Istruzioni di cancellazione

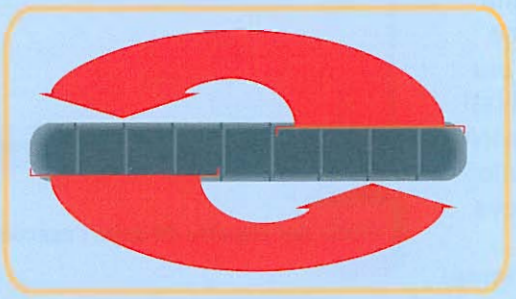
Fra le 35 istruzioni Assembler ve ne sono tre che non sono state classificate in alcun gruppo perché la loro funzione non ha similitudini con nessun'altra. Vediamo come si comportano con alcuni esempi.

Cambio di posizione dei bit di un registro

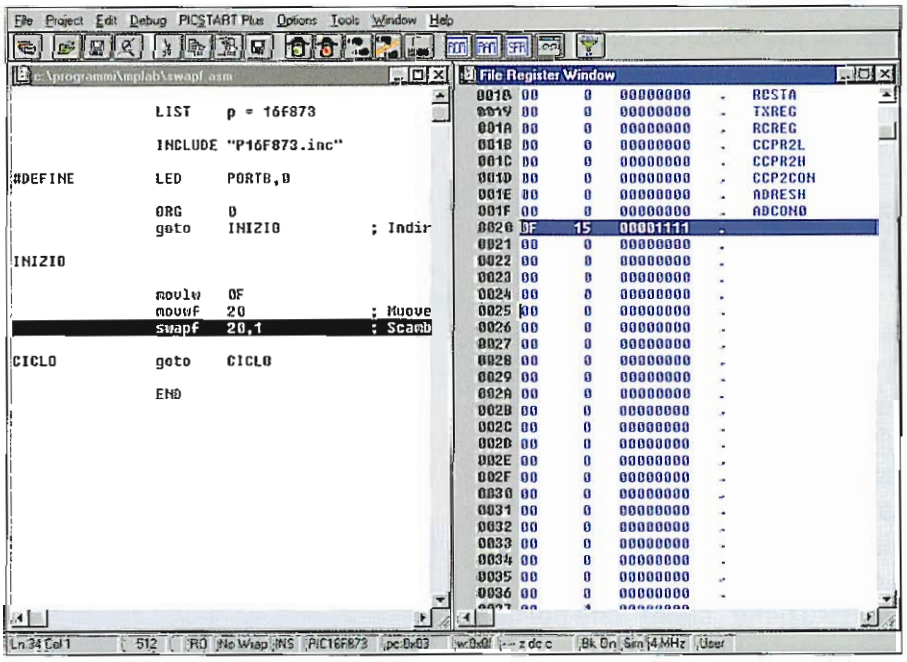
La prima delle istruzioni cambia le posizioni dei bit più significativi di un registro per i bit meno significativi. Il suo nome è "swapf" e, come con le altre istruzioni con lo stesso formato, il risultato dell'operazione si può depositare sul registro di origine o sul registro di lavoro W, a seconda del valore del parametro D.

Anche se il suo utilizzo come istruzione indipendente è semplice, di solito viene impiegata con efficacia in routines più complesse che lavorano con i numeri, ad esempio in BCD. Dato che i numeri in BCD hanno una dimensione di 4 bit, spesso vengono scritti due a due in un solo registro per risparmiare spazio. Al momento di lavorare con questi numeri, ad esempio per eseguire una somma, si presenta però un inconveniente, in quanto i numeri dovrebbero necessariamente occupare la parte bassa del registro e lasciare la parte alta a valore 0.

