

I nuovi controller PIC

Come in ogni settore dell'elettronica, anche nel campo dei microcontroller ogni giorno nascono delle novità. Ecco i nuovi componenti Microchip delle famiglie 12, 16 e 17xxx

Paolo Sbrana - 1ª parte

Quando nell'anno 1994 **Progetto** ha iniziato a parlare di microcontroller PIC, pochissimi in Italia li conoscevano, ad eccezione delle sole ditte ad alto livello tecnologico.

Abbiamo così approntato un primo corso sulla programmazione di questi microcontroller

RISC e successivamente si sono viste tutte le periferiche nate piano piano intorno alla CPU.

Ricordiamo che quando abbiamo trattato per la prima volta questi componenti, i tipi disponibili sul mercato italiano erano molto limitati: si potevano reperire i quattro PIC della famiglia base (PIC16C54,

PIC16C55, PIC16C56 e PIC16C57), l'unico PIC in tecnologia EEPROM (PIC16C84), l'unico PIC con un convertitore a bordo (PIC16-C71) e l'unico PIC della famiglia 17, ovvero il PIC17C42. Nel momento in cui scriviamo i tipi di chip disponibili sono già oltre 40, ognuno con caratteristiche proprie.

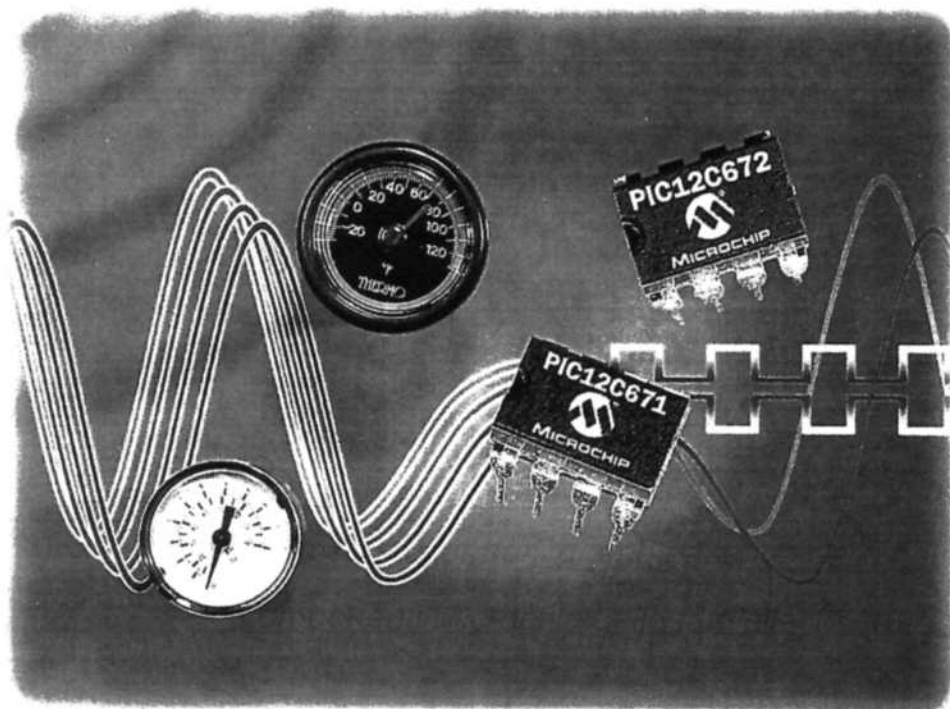
Nostra intenzione è di far conoscere ai lettori i modelli più recenti, analizzando attentamente famiglia per famiglia.

