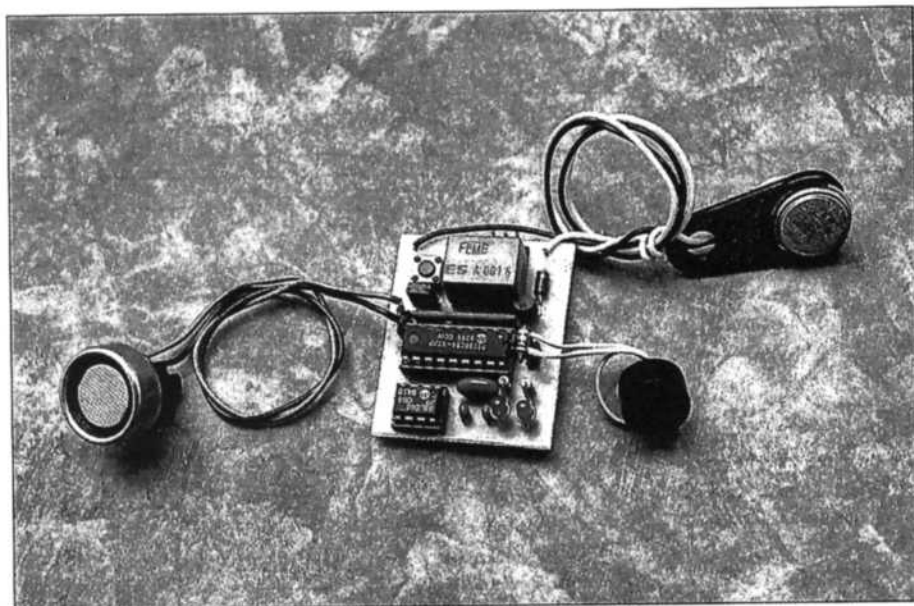


# CHIAVE ELETTRONICA CON CHIP DALLAS

**L'utilizzo di un particolare chip rende questa chiave elettronica sicura e affidabile non solo per il controllo accessi con oltre 250.000 miliardi di combinazioni**

Andrea Sbrana - 1ª parte



**C**hiavi elettroniche per sicurezza e controllo accessi si trovano ormai su tutti i mensili del settore, ma mai ne era stata pubblicata una con caratteristiche altamente professionali grazie all'impiego di chip fabbricati per questo esclusivo scopo.

Le caratteristiche principali sono quelle comuni a tutte le chiavi elettroniche, ma in particolar modo dobbiamo far risaltare due pregi importantissimi: il primo è la possibilità di avere a disposizione oltre 250.000 miliardi di combinazioni, programmate in fabbrica univocamente e, quindi, senza possibilità di copia, il secondo è la distanza operativa tra il circuito fisico e il sensore della chiave, distanza che può raggiungere e talvolta superare i 500 metri, ma con soli due fili!

Ben sapendo che lo scoglio maggiore in un impianto di allarme sono il numero di fili da passare da un punto a un altro, la DALLAS ha ideato un protocollo di comunicazione detto ONE-WIRE cioè un unico filo (ovviamente oltre alla massa) su cui scorrono la tensione di alimentazione e contemporaneamente i dati in arrivo e in partenza dalla chiave (essendo questo protocollo bidirezionale).

