovamente in modo da bloccare gli impulsi in ingresso, in attesa di una nuova pressione del pulsante start.

Utilizzazione del segnale di conteggio terminato

Abbiamo detto che quando la condizione di fine conteggio viene raggiunta, la porta Nand trova i suoi otto ingressi a livello H e quindi la sua uscita commuta, passando a livello L. Questo è il segnale che dobbiamo utilizzare per comandare qualunque circuito che ci interessi.



Integrato MM74C00 (4 porte nand a 2 ingressi)



Integrato CD4011 (4 porte nand a 2 ingressi)

Come esempio pratico, immaginiamo di volere generare un segnale acustico una volta terminato il conteggio; possiamo allora usare il circuito visto nel capitolo dedicato al 555, e che viene qui riportato in figura 12-12 per comodità. Sarà sufficiente scollegare il condensatore C1 dalla massa, e collegarlo all'uscita della porta NAND del contasecondi (figura 12-13); quando l'uscita della porta andrà a livello L, fornirà al condensatore C1 il collegamento di massa necessario per il funzionamento dell'oscillatore, che inizierà ad emettere la sua nota dall'altoparlante.

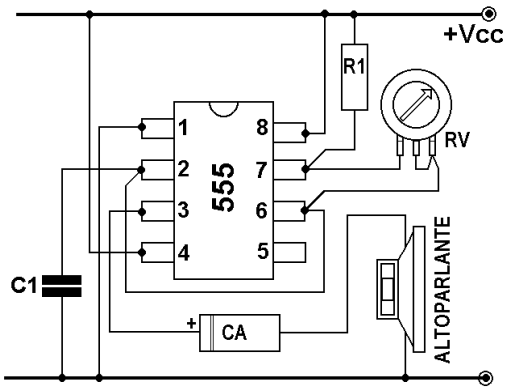


Figura 12-12

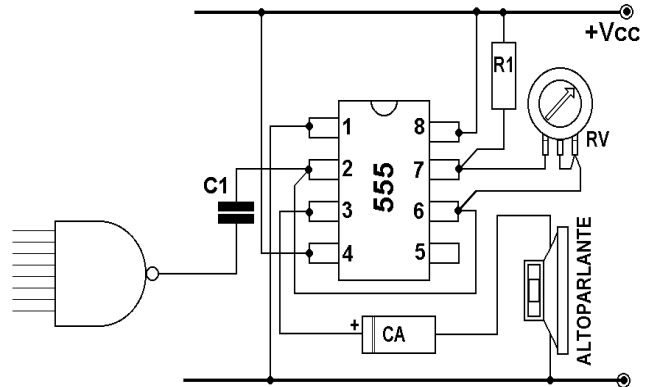


Figura 12-13

CAPITOLO 4

Come si progetta un circuito: “Il dado elettronico”

Scopo di questo capitolo è principalmente quello di fornire un esempio di come sia possibile progettare un circuito, partendo dall’analisi del funzionamento logico globale fino ad arrivare alla realizzazione pratica basata sulle funzioni semplici messe a nostra disposizione dai componenti elettronici presenti sul mercato. Come esempio di applicazione concreta, realizzeremo un “dado elettronico”, ovvero un circuito in grado di simulare il lancio di un dado, fornendo di volta in volta una delle caratteristiche combinazioni che corrispondono alle sei facce del dado classico.

Punto primo: come simulare le facce del dado

I metodi possono essere diversi; probabilmente, quello che meglio rende l’idea è l’uso di 7 diodi led disposti come i punti di un vero dado; accendendo ogni volta i led giusti, come si vede in figura 13-1, è possibile mostrare tutti i punteggi dall’uno al sei.

