

M.M.Electronics - <http://www.mmetft.it>



Michele Marino - mmelectronics@mmetft.it

Controllo analogico di una tastiera a matrice

V 0.1

Dicembre 2007

INFORMATIVA

Come prescritto dall'art. 1, comma 1, della legge 21 maggio 2004 n.128, l'autore avvisa di aver assolto, per la seguente opera dell'ingegno, a tutti gli obblighi della legge 22 Aprile del 1941 n. 633, sulla tutela del diritto d'autore. Tutti i diritti di questa opera sono riservati. Ogni riproduzione ed ogni altra forma di diffusione al pubblico dell'opera, o parte di essa, senza un'autorizzazione scritta dell'autore, rappresenta una violazione della legge che tutela il diritto d'autore, in particolare non ne è consentito un utilizzo per trarne profitto. La mancata osservanza della legge 22 Aprile del 1941 n. 633 è perseguibile con la reclusione o sanzione pecuniaria, come descritto al Titolo III, Capo III, Sezione II. A norma dell'art. 70 è comunque consentito, per scopi di critica o discussione, il riassunto e la citazione, accompagnati dalla menzione del titolo dell'opera e dal nome dell'autore.

AVVERTENZE

Chiunque decida di far uso delle nozioni riportate nella seguente opera o decida di realizzare i circuiti proposti, è tenuto pertanto a prestare la massima attenzione in osservanza alle normative in vigore sulla sicurezza.

L'autore declina ogni responsabilità per eventuali danni causati a persone, animali o cose derivante dall'utilizzo diretto o indiretto del materiale, dei dispositivi o del software presentati nella seguente opera.

Si fa inoltre presente che quanto riportato viene fornito così com'è, a solo scopo didattico e formativo, senza garanzia alcuna della sua correttezza.

L'autore ringrazia anticipatamente per la segnalazione di ogni errore.

Indice

| | | |
|----------|---|-----------|
| 1 | Introduzione | 4 |
| 2 | Panoramica sul controllo digitale standard | 4 |
| 3 | Il convertitore A/D | 4 |
| 4 | La tastiera a matrice | 8 |
| | Bibliografia | 12 |

Elenco delle figure

| | | |
|----|--|----|
| 1 | Schema elettrico di una tastiera a matrice 4x4 | 4 |
| 2 | Layout della tastiera a matrice 4x4 | 4 |
| 3 | Modello analogico di ingresso del convertitore A/D | 6 |
| 4 | Inizializzazione registri | 7 |
| 5 | Struttura del registro ADCON1 | 7 |
| 9 | Schema elettrico della tastiera | 8 |
| 6 | Inizializzazione convertitore A/D | 9 |
| 7 | Struttura del registro ADCON0 | 9 |
| 8 | Struttura del registro PIR1 | 9 |
| 10 | Dichiarazione registri | 10 |
| 11 | Codice di controllo tasto "Ok" | 10 |

Elenco delle tabelle

1 Introduzione

In questo articolo introduco brevemente un metodo per controllare una tastiera a matrice utilizzando una sola linea del PIC anziché otto. Il controllo è completamente analogico e fa quindi uso del convertitore A/D interno al PIC. L'idea nacque qualche anno fa a causa della mancanza di linee di I/O in un progetto abbastanza complesso. La prima parte riguarda una panoramica sul convertitore A/D a 10 bit presente nel PIC e successivamente il controllo della tastiera a matrice.

2 Panoramica sul controllo digitale standard

Il controllo standard prevede l'utilizzo di 8 linee digitali delle quali, 4 per il controllo delle righe e 4 per il controllo delle colonne. La figura 1 mostra lo schema elettrico di una tastiera a matrice 4x4.