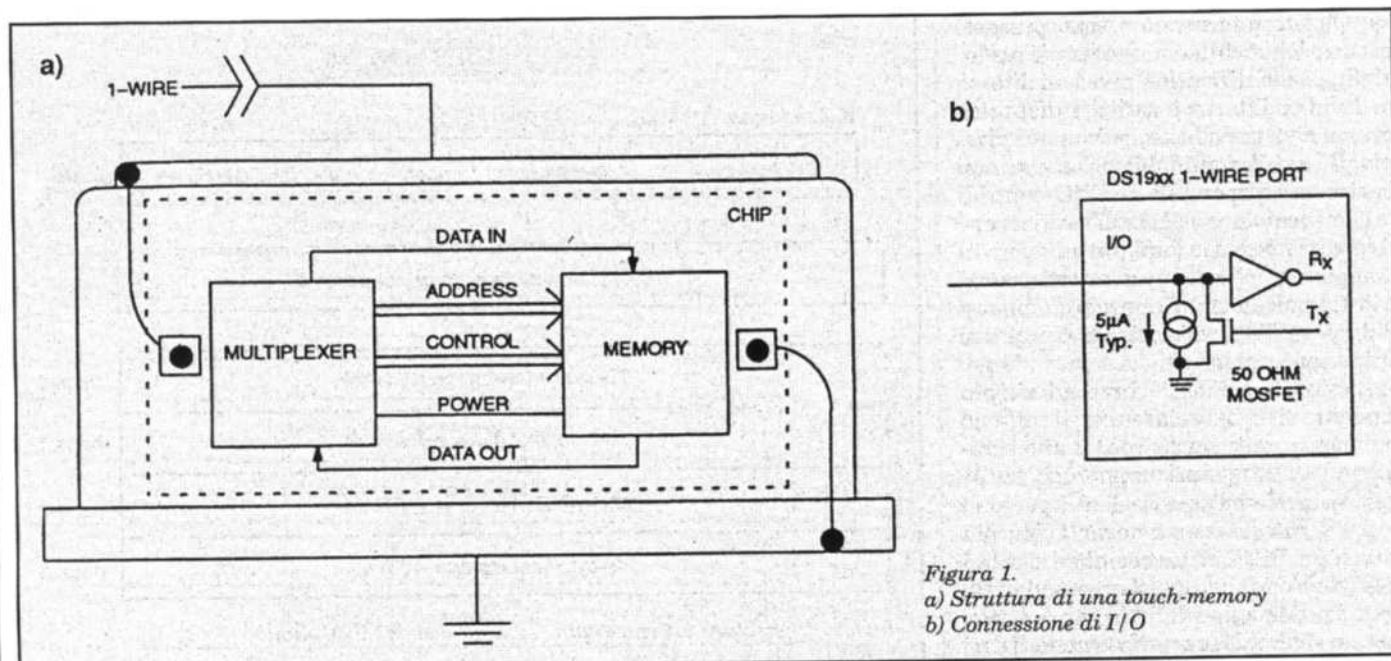


Figura 1.  
a) Struttura di una touch-memory  
b) Connessione di I/O

Oltre a questi due fattori già notevolmente importanti, dobbiamo accennare anche alla non necessità di un contenitore per la chiave stessa, riducendo tempi e costi di produzione.

Inoltre, vedremo che esiste tutta una famiglia di chiavi dette TOUCH MEMORY, alcune delle quali hanno al loro interno anche RAM tamponate e contatti per realizzare orologi, controllo accessi, apparecchiature a tempo.

## La famiglia delle touch-memory

Ci sono diversi tipi di touch-memory in commercio con strutture adatte a ogni esigenza. In Figura 1-a, possiamo vedere la struttura generale di una di queste: due blocchi, uno la memoria e uno il multiplexer, che interagiscono tra di loro e che dialogano con l'esterno attraverso il bus ONE-WIRE.

In Figura 1-b vediamo, infatti, la porta di I/O interna alle touch-memory.

Per collegarsi con una touch-memory, sono possibili due soluzioni: quella diretta (Figura 2-a) oppure quella indiretta (Figura 2-b). La prima è accessibile se si dispone di un controller che possa rendere bidirezionale una sua porta, la seconda è più sicura su collegamenti a lunga distanza.

In Figura 3, invece, troviamo come sono strutturate internamente le varie touch-memory, a partire dalla DS1990A che è quella impiegata per la nostra chiave: è la più semplice di tutte, in quanto al suo interno ha soltanto un codice di 6 byte, un CRC e un byte per distinguere la famiglia di appartenenza. Queste informazioni sono presenti in tutte le touch-memory, perché essenziali per identificarle univocamente.

Il codice interno è scritto in fabbrica con un sistema a laser, quindi non ricopiabile e molto affidabile. Come se non bastasse, è disponibile un CRC (controllo di errore) che consente di verificare se il codice letto più la famiglia sono corretti oppure no. Questa opzione è di grande utilità, poiché è facile leggere male un dato data l'alta velocità con cui questi transitano sul bus.

La memoria DS1991, invece, ha in più 4 pagine di 64 byte ciascuna, in cui sono memorizzate tre password e uno scratchpad. Questa soluzione può essere utile per permettere l'accesso di una persona a più strutture senza dover avere più chiavi. La DS1992, invece, oltre ai primi 64 byte ROM, ha uno scratchpad di 32 byte e poi 4 pagine di RAM non volatile, per consentire la memorizzazione di dati