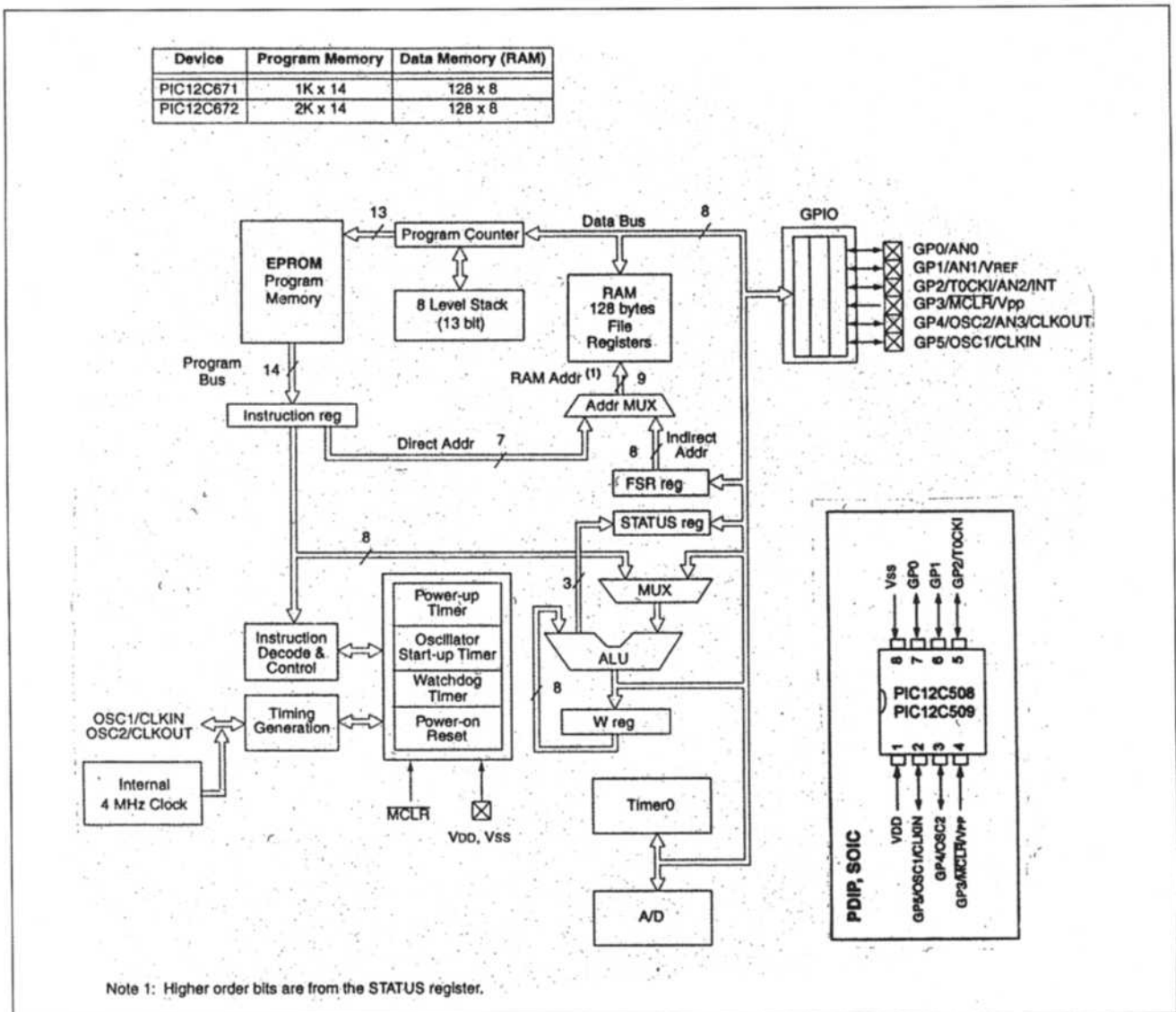
Questo mese inizieremo con i più recenti ed al tempo stesso i più piccoli PIC: la famiglia 12Cxx ad otto pin.

Con questa famiglia Microchip Technology entra in un mercato dove ancora pochi sono riusciti ad ottenere ottimi risultati, forse perchè gli otto pin hanno poche linee di I/O e questo può rivelarsi un handicap.

Le famiglie dei PIC12Cxxx

Iniziamo la nostra rassegna con i microcontroller più piccoli che possiamo trovare sul





mercato: le due famiglie 12C50x e 12C67x entrambe ad otto pin e disponibili perfino in versione SMD (con package detto SOIC per gli addetti ai lavori). Queste due famiglie hanno molto in comune, anche se vedremo che esistono delle differenze sostanziali per quanto riguarda le periferiche possedute e gli interrupt.

Nella tabella 1 valutiamo le caratteristiche principali: la memoria di programma spazia da 512 a 2048 locazioni, la RAM da 25 a 128 registri, le sorgenti di interrupt da 0 a 4, la tensione di alimentazione da 2,5 a 5,5 volt e la frequenza

di funzionamento da 0 a 10 MHz. Analizziamo quindi in dettaglio il diagramma a blocchi riportato in Figura 1, che vale sia per i 12C50x che per i 12C67x.

La struttura è quella tipica dei PIC delle famiglie mid-range, ovvero dei 16C6x e 16C7x, ma ovviamente le linee di I/O sono in numero minore (6).

Un modulo interno che però non è presente in alcun altro PIC è l'oscillatore interno a 4 MHz. Per non sprecare pin infatti, i progettisti hanno deciso di inserirne uno da circa 4 MHz all'interno, anche se poi è possibile applicare egualmente un

Figura 1.
Diagramma
a blocchi dei
PIC12Cxxx

oscillatore esterno per particolarissime applicazioni.

Vedremo inoltre che la frequenza di tale oscillatore è "regolabile" via software agendo su un ben preciso registro dedicato.

Si capisce subito la comodità di una scelta simile: per prima cosa si ha un componente in meno (o tre se si considerano anche i due condensatori) sul circuito con vantaggi economici e di dimensionamento, poi si liberano due pin altrimenti dedicati solo a questa ben definita funzione (che in un chip ad otto pin ne lascerebbero quindi solo 4 liberi).

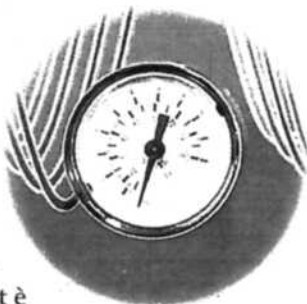
I livelli di stack sono rimasti due per la famiglia 12C-50x mentre sono aumentati a otto per la famiglia 12C67x.