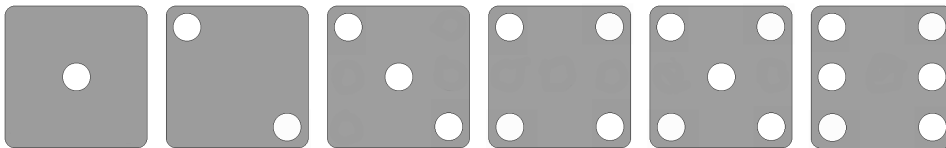


Figura 13-1

Considerazioni logiche

Poiché il dado ha sei facce, ci sono in pratica altrettante diverse combinazioni di led accesi e spenti,

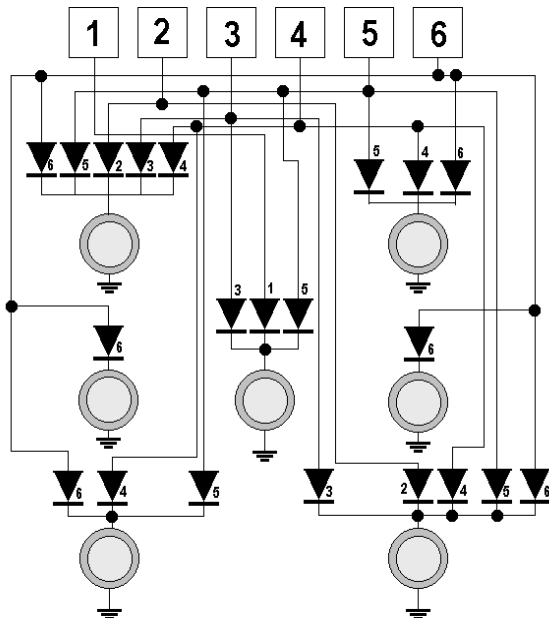


Figura 13-2

che occorre far apparire in sequenza come accade per il dado che rotola. Anche in questo caso le possibilità di realizzazione sono tante: per esempio, si potrebbero collegare insieme tramite diodi tutti i led che devono accendersi per ciascuna delle sei combinazioni; in tal modo, ogni combinazione farebbe capo ad un punto solo cui sarebbe sufficiente applicare una tensione di comando. Tale possibile soluzione è raffigurata in figura 13-2; si vede, per esempio, che applicando tensione al punto 4 si accendono i led ai quattro angoli, così come viene rappresentato il 4 sui comuni dadi da gioco. L’uso dei diodi è necessario poiché, in caso contrario, i vari led verrebbero a trovarsi collegati fra loro, col risultato che si accenderebbero tutti insieme.

Una simile soluzione in effetti funziona, ma non è certo quella che può essere definita una “soluzione elegante”; tutte le volte che si lavora ad un progetto è opportuno non limitarsi ad una soluzione *purché funzioni*, ma cercare piuttosto quel sistema che consenta di ottenere lo stesso risultato col minimo

dispendio di tempo e di componenti. Naturalmente l’uso ottimale di un componente presuppone che se ne conosca dettagliatamente il funzionamento; è necessario quindi disporre dei “data sheet”, cioè di quei fogli che riportano i vari dati tecnici, i grafici relativi alla temporizzazione e gli schemi applicativi.

Oggi, fortunatamente, chi sa usare Internet non deve affrontare né spese né difficoltà, visto che ogni casa produttrice rende liberamente disponibili in rete tutti i manuali tecnici necessari.

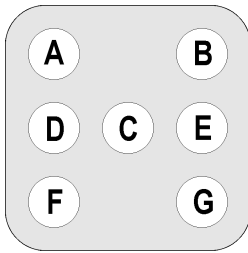


Figura 13-3

Ma vediamo, nel nostro caso, come sia possibile arrivare a una soluzione più razionale.

Consideriamo la disposizione dei punti sulle sei facce del dado; usando le

PUNTEGGIO	LED DA ACCENDERE
UNO	C
DUE	A-G
TRE	A-C-G
QUATTRO	A-B-F-G
CINQUE	A-B-C-F-G
SEI	A-B-D-E-F-G

ad

punti del

lettere riportate in figura 13-3, e facendo riferimento alla figura 13-1, possiamo creare una tabella dove indichiamo i led che devono accendersi per ogni

combinazione. Osservando la tabella notiamo quanto segue:

la coppia di led A-G è sempre accesa, tranne che nel caso dell'UNO

la coppia di led B-F deve accendersi per il QUATTRO, il CINQUE e il SEI

la coppia di led D-E deve accendersi solo per il SEI

Notiamo dunque che i led si accendono a gruppi; possiamo allora pensare di costruire un circuito che, invece di pilotare i sette led uno per uno, comandi l'accensione di uno o più gruppi. Come si è visto, i gruppi possono essere quattro:

la coppia A-G

la coppia B-F

la coppia D-C

il led centrale C.