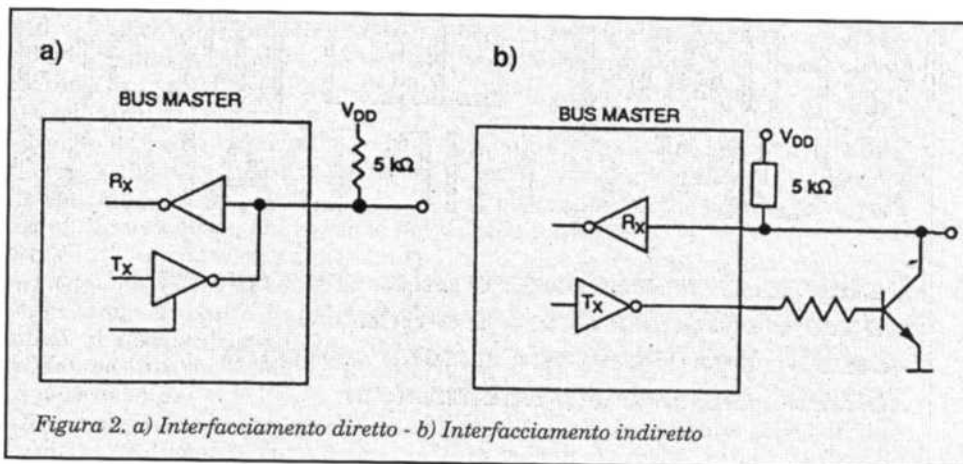


Figura 2. a) Interfacciamento diretto - b) Interfacciamento indiretto

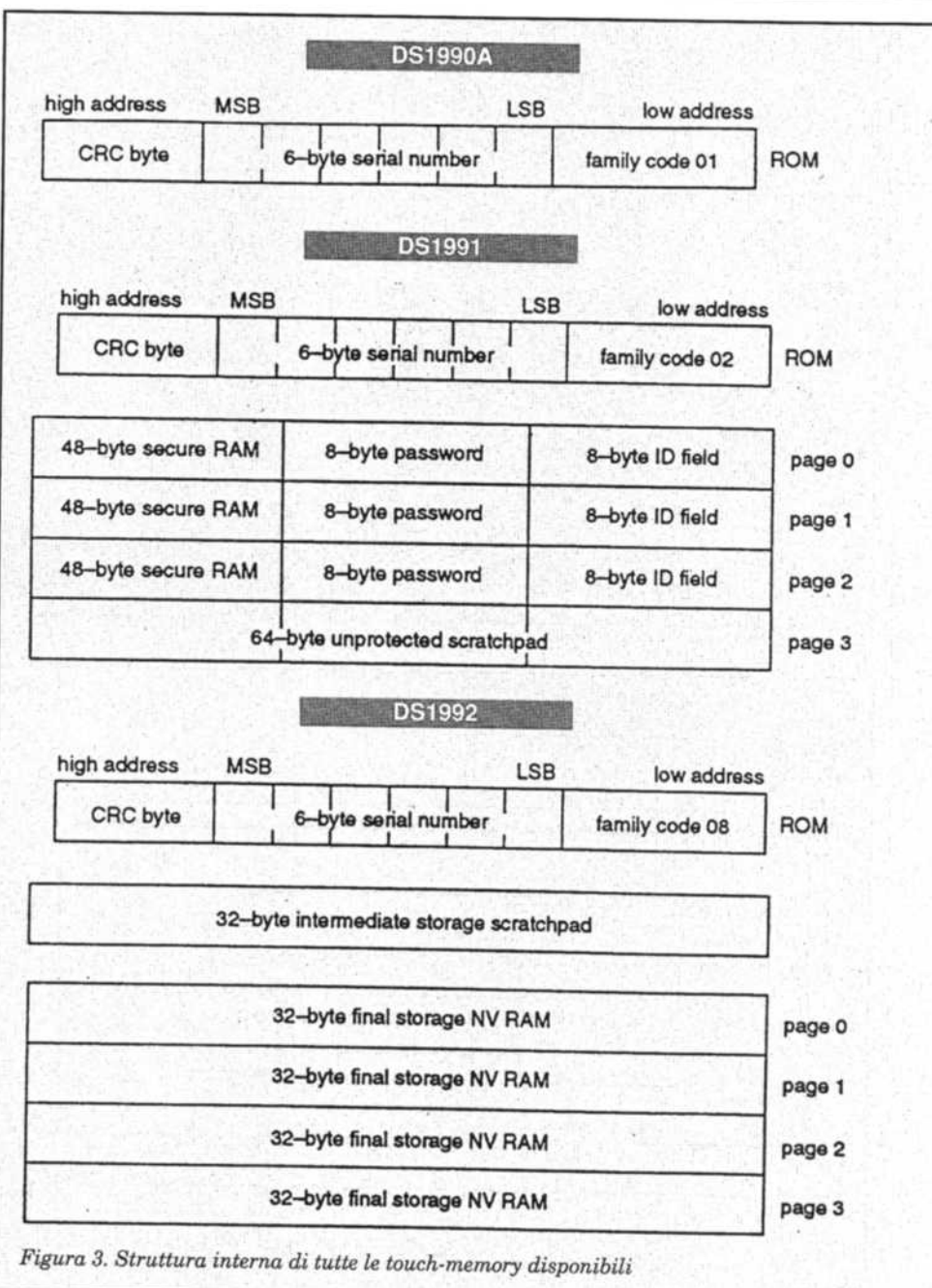
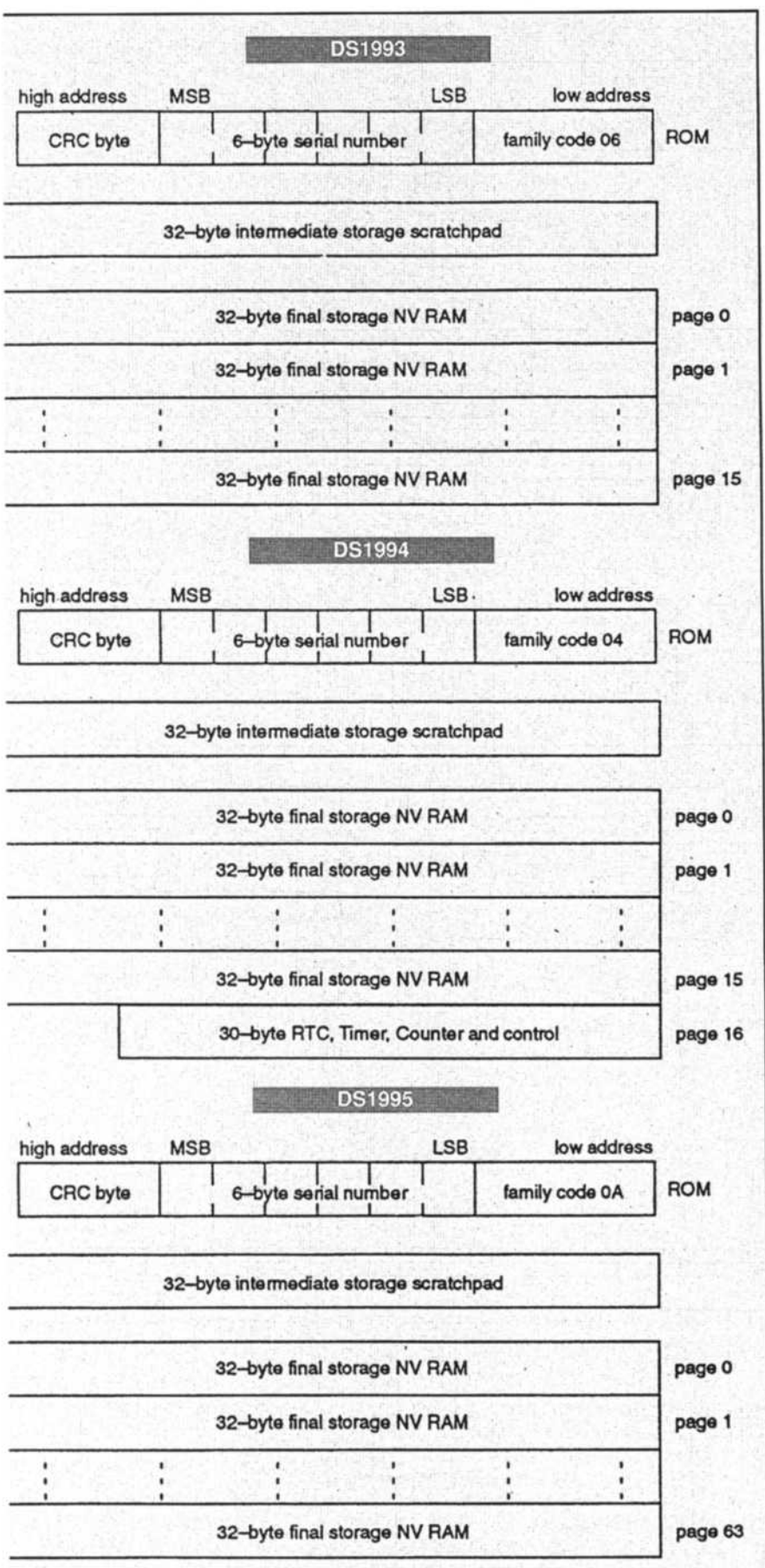