Il modulo di conversione A/D ad 8 bit è presente solo nei 12C67x. Per il resto tutti gli altri moduli sono identici agli altri controller Microchip.

Ciò che cambia nei quattro modelli attualmente disponibili sono la quantità della memoria di programma (ROM) e quella dei registri RAM.

In Figura 2-a possiamo vedere la suddivisione della ROM nei 12C50x ed in Figura 2-b quella relativa ai 12C67x.

Per quanto riguarda la RAM invece, in Figura 3 vediamo le tre diverse pezzature rispettivamente per i 12C508, 12C509 e 12C67x.



Il primo tipo rientra in un banco solo, mentre i rimanenti necessitano di un secondo banco. Come si vede dalla Figura, i progettisti hanno fatto in modo di unificare gli indirizzi comuni di tutti i nuovi controller, sacrificando così locazioni che

non possono essere mai utilizzate (ad esempio come la serie di registri che nei 12C-67x va dall'indirizzo 0xd all'indirizzo 0x1d).

di stato dei 12C50x (Figura 4-a) con quello dei 12C67x (Figura 4-b).

I bit dallo 0 al 4 hanno lo stesso significato ovvero: il bit 0 è un flag che indica se dopo un'operazione si è verificato un riporto.

Il bit 1 ha lo stesso significato del bit 0 ma riferito a soli quattro bit.

Il bit 2 indica se dopo un'operazione si è ottenuto risultato zero o diverso da zero.

Il bit 3 serve per capire se il controller si è resettato dopo una caduta di alimentazione oppure dopo un'istruzione di sleep.

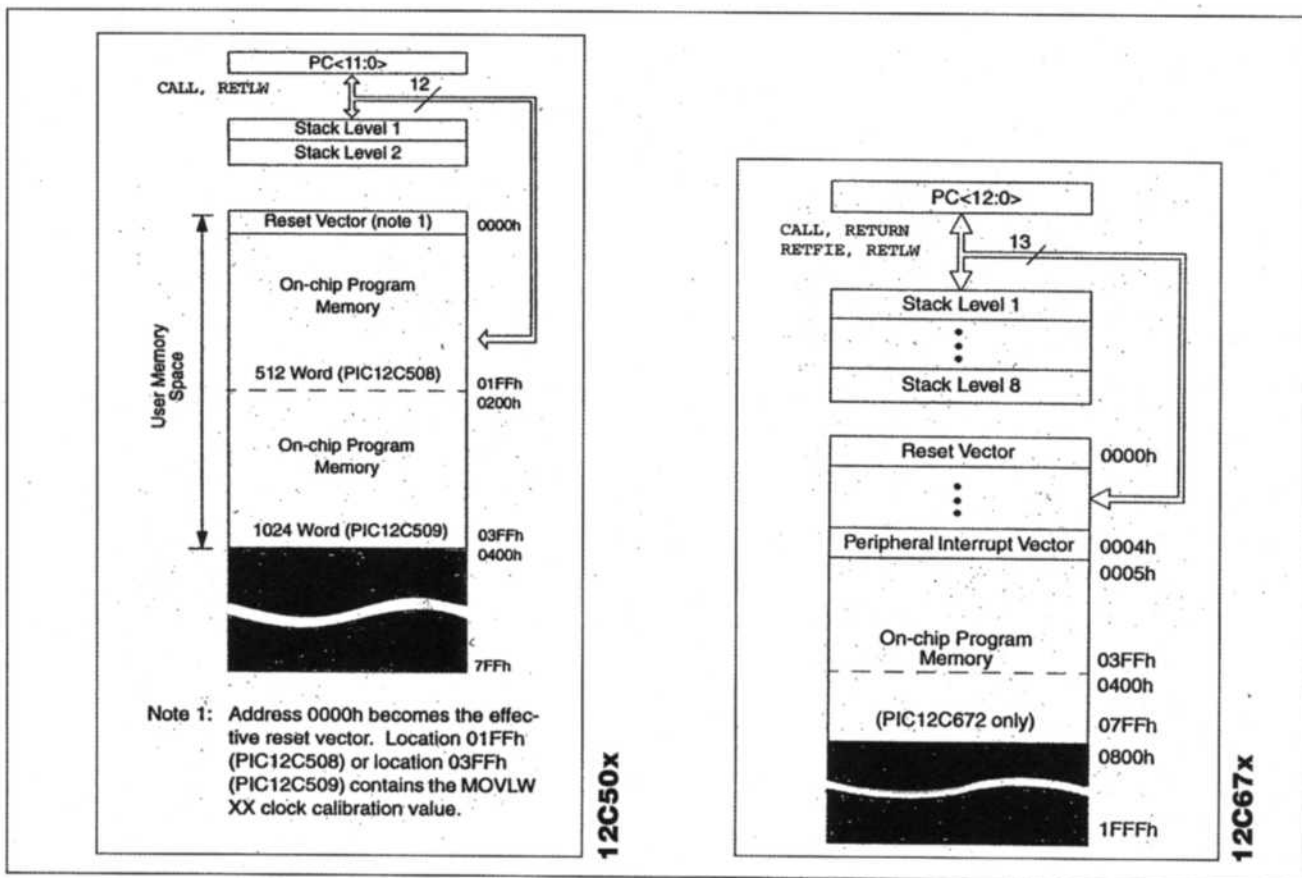
Il bit 4 invece indica se c'è stato il Time-out dall'watchdog interno.