La scelta dei componenti più adatti

Una volta concepita l'organizzazione logica del circuito, si passa alla ricerca del componente elettronico in grado di realizzare le funzioni necessarie. Poiché a noi occorre accendere dei gruppi in sequenza, una scelta logica è rappresentata da un contatore sequenziale, o "Johnson Counter", come il CD4017, facilmente reperibile sul mercato. Si tratta di un contatore fino a dieci, che prevede pertanto un ingresso per gli impulsi di clock e, nel caso specifico, dieci uscite. Le dieci uscite sono tutte a livello basso, tranne l'uscita 0, che è a livello H. Al primo impulso scatta a livello alto l'uscita 1; il secondo

impulso fa scattare a livello alto l'uscita 2, mentre la precedente, cioè l'uscita 1, torna a livello L. All'impulso 9 scatta a livello alto l'uscita 9, quindi, all'impulso successivo, torna a livello alto l'uscita 0 ed il ciclo ricomincia.

USCITE	IMPULSI									
	0	1	2	3	4	5	6	7	8	9
0	H	L	L	L	L	L	L	L	L	L
1	L	H	L	L	L	L	L	L	L	L
2	L	L	H	L	L	L	L	L	L	L
3	L	L	L	H	L	L	L	L	L	L
4	L	L	L	L	H	L	L	L	L	L
5	L	L	L	L	L	H	L	L	L	L
6	L	L	L	L	L	L	H	L	L	L
7	L	L	L	L	L	L	L	H	L	L
8	L	L	L	L	L	L	L	L	H	L
9	L	L	L	L	L	L	L	L	L	H

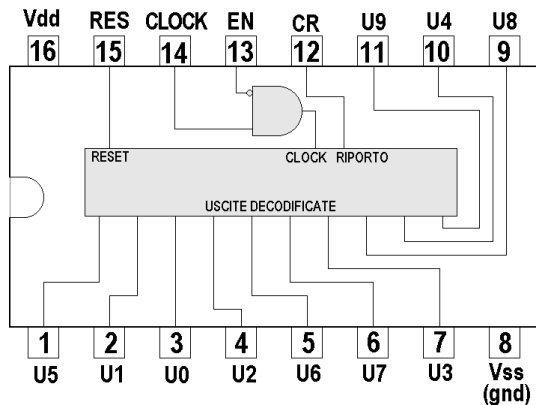


Figura 13-4

cioè riportato alle condizioni di inizio conteggio

un'uscita per il riporto (CR) sul piedino 12; nel caso di più integrati collegati in cascata, tale uscita si usa collegandola all'ingresso di clock dello stadio che segue.

Poiché a noi occorre comandare in sequenza sei combinazioni, mentre il contatore prevede un ciclo di dieci, dobbiamo provvedere a modificarne il comportamento con uno stratagemma; se colleghiamo

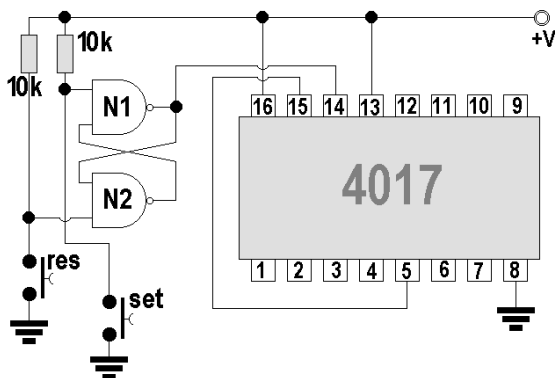


Figura 13-5

contatore un impulso alla volta, potremo analizzare il funzionamento delle varie uscite, compresa l'uscita "carry" del piedino 12 (il riporto).