Figura 3. Struttura interna di tutte le touch-memory disponibili



direttamente sulla chiave. All'interno della chiave stessa è presente una batteria al litio che ne garantisce una lunghissima autonomia.

Questo tipo di chiave ha molteplici applicazioni, ma la più immediata è la memorizzazione rapida di piccole quantità di dati da rileggere poi senza fretta con un sistema dedicato.

La DS1993 è identica strutturalmente alla precedente, ma possiede 16 pagine di RAM non volatile per una maggiore quantità di dati.

Ugualmente la DS1995 ha una capacità di 64 pagine.

La DS1994 invece è identica alla 1993 per quanto riguarda l'area di memoria, ma possiede in più 30 byte adibiti a contatori, timer e orologi: in pratica con questo tipo di memoria è possibile realizzare orologi, timer, circuiti a tempo, controlli accesso programmabili ecc.

## Il protocollo ONE-WIRE

La grande flessibilità di queste memorie, è dovuta in gran parte al protocollo ONE-WIRE adottato dalla DAL-LAS: ci sono soltanto due fili. Uno di questi è la massa, l'altro trasporta la tensione di alimentazione e i segnali che controllano la memoria.

In Figura 4 possiamo vedere le tempistiche principali per poter dialogare con le touch-memory. In Figura 4-a vediamo come fare a inviare un 1 alla memoria: abbiamo un tempo detto TIME-SLOT che deve necessariamente essere compreso tra 60 e 120 microsecondi mentre il tempo in cui viene scritto l'1 deve essere compreso tra 1 e 15 microsecondi.

Non rispettare questi tempi, significa non riuscire a far funzionare questa chiave. In Figura 4-b vediamo invece che per scrivere uno 0 è necessario un tempo superiore ai 60 microsecondi ma inferiore ai 120. Se, invece, vogliamo leggere dei dati dalla memoria, dobbiamo sfruttare il time-diagram di Figura 4-c: poiché la comunicazione avviene su un filo solo, per la sincronizzazione è necessario che il master invii un segnale detto "di sincronismo" e che poi si metta in ricezione con le tempistiche riportate. Questa sincronizzazione è necessaria per ogni bit da leggere.