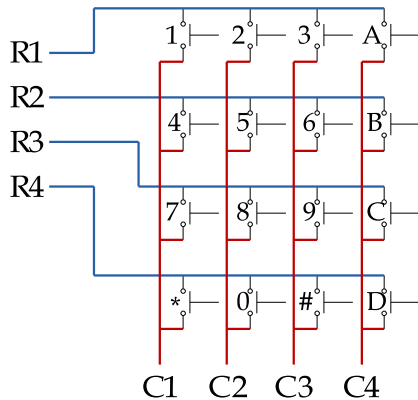


Figura 1: Schema elettrico di una tastiera a matrice 4x4

La gestione avviene eseguendo un polling sulle colonne. Supponiamo di porre un livello logico alto sulla riga R1 po-

nendo R2,R3,R4 a zero logico. A questo punto, si va ad eseguire una lettura consecutiva delle colonne C1..C4. Se per esempio viene premuto il tasto "1", sulla colonna C1 sarà presente un livello logico alto in quanto risulta cortocircuitata con la linea R1. Se viene premuto il tasto "3", il livello logico alto sarà presente sulla linea C3 e così via. La scansione procede fissando in sequenza i livelli logici alti sulle righe ad andando a fare un polling sulle colonne. Una volta individuato il tasto premuto si va ad eseguire la porzione di codice associata ovvero, la funzione relativa al tasto pigiato.

Infine, la figura 2 mostra il layout della tastiera a matrice 4x4.



Figura 2: Layout della tastiera a matrice 4x4

3 Il convertitore A/D

Il controllo della tastiera viene effettuato utilizzando il convertitore A/D (Analog to Digital) a 10 bit presente all'interno del PIC16F876. Come noto, le operazioni di conversione consistono nel porre una

corrispondenza tra stringhe di bit e valori analogici. Naturalmente la struttura di tale corrispondenza non è univoca e trova un riflesso nell'accuratezza delle operazioni. I parametri fondamentali di ogni elaborazione e quindi di ogni conversione sono i seguenti:

- Prestazione al rumore
- Dinamica dei segnali
- Capacità elaborativa

Il primo parametro può essere riguardato in termini di probabilità di errore per bit (Bit Error Rate - BER); il secondo sarà fissato dalla dinamica della parte analogica degli apparati convertitori, mentre il terzo parametro, dal numero di bit al secondo (Bit Rate - BR) che il sistema è in grado di processare. Il convertitore A/D presente nel PIC16F876 è un convertitore ad approssimazioni successive e nella figura 3 viene riportato il modello analogico di ingresso di ogni linea del convertitore presente nel PIC.

Per avere la massima accuratezza nella conversione è necessario caricare completamente la capacità di mantenimento (C_{HOLD}) al valore di tensione presente sul canale di ingresso da convertire. Come si può vedere dallo schema di figura 3, la resistenza di sorgente R_S e la resistenza interna dello switch di campionamento influenzano direttamente il tempo di caricamento della capacità prima citata (C_{HOLD}). Il costruttore raccomanda il valore massimo di $10K\Omega$ per la resistenza R_S di sorgente. Naturalmente più basso è il valore di questa minore sarà il tempo di acquisizione. Ovviamente, dopo aver selezionato il canale di ingresso, questa acquisizione deve essere effettuata prima che la conversione venga avviata.

Per calcolare il tempo minimo di acquisizione, può essere utilizzata la seguente equazione¹:

$$\begin{aligned} T_{ACQ} &= T_{AMP} + T_C + T_{COFF} = \\ &= 2\mu s + T_C + \\ &\quad + \left[(Temp - 25^\circ C) (0.05 \frac{\mu s}{^\circ C}) \right] \end{aligned} \quad (1)$$

dove T_{AMP} rappresenta il tempo di settling dell'amplificatore, T_C il tempo di caricamento della capacità di mantenimento e, T_{COFF} il coefficiente di temperatura. La carica della capacità di hold, impiega un tempo pari a:

$$\begin{aligned} T_C &= C_{HOLD} (R_{IC} + R_{SS} + R_S) \cdot \\ &\quad \cdot \ln \left(\frac{1}{2047} \right) \\ &= 120pF (1K\Omega + 7K\Omega + 10K\Omega) \cdot \\ &\quad \cdot \ln(0.0004885) = 16.47\mu s \end{aligned} \quad (2)$$

Il tempo di acquisizione è quindi dato da:

$$\begin{aligned} T_{ACQ} &= 2\mu s + 16.47\mu s + \\ &\quad + \left[(50^\circ C - 25^\circ C) \left(0.05 \frac{\mu s}{^\circ C} \right) \right] = \\ &= 19.72\mu s \end{aligned} \quad (3)$$

Questa equazione assume come errore il valore pari a 1/2 LSB (Least Significant Bit - 1024 passi per la conversione). L'errore 1/2 LSB è il massimo consentito affinché la conversione A/D rispecchi la risoluzione specificata.