In pratica, all'accensione prenderemo nota dello stato delle uscite: quali sono a livello alto e quali a

	Successione impulsi						
	0	1	2	3	4	5	6
Uscita 0 (piedino 3)	H	L	L	L	L	L	H
Uscita 1 (piedino 2)	L	H	L	L	L	L	L
Uscita 2 (piedino 4)	L	L	H	L	L	L	L
Uscita 3 (piedino 7)	L	L	L	H	L	L	L
Uscita 4 (piedino 10)	L	L	L	L	H	L	L
Uscita 5 (piedino 1)	L	L	L	L	L	H	L
Carry (piedino 12)	H	H	H	H	H	L	H

zero = livello L; tensione prossima a quella di alimentazione = livello H)

2- è sufficiente controllare lo stato delle uscite da 0 a 5 (le altre non ci interessano).

I risultati che si ottengono effettuando il test descritto sono quelli che si vedono nella tabella che segue, dove è riportato lo stato delle varie uscite all'inizio (prima colonna, impulsi=0), e quindi dopo 1 impulso,

Il funzionamento appena descritto è rappresentato nella tabella della pagina precedente, che poi è la famosa "truth table", o tavola della verità: ci sono tante colonne, ognuna delle quali rappresenta un impulso; per ogni colonna le varie righe indicano la situazione (L o H) di ciascuna uscita.

Lo schema del circuito integrato è riprodotto in figura 13-4; come si vede, oltre alle dieci uscite da U0 ad U9, sono presenti:

un ingresso impulsi (clock) sul piedino 14; il conteggio avanza in corrispondenza del fronte di salita di ogni impulso

un ingresso di abilitazione Enable sul piedino 13; se è a livello basso consente il passaggio degli impulsi di clock, altrimenti li blocca

un ingresso di reset sul piedino 15; applicando un livello alto a tale piedino, il contatore viene resettato,

l'uscita 6 al reset, succede che il livello alto che appare su detta uscita al sesto impulso provoca il reset del contatore, facendo ricominciare il conteggio dall'inizio: è esattamente quello che ci occorre. Costruiamo allora il circuito che si vede in figura 13-5; notiamo che il piedino 5 (corrispondente all'uscita 6) è collegato al piedino 15 (reset). Sulla sinistra è stato aggiunto un flip-flop del tipo set/reset, che servirà per inviare gli impulsi al contatore; quando premiamo il pulsante "set", l'uscita della porta N1 va a livello H ed, essendo collegata all'ingresso di clock del contatore (piedino 14), fa avanzare il conteggio di 1. Per inviare l'impulso successivo dovremo premere il pulsante "reset" e quindi ancora il pulsante "set". In questo modo, facendo arrivare al

livello basso; poi invieremo il primo impulso, e di nuovo prenderemo nota dello stato delle varie uscite. Ripeteremo l'operazione per sei volte, sino a quando il contatore si resetta ed il ciclo ricomincia.

Due precisazioni:

1- lo stato logico delle porte può essere controllato col tester (tensione prossima a

dopo 2 impulsi, ecc.; l'ultima colonna, corrispondente all'impulso 6, è in realtà quella dove il contatore si resetta ed il ciclo ricomincia (ed infatti è uguale alla colonna iniziale). Se continuassimo, ritroveremmo le combinazioni riportate nelle colonne 1, 2, 3, ecc. nella stessa identica successione.

La considerazione forse più notevole che si può fare osservando la tabella è che c'è un'uscita, esattamente quella del riporto (carry, piedino 12), che è sempre a livello H, tranne che in corrispondenza dell'ultimo impulso. Se pensiamo alle facce del dado ed alla divisione in gruppi che abbiamo illustrato nella parte dedicata alle considerazioni logiche, viene subito in mente che c'è uno dei gruppi che si comporta allo stesso modo: è il gruppo A-G (figura 13-3), che risulta sempre presente in tutte le combinazioni, tranne che nel caso dell'uno. Possiamo allora sfruttare questa coincidenza e comandare con l'uscita "carry" il gruppo di led A-G. Poiché l'uscita "carry" è a livello L solo nella colonna 5, faremo corrispondere tale situazione alla faccia del dado con punteggio 1. La colonna successiva, che è poi quella che ci riporta all'inizio, ovvero alla colonna zero, dovrà allora fornirci il punteggio 2; in effetti, poiché il 2 è rappresentato dalla coppia di led A-G che come abbiamo visto è sempre accesa, non occorre provvedere ad altri collegamenti, e quindi non utilizzeremo l'uscita 0 (piedino 3).