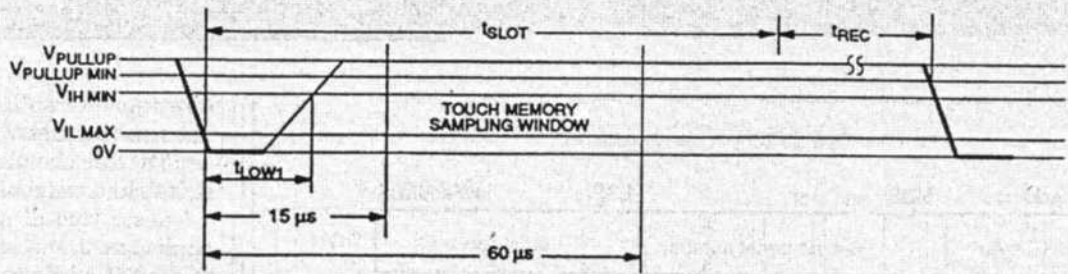
Infine, in Figura 4-d è possibile capire come avviene la rilevazione della chiave.

Infatti il master sta sempre a interrogare la chiave inviandogli un impulso di reset maggiore di 480 microsecondi. Se la chiave è presente, risponde entro 480 microsecondi, altrimenti il ciclo si ripete.

Figura 4.

a)

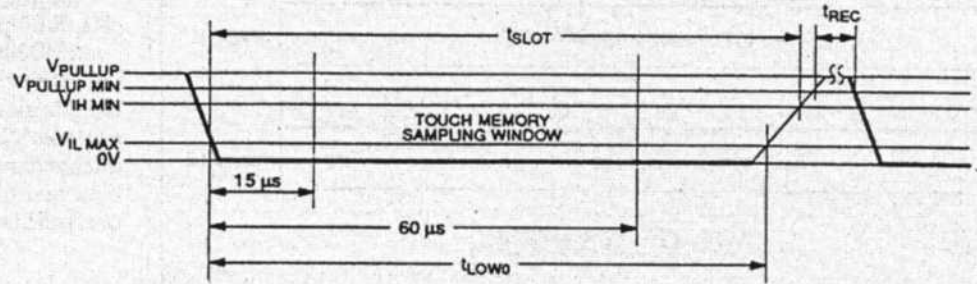
Scrittura di un 1



$60 \mu s \leq t_{SLOT} < 120 \mu s$   
 $1 \mu s \leq t_{LOW1} < 15 \mu s$   
 $1 \mu s \leq t_{REC} < \infty$

b)

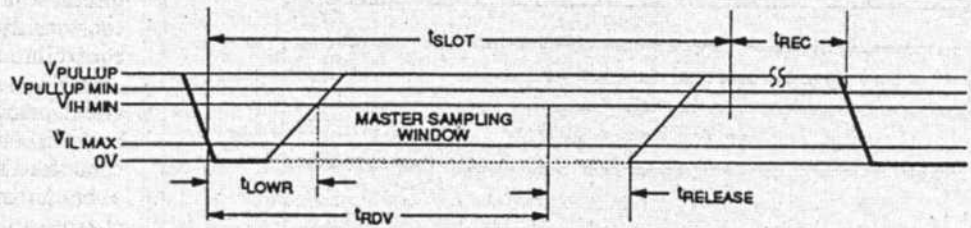
Scrittura di uno 0



$60 \mu s \leq t_{LOW0} < t_{SLOT} < 120 \mu s$   
 $1 \mu s \leq t_{REC} < \infty$

c)

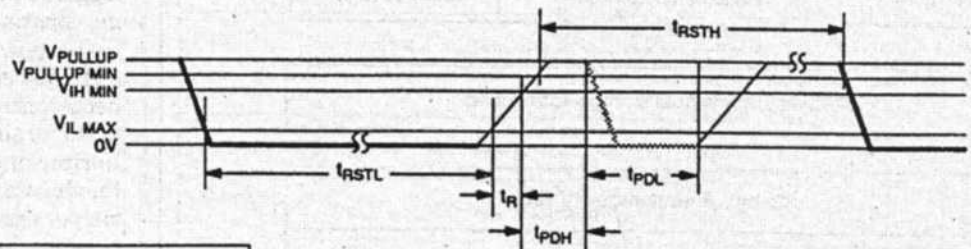
Lettura di dati



$60 \mu s \leq t_{SLOT} < 120 \mu s$   
 $1 \mu s \leq t_{LOWR} < 15 \mu s$   
 $0 \leq t_{RELEASE} < 45 \mu s$   
 $1 \mu s \leq t_{REC} < \infty$   
 $t_{RDV} = 15 \mu s$

d)

Impulso di reset e rilevazione presenza



$480 \mu s \leq t_{RSTL} < \infty$   
 $480 \mu s \leq t_{RSTH} < \infty$  (Includes recovery time)  
 $15 \mu s \leq t_{PDH} < 60 \mu s$   
 $60 \mu s \leq t_{PDL} < 240 \mu s$



# CHIAVE ELETTRONICA CON CHIP DALLAS

**Il segreto del successo che questa serratura elettronica sta ottenendo risiede, oltre che nella sicurezza garantita dall'alto numero di combinazioni, soprattutto nel fatto che utilizza è il primo integrato realizzato a forma di... chiave**

Andrea Sbrana - 2ª parte



**L**o scorso mese abbiamo presentato le memorie della DALLAS che prendono il nome di touch-memory e adesso ne vediamo una semplice applicazione nel campo della sicurezza.