Il bit 5 del registro di stato dei 12C50x serve per selezionare la pagina ROM desiderata (ovviamente solo per i 12C509) mentre il bit 6 non ha valore ed il bit

Figura 2a e2b.
Mapa della memoria ROM nelle famiglie 12C67x e 12C50x

Lo status register

Proprio per la diversità di implementazione, anche i registri interni hanno a volte delle piccole varianti, come si può vedere confrontando il registro



R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	TO	PD	Z	DC	C
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **IRP**: Register Bank Select bit (used for indirect addressing)
1 = Bank 2, 3 (100h - 1FFh)
0 = Bank 0, 1 (00h - FFh)
The IRP bit is reserved, always maintain this bit clear.

bit 6-5: **RP1:RP0**: Register Bank Select bits (used for direct addressing)
11 = Bank 3 (180h - 1FFh)
10 = Bank 2 (100h - 17Fh)
01 = Bank 1 (80h - FFh)
00 = Bank 0 (00h - 7Fh)
Each bank is 128 bytes. The RP1 bit is reserved, always maintain this bit clear.

bit 4: **TO**: Time-out bit
1 = After power-up, CLRWDT instruction, or SLEEP instruction
0 = A WDT time-out occurred

bit 3: **PD**: Power-down bit
1 = After power-up or by the CLRWDT instruction
0 = By execution of the SLEEP instruction

bit 2: **Z**: Zero bit
1 = The result of an arithmetic or logic operation is zero
0 = The result of an arithmetic or logic operation is not zero

bit 1: **DC**: Digit carry/borrow bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)(for borrow the polarity is reversed)
1 = A carry-out from the 4th low order bit of the result occurred
0 = No carry-out from the 4th low order bit of the result

bit 0: **C**: Carry/borrow bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)
1 = A carry-out from the most significant bit of the result occurred
0 = No carry-out from the most significant bit of the result occurred
Note: For borrow the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand. For rotate (RRF, RLF) instructions, this bit is loaded with either the high or low order bit of the source register.

attenzione ai chip finestrati, poiché tale porzione di memoria viene cancellata ogni volta che si impiegano lampade a raggi ultravioletti per cancellare tutto il chip. In questi casi, conviene memorizzare tale valore in un file. L'oscillatore interno ha a disposizione 16 livelli di aggiustamento dalla frequenza propria principale, che essendo in realizzata con una rete RC può variare in base a diversi fattori sia in più che in meno rispetto ai 4 MHz prefissati.

Questi 16 livelli sono suddivisi in livelli di "aumento" della frequenza prodotta ed in livelli di "diminuzione" della frequenza prodotta.

A questo punto possiamo vedere il significato del registro OSCCAL: il bit 0 ed il bit 1 non vengono utilizzati.

Con i bit 2 e 3 diciamo all'oscillatore se deve aggiustare la sua frequenza aumentandola o diminuendola.

Con i quattro bit dal 4 al 7 si seleziona di quanto ci si deve discostare dalla frequenza di oscillazione principale. Con i 4 bit sono quindi disponibili 2 elevato alla quarta cioè 16 livelli (o step) di aggiustamento.

I registri delle interruzioni

Gli interrupt sono disponibili solamente per la famiglia 12C-67x ed in Figura 7 troviamo il registro INTCN che li gestisce completamente.

Figura 4-b.
Il registro di stato nei 12C67x



Il bit 0 è un flag che viene settato quando giunge un interrupt dal cambio di stato di uno dei tre pin GP0, GP1 e GP3. Il bit 1 è il flag che segnala l'interrupt dal pin GP2.

Il bit 2 invece segnala l'interrupt proveniente dal timer0. I bit 3, 4 e 5 sono i rispettivi bit di abilitazione al funzionamento delle sorgenti di interrupt appena viste.

Il bit 6 abilita gli interrupt da periferica (che in questo chip è rappresentata dal solo modulo convertitore A/D). Il bit 7 abilita o meno tutti i tipi di interrupt. La gestione dell'interrupt da conversione, viene effettuata attraverso i due registri dedicati PIR1 e PIE, visibili rispettivamente in Figura 8-a e Figura 8-b. Nel primo al bit 6 si ha il flag di avvenuto interrupt,