Alla colonna successiva (la 1) dovrà corrispondere il punteggio 3; poiché la coppia A-G è già accesa, dovremo solo preoccuparci di far accendere il led centrale C: collegheremo allora tale led all'unica uscita che risulta a livello H nella colonna 1, e cioè l'uscita uno (piedino 2).

Alla situazione della colonna 2 dovrà corrispondere il punteggio 4; essendo già accesa la coppia A-G, dovremo collegare all'uscita 2 (piedino 4) la coppia di led B-F.

Con l'impulso successivo (colonna 3) dovrà apparire il punteggio 5; l'uscita 3 dovrà allora risultare collegata al gruppo B-F ed al led centrale C.

Infine il punteggio 6, che dovrà apparire in corrispondenza della colonna 4: l'uscita a livello H, che è la 4 (piedino 10) dovrà allora comandare l'accensione delle coppie di led B-F, D-E e del led centrale C.

Poiché le uscite del contatore 4017 non sono in grado di fornire la corrente necessaria per l'accensione dei led, occorre usare un transistor per ciascuno dei quattro gruppi di led; abbiamo visto inoltre che uno stesso gruppo di led deve accendersi in più di un caso, e quindi deve essere comandato da più di un'uscita del circuito integrato: occorre allora realizzare i vari collegamenti sempre tramite l'uso di diodi, per evitare che i diversi gruppi si trovino fra loro collegati, accendendosi tutti insieme. Fa eccezione la coppia D-E, che si accende solo per il 6, e quindi può essere collegata senza l'interposizione di un diodo.

In figura 13-6 si vede finalmente lo schema completo del circuito. Come generatore di clock può essere utilizzato uno dei tanti oscillatori che abbiamo visto nei capitoli precedenti, purchè il segnale generato abbia una frequenza compresa più o meno fra 500 hertz e qualche migliaio di hertz. Nella descrizione del funzionamento del 4017 abbiamo visto che il piedino 13 (enable) ha la funzione di abilitare il passaggio del segnale di clock; noi utilizzeremo tale funzione per simulare il lancio del dado. Come si vede il piedino 13 risulta a livello H, essendo collegato alla tensione di alimentazione +V tramite la resistenza RE da 22k Ω ; in tale condizione l'ingresso di clock è bloccato ed il dado è, per così dire, fermo.