¹I calcoli si riferiscono al caso in cui la resistenza di sorgente sia di $10K\Omega$ e la temperatura di esercizio di $50^\circ C$.

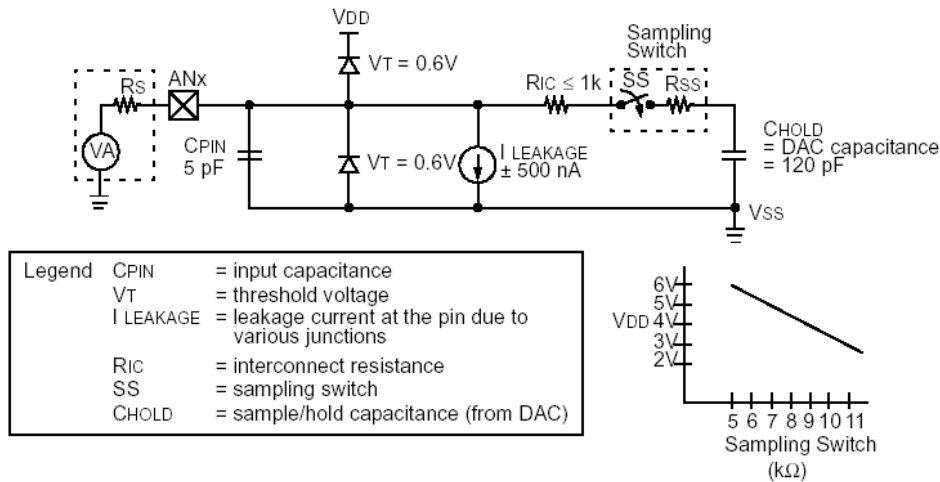


Figura 3: Modello analogico di ingresso del convertitore A/D

I registri interessati nella conversione A/D sono i registri ADCON0, ADCON1, ADRESH e PIR1 e la conversione avviene seguendo i seguenti passi:

1. Configurazione del modulo A/D:

- Configurazione dei pin come ingressi analogici o come ingressi/uscite digitali e relative tensioni di riferimento mediante i bit 0,1,2,3 del registro ADCON1
- Selezione del canale di ingresso mediante i bit 3,4,5 del registro ADCON0
- Selezione della frequenza di conversione (clock) mediante il registro ADCON0
- Accensione del modulo di conversione A/D mediante il settaggio del bit 0 del registro ADCON0

2. Attesa per il tempo di acquisizione

3. Inizio della conversione:

- Settaggio del bit 2 ($\overline{GO/DONE}$) del registro ADCON0

4. Attesa fino al completamento della conversione

5. Lettura del risultato contenuto nel registro ADRESH

La figura 4 mostra l'inizializzazione dei registri relativi alle linee di I/O e al convertitore A/D. La linea RA0 viene configurata come ingresso ponendo a 1 logico il bit 0 del registro TRISA, mentre tutte le altre linee vengono configurate come uscite. Il registro ADCON1 viene configurato con il valore binario '00001110' il che implica che il risultato della conversione è giustificato a sinistra, e l'unica linea configurata come analogica è la RA0. Tutte le altre linee di I/O sono digitali. Tutte le interrupt vengono disabilitate ponendo il bit 7 del registro INTCON a zero.

La figura 6 riporta il codice assembler relativo al controllo del convertito-

```

Init
    Bsf STATUS,0      ;Passa al banco registri 1
    Movlw 0x01        ;Linea RA0 configurata come ingresso
    Movwf TRIS_A
    Clrf TRIS_B
    Clrf TRIS_C
    Clrf TRIS_D
    Clrf TRIS_E
    Movlw B'00001110'
    Movwf ADCON1
    Bcf STATUS,0      ;Passa al banco registri 0
    Bcf INTCON,7      ;Disabilita tutte le interrupt

```

Figura 4: Inizializzazione registri

| U-0 | U-0 | R/W-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|-------|-----|-------|-----|-------|-------|-------|-------|
| ADFM | — | — | — | PCFG3 | PCFG2 | PCFG1 | PCFG0 |
| bit 7 | | | | | | | bit 0 |

bit 7 **ADFM:** A/D Result Format Select bit
1 = Right justified. 6 Most Significant bits of ADRESH are read as '0'.
0 = Left justified. 6 Least Significant bits of ADRESL are read as '0'.