Con riferimento alla Figura 1 dell'articolo pubblicato il mese scorso, possiamo notare che il circuito elettrico è ridotto all'essenziale: un integrato che dialoga con la chiave e una memoria di tipo non volatile che memorizza fino a 8 chiavi, un numero più che sufficiente per piccole e medie applicazioni. Sono presenti due Led per la segnalazione ottica dello stato della chiave e un buzzer per una segnalazione acustica.

L'uscita della chiave viene affidata a un relè a deviazione monopolare.

Sul circuito sono presenti un pulsante per la programmazione delle chiavi e un jumper per la selezione del modo di funzionamento: impulsivo o set/reset.

## Il software del PIC

Passiamo allora alla Figura 6 e andiamo ad analizzare il diagramma a blocchi del software di gestione inserito nel microcontroller.

L'inizializzazione dei registri all'atto della prima accensione è ormai una fase obbligatoria per qualsiasi applicazione.

Poi il microcontroller passa a testare in continuazione la pressione del pulsante P1 oppure la presenza della chiave.

Per quest'ultimo test, il controller deve

rispettare alcuni parametri che specificheremo in seguito. Se viene rilevata la pressione del pulsante, la prima cosa da fare è la verifica dello stato della chiave, in quanto non è permesso riprogrammarla se il relè è nello stato di attività, nel qual caso si torna allo stato iniziale.

Se il relè è spento, il Led verde inizia a lampeggiare fino a quando una prima chiave non viene inserita. Una volta riconosciuta la prima chiave, i due Led si spengono in attesa della successiva.

All'inserimento della successiva si avrà un lampeggio del Led rosso e poi di nuovo si attenderà la terza chiave e così via fino alla ottava.

Apriamo una piccola parentesi per chiarire che anche se volete memorizzare una sola chiave, dovrete inserire la stessa per le otto volte necessarie alla fase di programmazione, in quanto il microcontroller non può sapere che voi ne volete memorizzare una sola.

Lo stesso dicasi ad esempio per tre chiavi: si memorizzerà la prima, poi la seconda e poi la terza per sei volte.

Proseguiamo allora con la Figura 6: dopo la fase di memorizzazione delle chiavi, il Led rosso si spegnerà, quello verde si accenderà e il controller tornerà al test iniziale.

Supponiamo ora di inserire una chiave: il PIC la leggerà e confronterà il suo codice con quello inserito nella prima locazione della EEPROM esterna. Se questo non coincide, il PIC prova con la seconda locazione, poi con la terza fino ad arrivare alla ottava. Se anche questa dà esito negativo, il buzzer emetterà quattro beep intermittenti e poi il PIC si rimetterà nello stato di test iniziale.

Al contrario, se il codice della chiave viene riconosciuto come valido, il controller va a vedere lo stato del jumper J1: se questo è stato predisposto per il funzionamento impulsivo, si sentiranno 2 beep intermittenti e il relè si ecciterà per circa 1 secondo, mentre se è stato predisposto per il funzionamento di tipo set/reset, si sentirà un solo beep e il relè muterà il suo stato qualunque esso sia stato in precedenza.

Al termine di queste due sequenze si passa all'attesa del rilascio della chiave e successivamente al test iniziale.