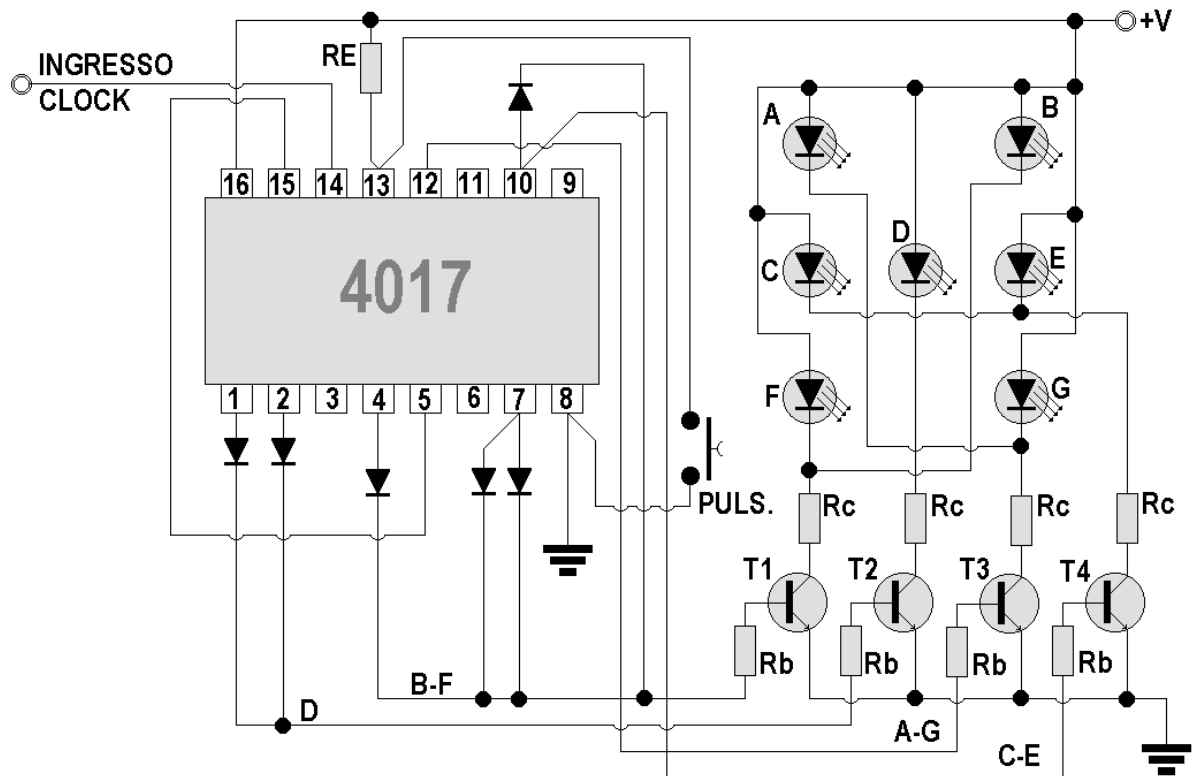


figura 13-6

Premendo il tasto PULS viene abilitato il passaggio degli impulsi provenienti dal generatore di clock, e quindi i led si accendono e spengono rapidissimamente secondo le varie combinazioni, per tutto il tempo che il pulsante rimane premuto; quando si lascia il tasto, i led rimangono a caso in una delle sei possibili configurazioni, simulando l'effetto del dado lanciato. Naturalmente, a causa della velocità elevata, non è possibile decifrare ad occhio le combinazioni che si susseguono mentre si tiene premuto il pulsante: se avete utilizzato un clock a 1000 hz, si tratta di circa 1000 combinazioni ogni secondo! I quattro transistor che comandano i quattro gruppi di led sono degli NPN di bassa potenza; va bene qualunque tipo (BC107 – BC108 – BC109 – BC546 – BC547 – BC548 ecc.). Le resistenze di collettore Rc servono a limitare la corrente nei led; sono tutte da 330 Ω . Le resistenze di base Rb sono quattro resistenze da 56 k Ω . I diodi possono essere di qualunque tipo (1N4001 ecc.). Il pulsante PULS è del tipo normalmente aperto. Il circuito può essere alimentato con la solita pila da 4,5 volt.