bit 6-4 **Unimplemented:** Read as '0'

bit 3-0 **PCFG3:PCFG0:** A/D Port Configuration Control bits:

| PCFG3: PCFG0 | AN7 ⁽¹⁾ RE2 | AN6 ⁽¹⁾ RE1 | AN5 ⁽¹⁾ RE0 | AN4 RA5 | AN3 RA3 | AN2 RA2 | AN1 RA1 | AN0 RA0 | VREF+ | VREF- | CHAN/ Refs ⁽²⁾ |
|-----------------|---------------------------|---------------------------|---------------------------|------------|------------|------------|------------|------------|-------|-------|------------------------------|
| 0000 | A | A | A | A | A | A | A | A | VDD | VSS | 8/0 |
| 0001 | A | A | A | A | VREF+ | A | A | A | RA3 | VSS | 7/1 |
| 0010 | D | D | D | A | A | A | A | A | VDD | VSS | 5/0 |
| 0011 | D | D | D | A | VREF+ | A | A | A | RA3 | VSS | 4/1 |
| 0100 | D | D | D | D | A | D | A | A | VDD | VSS | 3/0 |
| 0101 | D | D | D | D | VREF+ | D | A | A | RA3 | VSS | 2/1 |
| 011x | D | D | D | D | D | D | D | D | VDD | VSS | 0/0 |
| 1000 | A | A | A | A | VREF+ | VREF- | A | A | RA3 | RA2 | 6/2 |
| 1001 | D | D | A | A | A | A | A | A | VDD | VSS | 6/0 |
| 1010 | D | D | A | A | VREF+ | A | A | A | RA3 | VSS | 5/1 |
| 1011 | D | D | A | A | VREF+ | VREF- | A | A | RA3 | RA2 | 4/2 |
| 1100 | D | D | D | A | VREF+ | VREF- | A | A | RA3 | RA2 | 3/2 |
| 1101 | D | D | D | D | VREF+ | VREF- | A | A | RA3 | RA2 | 2/2 |
| 1110 | D | D | D | D | D | D | D | A | VDD | VSS | 1/0 |
| 1111 | D | D | D | D | VREF+ | VREF- | D | A | RA3 | RA2 | 1/2 |

A = Analog input D = Digital I/O

Figura 5: Struttura del registro ADCON1

re A/D. Il bit 7 indica la giustificazione del risultato a 10 bit verso sinistra ponendo i bit meno significativi a zero logico. Il registro ADCON0 viene caricato con il valore $(11000001)_2$ il quale indica che il canale di ingresso è la linea RA0 e il settaggio a 1 del bit 0 indica che il convertitore A/D è operativo. Il bit 2 avvia la conversione vera e propria e viene settato al passo 3, cioè dopo l'acquisizione del livello di tensione da convertire.

La fine della conversione viene individuata mediante il bit 6 del registro PIR1, il quale viene settato ogni qualvolta una conversione viene portata a termine.

Le ultime due istruzioni consentono di stabilire se la conversione è terminata o meno testando il bit 6 del registro PIR1: se la conversione è terminata, il bit 6 si trova ad 1 logico e quindi viene saltata l'istruzione Goto Back uscendo dal loop altrimenti si ritorna alla label Back. E' evidente che tra un'acquisizione e l'altra è necessario inserire una routine di ritardo per permettere alla capacità di hold di caricarsi al valore di tensione da campionare. A tal proposito si rimanda alla routine di ritardo riportata nell'articolo per il controllo di un display LCD 16x2².

4 La tastiera a matrice

Ogni tasto della tastiera è collegato ad una resistenza diversa e quindi ad ogni tasto corrisponde un valore di tensione diverso. Il riconoscimento di ogni tasto viene fatto dunque in base alla stringa di bit corrispondente al livello di tensione convertito dal modulo A/D. Se tale valo-

²Si faccia riferimento all'articolo "Controllo di un display LCD tramite PIC" - <http://www.mnetft.it>

re si trova all'interno dell'intervallo relativo per esempio al tasto 1, si eseguono le operazioni relative al tasto 1 e così via per tutti gli altri tasti. In questo modo si utilizza una sola porta del PIC per controllare la tastiera e quindi si ottiene un evidente risparmio di risorse di I/O.