MNEMONICO	PARAMETRI	SIGNIFICATO
swapf	F,D	Cambia la posizione fra i 4 bit più significativi e i 4 meno significativi di un registro F. il risultato è lasciato nello stesso registro (se D = 1) o sul registro di lavoro W (se D = 0)



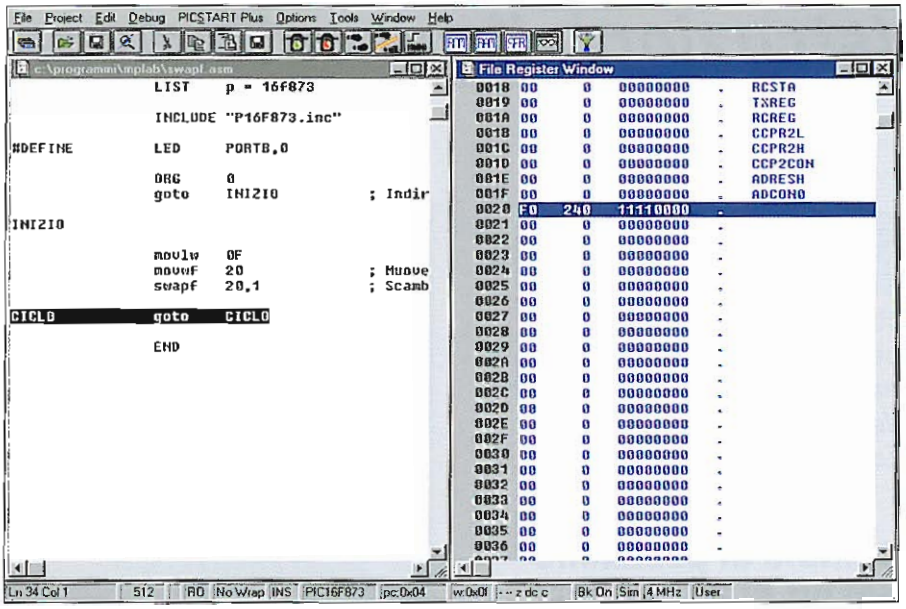
Utilizzo dell'istruzione *"swapf"*.



Stato del registro 20 prima dell'esecuzione dell'istruzione *swapf*.

È a questo punto che si utilizza l'istruzione swapf — in combinazione con altre istruzioni — per realizzare delle maschere che permettano di selezionare solamente i bit che interessano per ogni operazione. Il programma della figura mostra un semplice esempio di utilizzo di questa istruzione. Il numero da muovere sul registro contiene i valori uno sulla parte bassa e zero sulla parte alta. Dopo l'esecuzione dell'istruzione i quattro bit più significativi sono stati cambiati con i quattro meno significativi. Il risultato è stato lasciato sullo stesso registro, però si poteva anche trasferire a W per non modificare il dato di origine.

Se non si varia il registro di destinazione, la prossima volta che si avrà bisogno del valore originale si eseguirà nuovamente l'istruzione swapf, ma in questo caso bisognerà utilizzare un bit come se fosse un flag per sapere in quale situazione si trova il registro in ogni momento.



Stato del registro 20 dopo l'esecuzione dell'istruzione *swapf*.

Basso consumo di energia

In un sistema a microcontroller il risparmio di energia è un fattore molto importante, dato che spesso questi sistemi sono alimentati con batterie la cui durata è limitata. Sia nel LetPicBasicLite che nel LetPicBasicPlus esiste un'istruzione "sleep" che porta il microcontroller nello stato di "riposo" o "basso consumo". I modi per uscire da questo stato, invece, sono diversi, nel caso in cui si produca un reset o sia passato il tempo specificato in secondi. In Assembler, il nome dell'istruzione continua a essere lo stesso, però il suo utilizzo è differente.

L'utilizzo dell'istruzione sleep è semplice perché non ha bisogno di parametri. Quando viene eseguita il microcontroller entra in uno stato in cui consuma pochissima energia, non vengono eseguite istruzioni e le uscite rimangono nello stato in cui si trovavano prima dello sleep. Per uscire da questo stato, oltre al reset del sistema è necessario che si verifichi un overflow del Watchdog (se è stato attivato durante la programmazione del microcontroller) oppure un interrupt. Quando esce dallo stato di riposo, il programma continua con l'istruzione successiva a quella di sleep (a meno che non si tratti di un reset). Quindi non si conosce il tempo in cui il sistema rimarrà in stato di riposo, dato che dipende dall'azione di un utente esterno. In questo modo lavorano, per esempio, i telecomandi.

Un diverso modo di procedere potrebbe essere quello in cui il programma del microcontroller "sorveglia" costantemente il valore dei pulsanti del telecomando di un televisore,

MNEMONICO	PARAMETRI	SIGNIFICATO
sleep		Porta il microcontroller in stato di riposo o di basso consumo

Utilizzo dell'istruzione *"sleep"*.