CAPITOLO 5

La conversione analogico/digitale

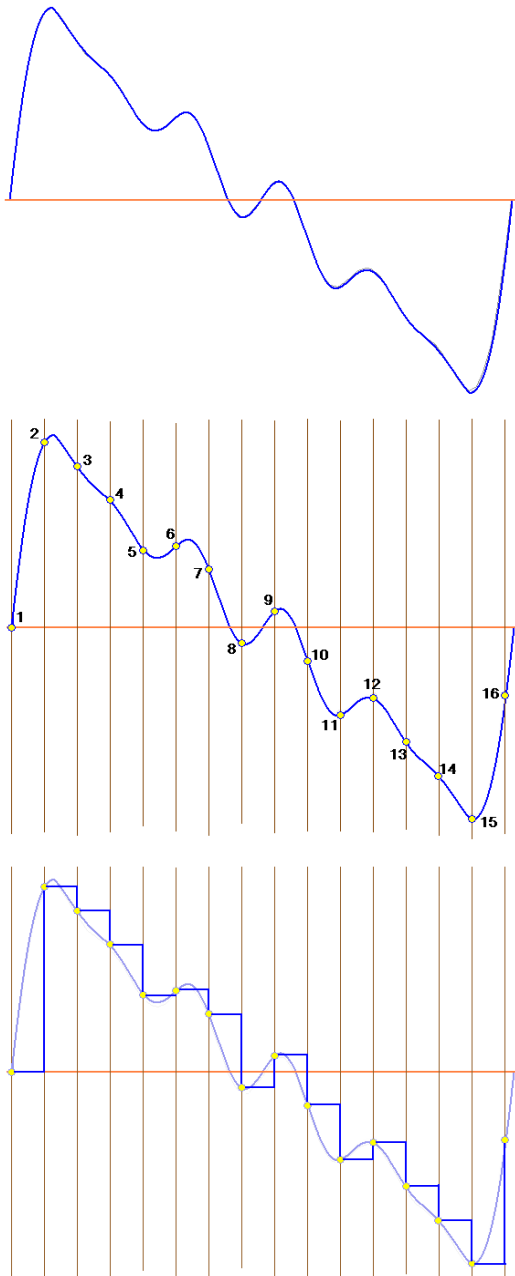


Figura 14-1

Il segnale (figura in alto) viene campionato effettuando letture a intervalli costanti (figura al centro). La linea spezzata della figura in basso rappresenta la ricostruzione del segnale in base ai campioni rilevati.

Abbiamo visto che l'elaborazione digitale prevede l'utilizzo di grandezze esattamente quantificate, espresse come sequenze di bit aventi ciascuno valore uno o zero; tutte le grandezze fisiche, invece, si modificano in natura assumendo da un istante all'altro una serie infinita di valori, con variazioni a volte sono talmente piccole che quasi non riesce possibile misurarle.

I segnali prodotti dai trasduttori fisici (come, per esempio, un microfono) sono essenzialmente di tipo "analogico": si tratta, nel caso specifico, di tensioni che variano con continuità nel tempo, potendo assumere qualsiasi valore. L'elaborazione numerica dei segnali, problema che mai come oggi riveste estrema importanza, richiede di "digitalizzare" il segnale analogico, possibilmente senza introdurre in esso alterazioni e perdita di informazione rispetto al contenuto originale.

La conversione di un segnale analogico in uno digitale può essere suddivisa in due fasi principali: il campionamento e la quantizzazione.

Il campionamento

Il campionamento consiste nell'effettuare delle letture del segnale ad intervalli regolari, memorizzando i valori letti (tecnica detta del "sample and hold"); tali valori, considerati nella loro sequenza temporale, vengono definiti "campioni" del segnale originale. Più piccoli sono gli intervalli fra una lettura e l'altra (ovvero più di frequente viene letto il segnale), migliori sono i risultati ottenuti. In base al teorema di Shannon (o del campionamento), la frequenza con cui viene letto il segnale deve corrispondere almeno al doppio della frequenza più alta che si desidera campionare. I CD audio, per esempio, utilizzano una frequenza di campionamento di 44kHz, il che vuol dire che il segnale originale viene letto 44.000 volte ogni secondo, e che quindi, per ogni secondo di durata dell'audio da riprodurre, sul CD vengono memorizzati 44.000 campioni del segnale originale.

La quantizzazione

La quantizzazione consiste nel passaggio da un valore analogico, ovvero appartenente ad un insieme continuo di punti, ad un valore discreto. Ogni singolo campione viene confrontato con tutta la serie di valori discreti disponibili; si è detto altrove che il numero di valori possibili dipende dal numero di bit utilizzati per rappresentare la grandezza misurata: è evidente che se i livelli tra cui scegliere sono in numero più elevato, diventa possibile ottenere una

corrispondenza più precisa. Si conclude pertanto che è opportuno utilizzare nella trasformazione un numero di bit il più possibile elevato.

Un CD audio utilizza per ogni campione una sequenza di 16 bit, cui corrispondono 65536 combinazioni o livelli; in pratica il valore di ogni campione viene “discretizzato”, scegliendo tra questi 65536 livelli quello che gli risulta più vicino.

L'errore di quantizzazione

Nel passare dal valore *continuo* al valore *discreto* si commette comunque un certo errore, che viene detto *errore di quantizzazione*; tale errore introduce in pratica un rumore indesiderato che si sovrappone al segnale utile. Rapportando il rumore al segnale utile si ottiene quello che viene definito “rapporto segnale-disturbo”; si tratta di una grandezza che esprime la bontà del processo di conversione analogico/digitale, e viene espressa in dB (decibel). Come si è detto, con un maggior numero di bit si dispone di più livelli, e quindi si riduce l'errore di quantizzazione; per essere precisi, ogni bit in più utilizzato nella conversione fa migliorare di 6 decibel il rapporto segnale-rumore. Con 16 bit, per esempio, si ottiene un rapporto segnale-rumore di 101,1 dB.