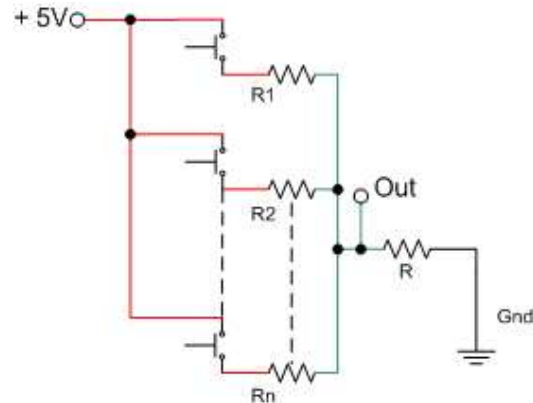


Figura 9: Schema elettrico della tastiera

D'altra parte è necessario utilizzare diversi valori di resistenza per ogni tasto e questo rappresenta una limitazione per questo tipo di controllo. Il valore di tensione fornito dalla pressione di ogni tasto è data dal partitore in tensione:

$$V_{Out} = 5 \cdot \frac{R}{R + R_{\ddagger}} \quad (4)$$

dove \ddagger indica la resistenza relativa ad ogni tasto.

La figura 10 mostra la dichiarazione dei registri utilizzati nel controllo ovvero, i registri relativi alla gestione del modulo A/D. E' importante notare che mentre il registro ADCON0 si trova nel banco registri 0, ADCON1 si trova sul banco registri 1.

Il codice di figura 11 è quello relativo al riconoscimento del tasto "Ok" di un menu.


```

Movlw B'11000001' ;Porta di ingresso analogico linea RA0
Movwf ADCON0
Bcf PIR1,6 ;Poni a zero il bit 6 del registro PIR1
Bsf ADCON0,2 ;Inizia la conversione
Back Btfss PIR1,6 ;Esegui loop fino alla fine della conversione
Goto Back
;Istruzione 1
;...
;Istruzione n

```

Figura 6: Inizializzazione convertitore A/D

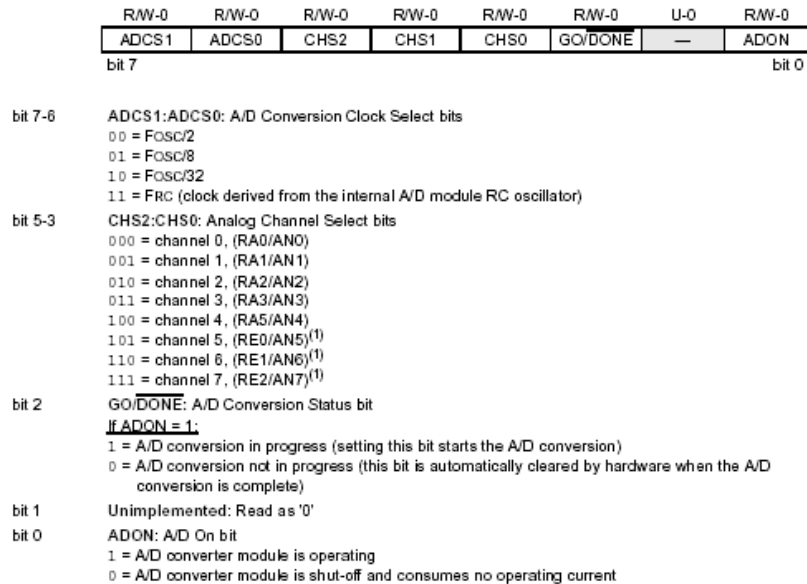


Figura 7: Struttura del registro ADCON0

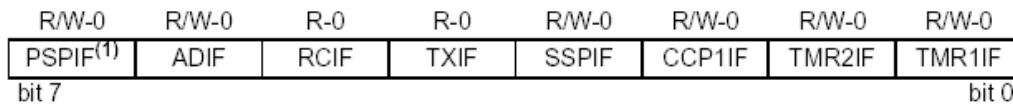


Figura 8: Struttura del registro PIR1

```

STATUS EQU 0X03
PORT_A EQU 0X05
PORT_B EQU 0X06
PORT_C EQU 0X07
PORT_D EQU 0X08
PORT_E EQU 0X09
INTCON EQU 0X0B
TRIS_A EQU 0X85
TRIS_B EQU 0X86
TRIS_C EQU 0X87
TRIS_D EQU 0X88
TRIS_E EQU 0X89
PIR1 EQU 0x0C
ADRESH EQU 0x1E
ADCON0 EQU 0x1F
ADRESL EQU 0x9E
ADCON1 EQU 0x9F

