

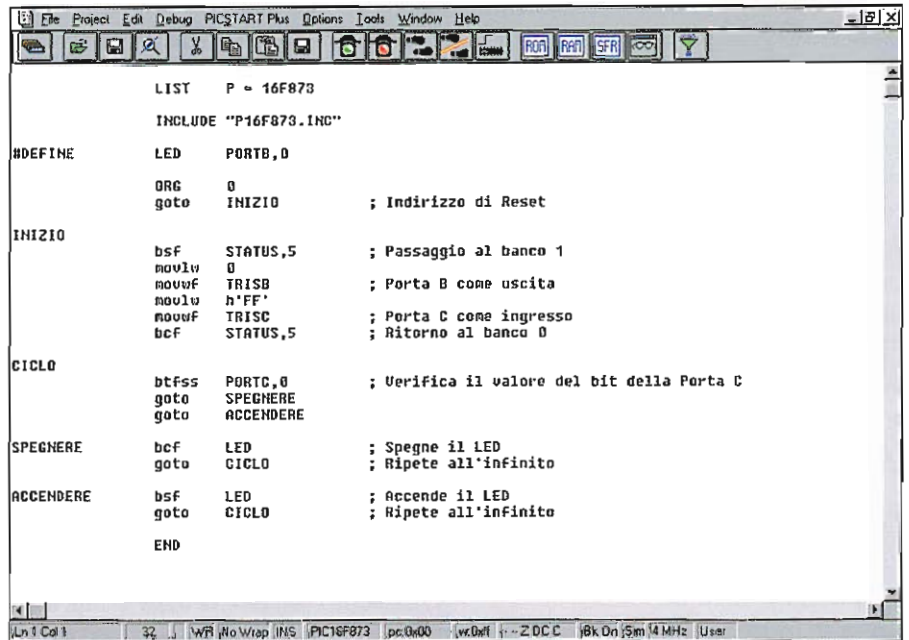
Lavoro con dati semplici

L'acquisizione e la generazione di dati semplici o raggruppati sotto forma di byte, è una caratteristica fondamentale dei sistemi governati da microcontroller. Tuttavia le istruzioni che fornisce l'assembler a questo scopo non sono né numerose né particolari, come capitava con il LetPicBasicLite e con il LetPicBasicPlus, dato che si utilizzano le istruzioni standard, combinate in strutture dal programmatore stesso, per ottenere l'obiettivo desiderato.

Acquisizione di bit singoli

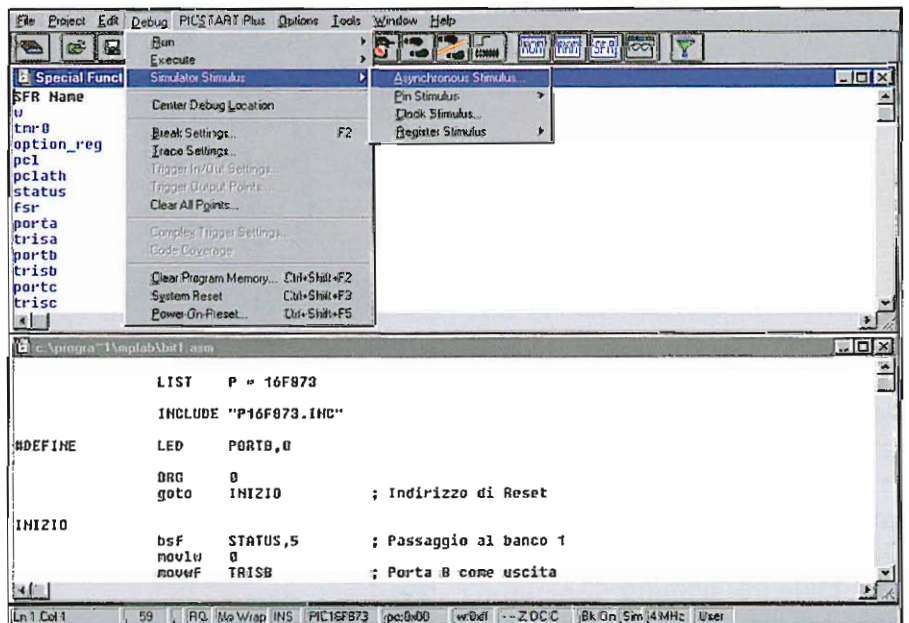
Per eseguire un test su di un singolo bit si utilizzano le istruzioni "btfss" o "btfsc" che abbiamo già visto nel trattamento dei cicli.