Per maggior chiarezza, vediamo adesso un esempio pratico di come avviene la trasformazione di un valore *analogico* in valore *discreto*.

Supponiamo che il valore campionato in un certo istante sia una tensione di 177,4 millivolt. Come si può rendere tale valore in forma discreta, ovvero digitale? A nostra disposizione immaginiamo di avere 8 bit, il cui peso raddoppia passando dal precedente al successivo. Possiamo stabilire che il bit più piccolo vale 1; di conseguenza quello successivo avrà valore 2, quello dopo valore 4, quello dopo ancora valore 8, e così via. Ricordate quelle bilance di una volta con due piatti? Si poggiava ciò che si voleva pesare in un piatto e nell'altro si provava ad aggiungere e togliere i pesi di varia misura fino a raggiungere l'equilibrio dei due piatti; la nostra situazione è un po' la stessa: vogliamo vedere quanti bit e di quale peso entrano nel valore da misurare. In base a ciò che abbiamo stabilito prima, i bit di cui disponiamo sono:

128 64 32 16 8 4 2 1

Cominciamo dal più grande, ovvero dal bit che vale 128; il nostro valore è di 177,4 quindi il bit da 128 vi è contenuto. Indichiamo 1 al posto del primo bit, e ricordiamo ciò che rimane del valore iniziale, ovvero 177,4 - 128 e cioè 49,4

Il secondo bit vale 64; tale valore non è contenuto in 49,4; metteremo quindi 0 al posto di tale bit

Il terzo bit vale 32, ed è contenuto in 49,4; mettiamo 1 al suo posto e ricordiamo che la rimanenza è 49,4 - 32, cioè 17,4

Il quarto bit vale 16, ed anch'esso è contenuto nella rimanenza; metteremo quindi 1 al suo posto, ricordando che la rimanenza è adesso di 1,4

I bit di valore 8, 4, e 2 non sono contenuti nella rimanenza, perciò metteremo al loro posto tre zeri

L'ultimo bit vale 1; poiché il valore che rimaneva da compensare era 1,4 metteremo un 1 al posto del bit più piccolo.

Abbiamo così ottenuto il numero 10110001, che esprime in forma binaria il valore 177

Il valore reale da convertire era però 177,4: poiché non disponiamo di bit che valgano meno di 1, non possiamo tenere conto di questa rimanenza e dobbiamo considerare come buono il valore trovato, accettando un errore.

Se, per esempio, invece di 8 bit ne avessimo usati 10, la serie di pesi a nostra disposizione sarebbe diventata:

128 64 32 16 8 4 2 1 0,5 0,25

permettendoci quindi di realizzare una conversione più precisa.

Indice generale

CAPITOLO 1.....	2
Il sistema binario.....	3
Display a sette segmenti e rappresentazione dei numeri	10
CAPITOLO 2.....	13
I flip-flop.....	13
Il flip-flop come divisore di frequenza.....	15
Il flip-flop come contatore.....	15
Oscillatori realizzati con porte logiche.....	18
Tastiera a combinazione.....	19
Serratura elettrica a combinazione.....	22
CAPITOLO 3.....	23
Realizzazione di un contasecondi.....	23
Il generatore di clock.....	23
Contatore binario CD4520.....	23
Circuito di verifica di raggiunto conteggio.....	25
Utilizzazione del segnale di conteggio terminato.....	28
CAPITOLO 4.....	30
Come si progetta un circuito: “Il dado elettronico”	30
Punto primo: come simulare le facce del dado.....	30
Considerazioni logiche.....	30
La scelta dei componenti più adatti.....	31
CAPITOLO 5.....	35
La conversione analogico/digitale.....	35
Il campionamento.....	35
La quantizzazione.....	35
L'errore di quantizzazione.....	36