	W-1	W-1	W-1	W-1	W-1	W-1	W-1	W-1
	GPWU	GPPU	T0CS	T0SE	PSA	PS2	PS1	PS0
bit7:	6	5	4	3	2	1	bit0	

bit 7: **GPWU:** Enable wake-up on pin change (GP0, GP1, GP3)
1 = Disabled
0 = Enabled

bit 6: **GPPU:** Enable weak pull-ups (GP0, GP1, GP3)
1 = Disabled
0 = Enabled

bit 5: **T0CS:** Timer0 clock source select bit
1 = Transition on T0CKI pin
0 = Transition on internal instruction cycle clock, Fosc/4

bit 4: **T0SE:** Timer0 source edge select bit
1 = Increment on high to low transition on the T0CKI pin
0 = Increment on low to high transition on the T0CKI pin

bit 3: **PSA:** Prescaler assignment bit
1 = Prescaler assigned to the WDT
0 = Prescaler assigned to Timer0

bit 2-0: **PS2:PS0:** Prescaler rate select bits

Bit Value	Timer0 Rate	WDT Rate
000	1:2	1:1
001	1:4	1:2
010	1:8	1:4
011	1:16	1:8
100	1:32	1:16
101	1:64	1:32
110	1:128	1:64
111	1:256	1:128

W = Writable bit
U = Unimplemented bit
- n = Value at POR reset
Reference Table 4-1 for other resets.

mentre nel secondo, sempre al bit 6, si ha l'abilitazione o meno a quel tipo interruzione.

Tutti gli altri bit non vengono impiegati.

I registri del convertitore A/D

Passiamo adesso alla periferica del convertitore. Per capire a che cosa servono i bit dei due registri dedicati, vediamo prima in Figura 9 il diagramma a blocchi del convertitore e dei suoi possibili collegamenti con il mondo esterno.

Come si capisce dal grafico, al modulo arriva soltanto un segnale da convertire, gestito su 4 canali con un multiplex interno. Allo stesso modo, la tensione di riferimento Vref necessaria allo stadio convertitore, può sia essere presa direttamente dal positivo di alimentazione inter-

Figura 5.
Il registro delle opzioni nei 12Cxxx

no, sia prelevata esternamente dal pin GP1.

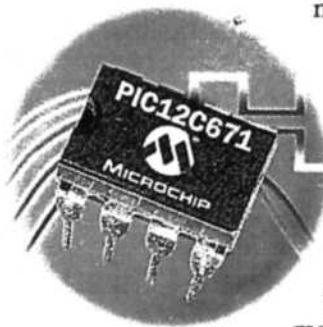
Se dopo questa premessa si passa alla Figura 10, si riesce a capire meglio il significato dei bit del registro ADCON0: il bit 0 serve per attivare o meno il modulo convertitore. Ovviamente, quando tale modulo

non serve, è inutile che assorba corrente restando acceso.

Il bit 1 è riservato. Il bit 2 fa partire la conversione e ci avvisa quando è terminata (unitamente all'eventuale interrupt se abilitato).

I due bit 3 e 4 consentono di selezionare il canale prescelto sul multiplex prima visto.

Chiaramente abbiamo due bit per consentire quattro possibili ingressi diversi.



Il bit 5 è riservato. Con i due bit 6 e 7 invece è possibile selezionare l'oscillatore relativo alla periferica del convertitore: si può avere la frequenza del clock divisa per 2, per 8 o per 32, oppure ottenere una frequenza da un oscillatore interno RC dedicato solo a questo compito.

Tale oscillatore viene spento quando il bit 0 di attivazione viene azzerato e viene riacceso quando tale bit viene settato.

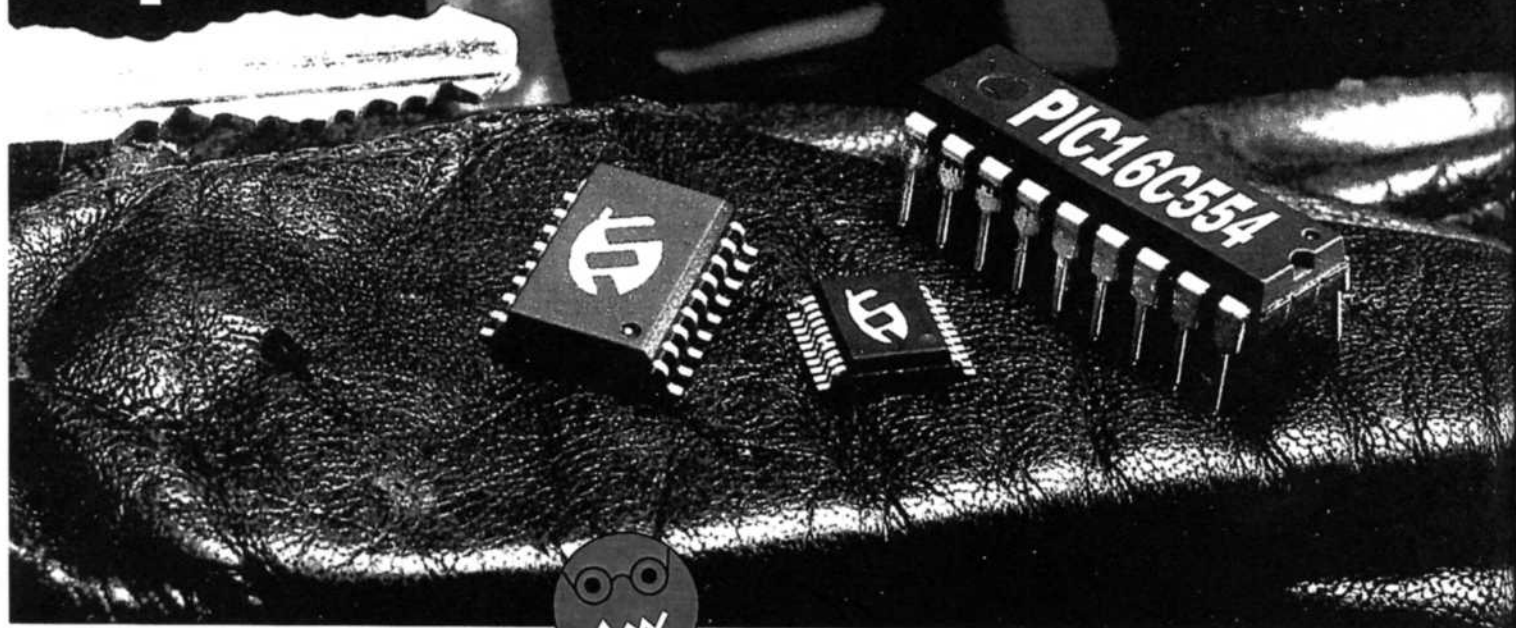
L'ultimo registro dedicato al modulo convertitore è ADCON1, visibile in Figura 11: con i primi tre bit è possibile impostare sul controller le configurazioni sui pin GP0, GP1, GP2 e GP4 in modo da selezionare il Vref voluto e gli ingressi analogici o digitali.