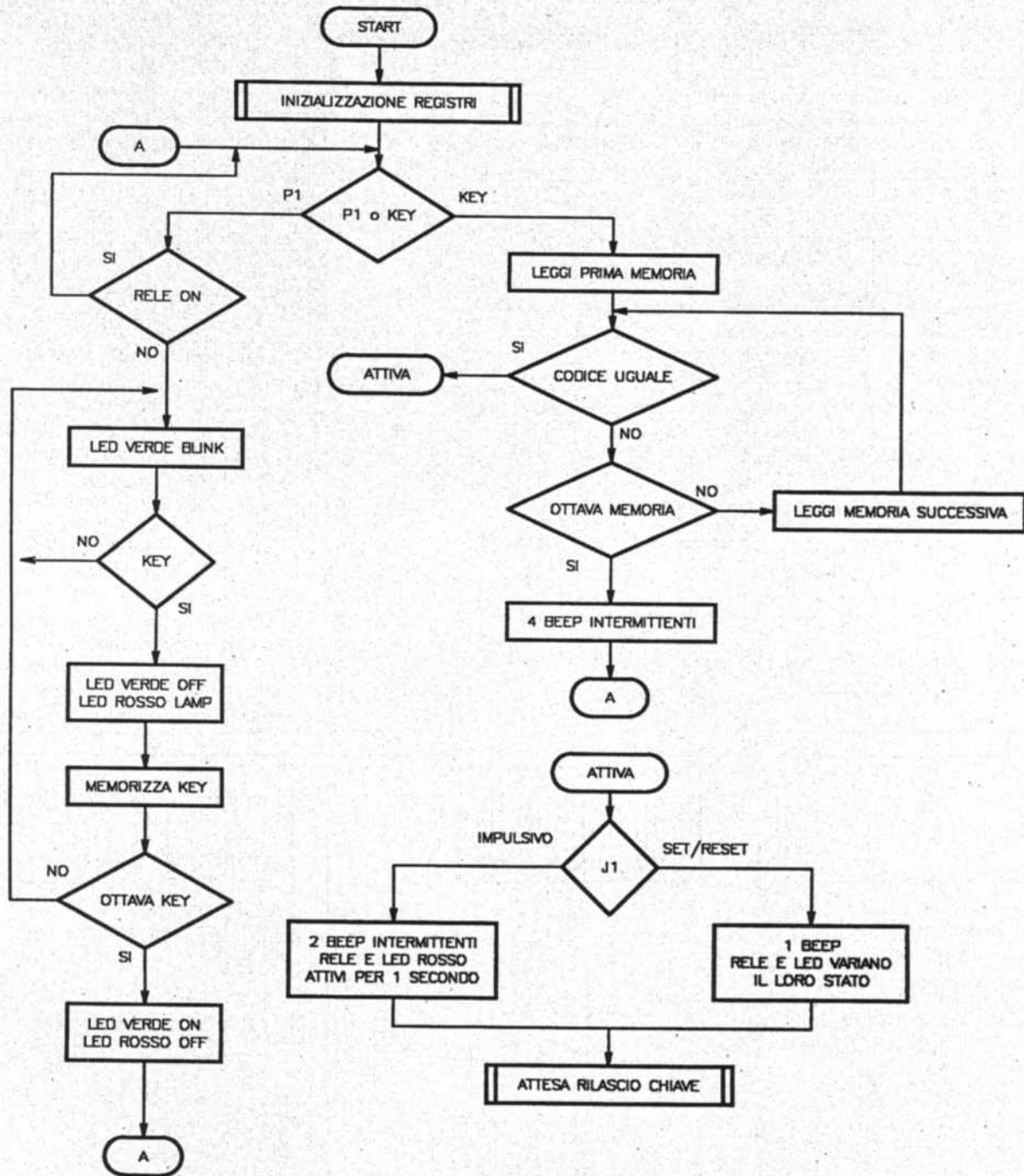


Figura 6. Diagramma a blocchi del software inserito nel PIC

**Il dialogo con la chiave**

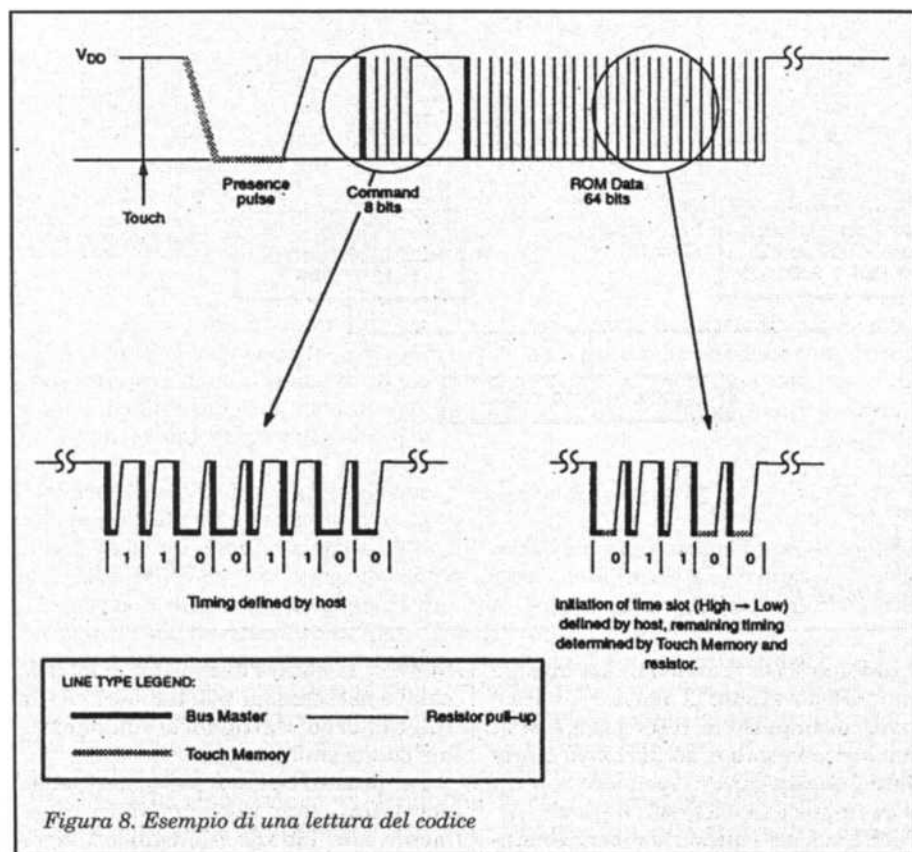
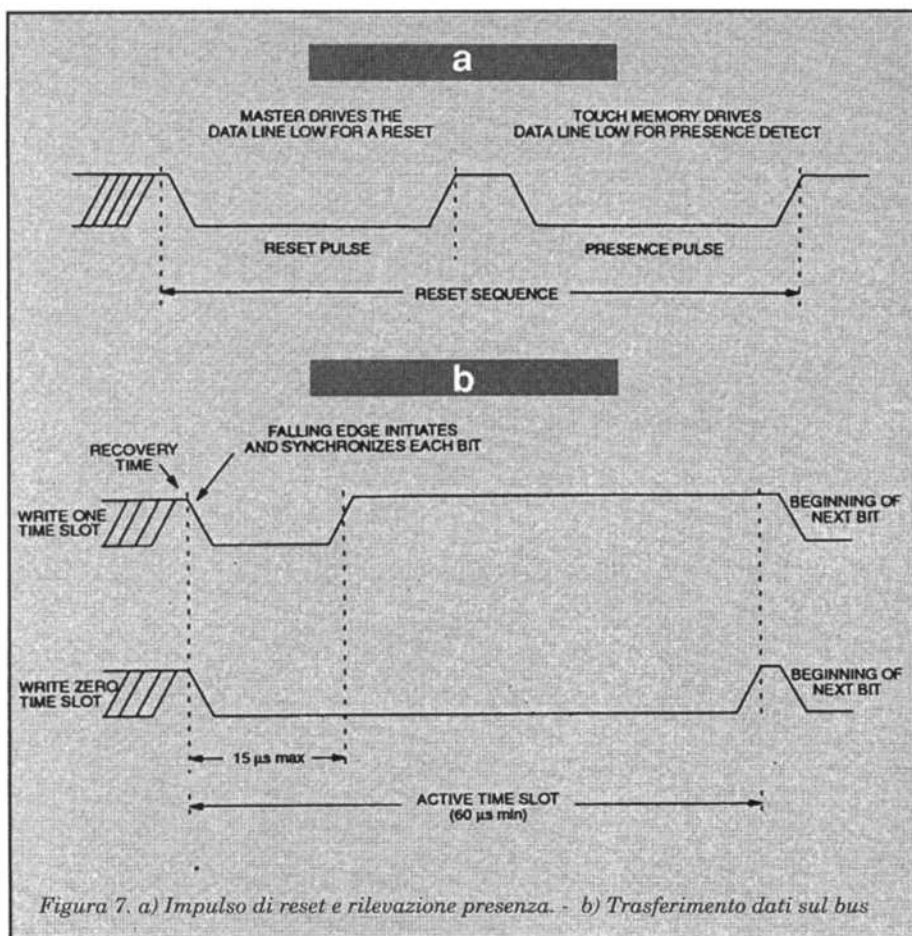
Come abbiamo accennato il mese scorso, il protocollo di comunicazione con la chiave è abbastanza articolato, anche se la chiave stessa è del tipo più semplice, ovvero una DS1990A.

In Figura 7-a vediamo il test iniziale come viene svolto: il master (il PIC) invia un impulso di reset maggiore di 480 microsecondi e, se la chiave è presente, questa deve rispondere con un altro impulso di adeguata durata.

Il PIC sfrutta questo sistema per sta-

bilire se la chiave è inserita, mentre la chiave necessita di tale impulso per un reset interno. Entrambi si sincronizzano su questo impulso.

Per quanto riguarda il trasferimento dei dati, invece, dobbiamo fare riferimento alla Figura 7-b: ogni bit viene



sincronizzato sul fronte discendente e la differenza tra un 1 e uno 0 logico sta nella durata dell'impulso a livello basso.

In Figura 8 potete vedere un esempio di lettura del codice in ROM: dopo l'impulso di reset e di presenza, il master invia alla chiave il comando di lettura della ROM e, successivamente, si sincronizza a ogni bit di risposta da parte della chiave stessa.