```

Figura 10: Dichiarazione registri

```

TastOk Movlw 0xA3
        Subwf ADRESH,0
        Btfss STATUS,0
        Goto Next
        Movlw 0xA7
        Subwf ADRESH,0
        Btfss STATUS,0
        Goto OpOk
        Goto Next

```

Figura 11: Codice di controllo tasto "Ok"

La prima istruzione carica il valore $(A3)_H$ nel registro di lavoro W, mentre la seconda istruzione sottrae tale valore al registro ADRESH (che contiene il dato letto dal convertitore A/D) ponendo a sua volta, il risultato ottenuto nel registro W. La terza istruzione esegue il test del bit zero del registro STATUS il quale risulta settato a uno se il risultato della sottrazione è positivo, mentre è zero nel caso in cui il risultato sia negativo. Quindi se il risultato è negativo vuol dire che il valore letto dalla conversione è fuori dall'intervallo $(A3 \div A7)_H$ e quindi viene eseguita l'istruzione Goto Next verificando se il valore misurato contenuto in ADRESH appartiene a qualche altro intervallo. Se il bit 0 di STATUS è settato vuol dire che il valore di ADRESH è maggiore di $(A3)_H$ e quindi viene saltata l'istruzione Goto Next al fine di verificare se è minore di $(A7)_H$ e quindi se appartiene a tale intervallo. Infatti, qui viene caricato il valore $(A7)_H$ nel registro di lavoro W e viene sottratto al registro ADRESH, ponendo il risultato nuovamente nel registro W. A questo punto viene testato il bit 0 del registro STATUS: se il risultato è positivo (bit 0 del registro STATUS settato a 1), vuol dire che il valore di ADRESH è maggiore di $(A7)_H$ e quindi fuori dall'intervallo, mentre se il risultato è negativo (bit 0 del registro STATUS settato a 0) vuol dire che il valore di ADRESH è minore di $(A7)_H$ e quindi appartiene all'intervallo. Nell'ultimo caso viene eseguita la procedura relativa alla label OpOk, mentre se il valore è fuori dall'intervallo si prosegue verificando se appartiene ad altri intervalli. Sarebbe interessante realizzare un controllo del genere sostituendo le resistenze con dei transistor in modo tale

da ottenere tensioni più precise, riducendo al minimo gli errori di riconoscimento dei tasti e magari realizzando il tutto in tecnologia monolitica.

Nell'esempio mostrato, la tensione massima è stata assunta pari a 5V ed essendo il convertitore a 10 bit, il campionamento avviene mediante quanti di valore pari a:

$$\begin{aligned} AD_{step} &= \frac{5}{1024} = 0.0048828125V = \\ &\approx 4.8mV \end{aligned} \quad (5)$$

Quindi il range di tensioni riportato nell'esempio di figura 11, è definito dai seguenti valori:

$$\begin{aligned} V_{sup} &= (A3)_H \cdot AD_{step} \\ V_{inf} &= (A7)_H \cdot AD_{step} \end{aligned} \quad (6)$$

$$\begin{aligned} V_{sup} &= (163)_D \cdot 4.8 \cdot 10^{-3} = 782,4mV \\ V_{inf} &= (167)_D \cdot 4.8 \cdot 10^{-3} = 801.6mV \end{aligned} \quad (7)$$

Il metodo riportato nell'articolo è stato testato per controllare una tastiera a matrice 4x4, facendo uso di resistenze con tolleranza al 5% con risultati molto soddisfacenti.

Riferimenti bibliografici

- [1] Microchip, '*PIC16F87X Datasheet*', Microchip Arizona.
- [2] G.F. Ciccarella, A. Londei, F. Loriga P. Marietti, '*Sistemi Elettronici Digitali*', Masson, 1994.