Poichè non sempre sono indispensabili gli otto bit di definizione relativi al convertitore, è possibile ottenere conversioni con una risoluzione più bassa, lavorando sul tempo di conversione. È sufficiente infatti bloccare il convertitore dopo un certo tempo, minore del tempo totale necessario ad una conversione completa ad otto bit. Così facendo, nel registro ADRES verranno settati solamente n bit con n inferiore ad 8.

Conoscendo per esempio il tempo di una conversione completa, interrompendola dopo tempo/2, si otterrà una risoluzione di 4 bit.

Per concludere, ricordiamo che questa periferica lavora con il metodo delle approssimazioni successive ovvero il segnale viene comparato con la metà della tensione di riferimento e, se maggiore, il bit MSB del risultato viene posto ad 1, viceversa viene azzerato. La comparazione successiva viene eseguita con la tensione di riferimento divisa per 4 e così via fino ad arrivare alla tensione di riferimento divisa per 256. ■

I microcontroller PIC più recenti



*Proseguiamo la nostra
camminata all'interno
dei microcontroller
Microchip esaminando
questo mese la famiglia
dei PIC16C55x*

Paolo Sbrana

Seconda parte

Fanno parte di questa recente famiglia tre microcontroller: il PIC16C554, il PIC16C556 ed il PIC16C558, tutti a 18 pin.

Il nome è molto simile ai 18 pin della famiglia base (16C5x), ed infatti nascono proprio per ampliare tale fascia inserendo al tempo stesso una serie di innovazioni. Cerchiamo di individuarle andando a vederne il diagramma a blocchi in figura 12.

Si nota subito che l'area della memoria di programma (EPROM) è diversa da chip a chip: i PIC16-C554 hanno 512 locazioni, i PIC16C556 hanno 1024 locazioni ed i PIC16C558 ne hanno 2048. Per quanto riguarda l'area di memoria dati (o utente o RAM) i PIC16C554 e PIC16C556 hanno a disposizione 80 registri, mentre i PIC16C558 ne hanno 128.

Un'altra sostanziale differenza rispetto alla famiglia base

sono i livelli disponibili per lo stack: otto contro due.

Ma la vera innovazione è la possibilità di lavorare con delle sorgenti di interrupt che, nei primi chip, non erano possibili.

Ultima peculiarità, la capacità di essere programmati con la programmazione in circuit ovvero con la ISP (In Serial Programming).

La gestione della memoria

Come già accennato, ogni microcontroller ha una propria gestione delle due aree di memoria (EPROM e RAM). In figura 13-a, 13-b e 13-c distinguiamo le diverse capacità.

Il PIC 16C554 ha il vettore di start alla locazione 000h, il vettore di interrupt alla locazione 4 e l'ultima locazione di memoria disponibile all'indirizzo 01FFh.

Il PIC 16C556 ha gli stessi indirizzi eccetto che per l'ultima locazione disponibile che si trova all'indirizzo 03FFh.

Il PIC16C558 invece ha l'ultima locazione all'indirizzo 07F6Fh. A differenza della famiglia base, non sono previste paginazioni della memoria EPROM, abbastanza complesse da gestire specie per i meno esperti.

Ad esempio confrontando il PIC16C58 con il corrispondente PIC16C558, si sa che il primo ha la memoria di programma suddivisa in quattro

banchi da 512 locazioni ciascuno. Ciò implica che per passare da una pagina all'altra si devono settare correttamente i due bit appositi e per tornare indietro dobbiamo riconfigurarli.