Non fa differenza se questi bit appartengono a un registro normale o a una porta di ingresso/uscita. Allo stesso modo l'assegnazione di un valore si realizza con le istruzioni "bsf" e "bcf", utilizzate anche per molte altre cose. Nell'esempio della figura si definisce la Porta B come uscita, e la Porta C come ingresso. A seconda del valore del bit 0 della Porta B si accende un LED collegato al bit 0 della Porta C, che si suppone collegato per livello alto.



```
LIST P = 16F873
INCLUDE "P16F873.INC"
#DEFINE LED PORTB,0
ORG 0
goto INIZIO ; Indirizzo di Reset
INIZIO
bsf STATUS,5 ; Passaggio al banco 1
movlw 0
movwf TRISB ; Porta B come uscita
movlw h'FF'
movwf TRISC ; Porta C come ingresso
bcf STATUS,5 ; Ritorno al banco 0
CICLO
btfss PORTC,0 ; Verifica il valore del bit della Porta C
goto SPEGNERE
goto ACCENDERE
SPEGNERE
bcf LED ; Spegne il LED
goto CICLO ; Ripete all'infinito
ACCENDERE
bsf LED ; Accende il LED
goto CICLO ; Ripete all'infinito
END
```

Per il lavoro con il bit si utilizzano istruzioni standard.



```
LIST P = 16F873
INCLUDE "P16F873.INC"
#DEFINE LED PORTB,0
ORG 0
goto INIZIO ; Indirizzo di Reset
INIZIO
bsf STATUS,5 ; Passaggio al banco 1
movlw 0
movwf TRISB ; Porta B come uscita
```

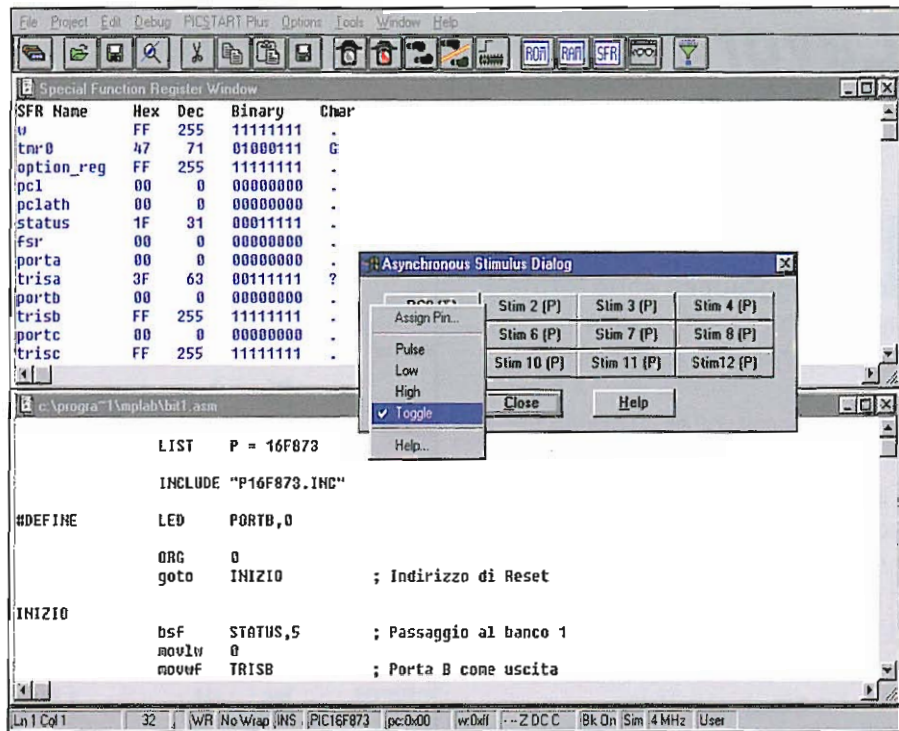
Per la simulazione del programma utilizzeremo un'altra modalità di "Debug".