Questo meccanismo è valido per la lettura di tutti i tipi di chiavi della DALLAS. Ovviamente a ogni tipo di chiave, corrisponderà un "family code" diverso.

## Montaggio e collaudo

La realizzazione della chiave completa è semplicissima, rispettando però le regole fondamentali dei montaggi elettronici.

Per prima cosa è consigliabile usufruire della traccia proposta in Figura 9 per quanto riguarda il circuito stampato. Poi è bene inserire degli zoccoli sia per IC1 che per IC2.

Dovrete fare attenzione ai componenti polarizzati e anche alla rete resistiva.

A proposito di queste reti resistive, molti lettori ci informano che la loro reperibilità a volte è difficoltosa, quindi

## ELENCO COMPONENTI

### Semiconduttori

- IC1: PIC16C54XT programmato (0337/259730)
- IC2: 93LC46
- T1: BC337
- D1: 1N4001
- LED1: Led verde 3 mm
- LED2: Led rosso 3 mm

### Resistori

- R1: 10 kΩ
- R2, R3: 220 Ω
- Rp1: Rete resistiva 4,7 kΩ 1+8

### Condensatori

- C1, C2: 33 pF
- C3: 100 nF

### Varie

- Q1: Oscillatore ceramico 3.58 MHz
- Piezo: Buzzer piezoelettrico