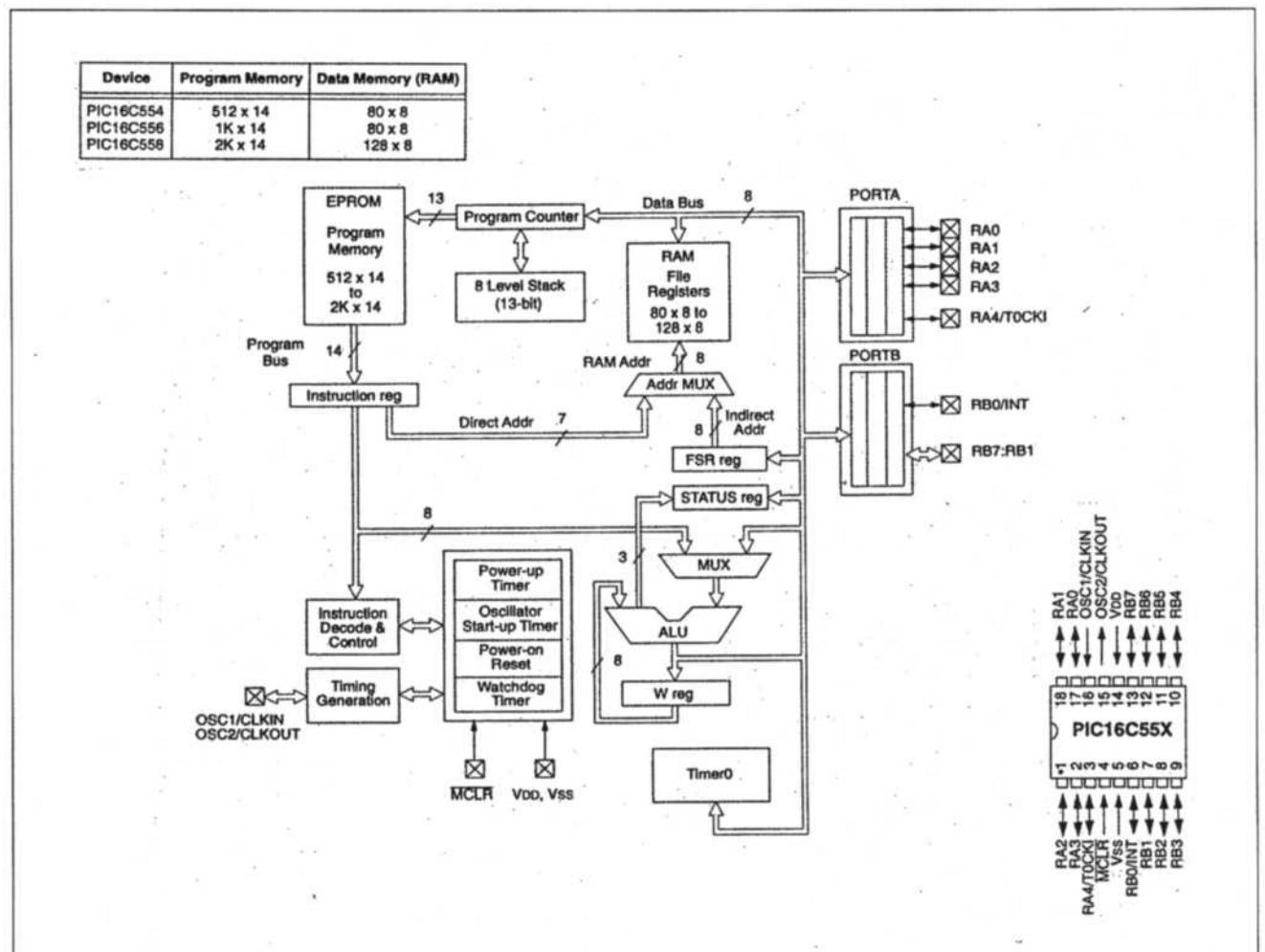
Se ci scordiamo di farlo il programma continuerà a lavorare nello stesso banco su locazioni casuali.