Assembler per PIC

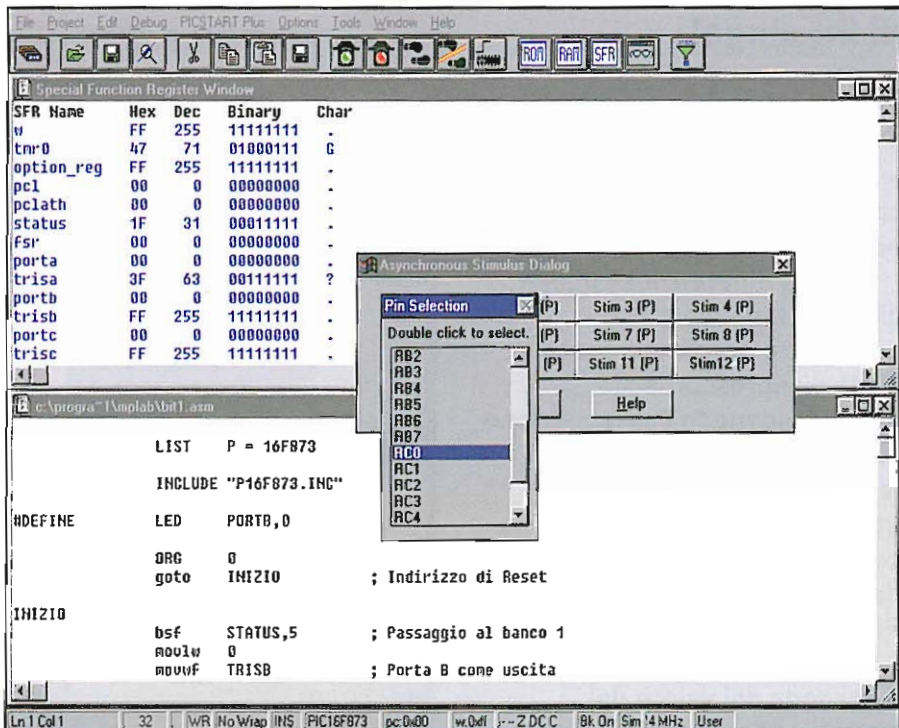
L'assembler, come si può vedere in questo esempio, ci permette di definire sia un bit individualmente, che nell'insieme "PORTA,BIT" con la direttiva "#DEFINE" rendendo più semplice la comprensione del programma. Per verificare il funzionamento di questo programma, utilizzeremo una modalità nuova del menu "Debug". Apriremo la finestra dei registri per poter visualizzare i dati delle porte B e C. All'interno di "Debug", sceglieremo l'opzione "Simulator Stimulus" e all'interno di questa "Asynchronous Stimulus".

Apparirà a questo punto una finestra come quella della figura, dove dovremo scegliere un pin (che sarà la linea di ingresso) e specificare come vogliamo che si comporti. Nel nostro caso sceglieremo "Toggle", cioè ogni volta che attiveremo la linea cambierà di valore, come se si trattasse di un interruttore. Potremmo anche scegliere la funzione di pulsante (Pulse) ovvero l'inserimento di impulsi alti o bassi.

Questo menù si visualizza premendo il tasto destro del mouse. Fra tutte le opzioni possibili dobbiamo anche scegliere la linea della porta a cui ci assoceremo, nel nostro caso il bit 0 della Porta C. Queste definizioni devono essere realizzate per ognuno dei bit di ingresso che vogliamo utilizzare. Anche il modo di eseguire la simulazione dovrà cambiare. Sino a



Bisogna scegliere la linea di ingresso e il suo comportamento.



Scelta del bit 0 della Porta C.

CP065

COME PROGRAMMARE