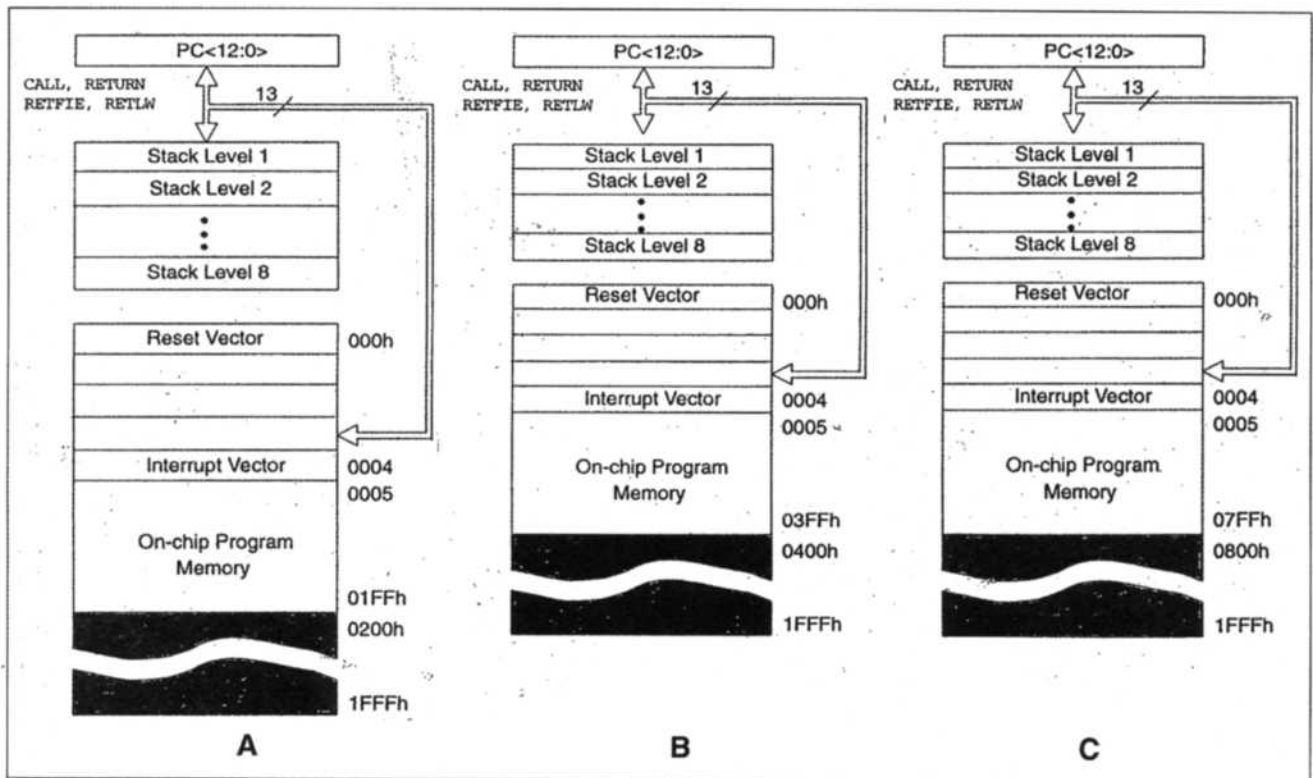
Il problema della paginazione della EPROM non sussiste invece nella famiglia 16C55x.

Viceversa, per uniformarsi alle nuove famiglie, sono paginati i registri specifici e quelli ad uso generale come si vede in figura 14-a e 14-b.

I registri speciali sono gli stessi per entrambi, mentre i registri ad uso generale (sfruttabili dal

Figura 12. Diagramma a blocchi della famiglia 16C55x





programmatore) vanno dall'indirizzo **20h** all'indirizzo **6Fh** per i 16C554 e 16C556 mentre vanno dall'indirizzo **20h** al **7Fh** e dallo **A0h** al **BFh** per il 16C558.

Come si vede chiaramente, ci sono molte locazioni che non vengono utilizzate, come per esempio quelle dall'indirizzo **0Ch** all'indirizzo **1Fh** compreso.

Il motivo di tale decisione è che la Microchip ha voluto standardizzare tutte le nuove famiglie almeno a livello di indirizzi, quindi troveremo ad esempio che molti chip hanno la prima locazione disponibile di RAM all'indirizzo **20h** ma che, a seconda delle famiglie e conseguentemente delle periferiche possedute, hanno dei registri speciali in locazioni qui non utilizzate.

I cosiddetti registri speciali (o dedicati) sono tutti quei registri che si trovano in un controller e che hanno un nome ben definito.

Figura 13.
Locazioni
EPROM nei
PIC16C554,
PIC16C555
e PIC16C558

I registri speciali

Tra i registri speciali notiamo il TMR0 (TiMeR 0) che viene impiegato come un vero e proprio contatore ad 8 bit con o senza prescaler, il PCL (Program Counter Load) che è il registro di indirizzamento delle istruzioni, lo STATUS REGISTER dove si trovano i risultati delle operazioni ed alcune impostazioni del controller, il FSR (File Select Register) che permette di ottenere un indirizzamento indiretto, i registri PORTA e PORTB che sono le vere e proprie porte del microcontrollore, il PCLATH che è la parte alta del registro PCL visto prima, il registro INTCON che vedremo tra poco gestisce le interruzioni, il registro OPTION che permette di impostare alcune opzioni di funzionamento, i due registri TRISA e TRISB che permettono di settare un pin come ingresso o come uscita ed infine il registro PCON il cui contenuto è visibile in figura 15.

L'unico bit utilizzato dal registro PCON è un flag che ci dice se il microcontrollore si è risvegliato da uno stato di reset oppure dopo che gli è stata fornita alimentazione. In pratica ci serve per capire se si è resettato per un qualsiasi motivo software (stato di sleep, reset sul pin MCLR, watchdog) oppure perché semplicemente si è tolta e ridata alimentazione.

L'altro registro che nella famiglia base non è presente è quello della gestione degli interrupt rappresentato in figura 16 (INTCON). Il bit GIE (Global Interrupt Enable) abilita o meno il salto alla routine di interrupt.

Il bit TOIE (Timer0 Overflow Interrupt Enable) abilita o meno l'interrupt generato dall'overflow del Timer0.

Il bit INTE (RB0/INT External interrupt enable bit) abilita o meno l'interrupt esterno dal pin RB0.

Il bit RBIE (RB port change Interrupt Enable) abilita o meno

