

Assembler per PIC

esecuzione del programma. Dire che un programma è eseguito in modo corretto significa — dando per scontata la sua correttezza semantica — che dopo aver verificato che faccia ciò che deve, non si produce nell'esecuzione alcun fattore esterno, come ad esempio delle interferenze, che portino alla perdita della sequenza, oppure inducano il PC a puntare a posizioni sbagliate della memoria, ecc.

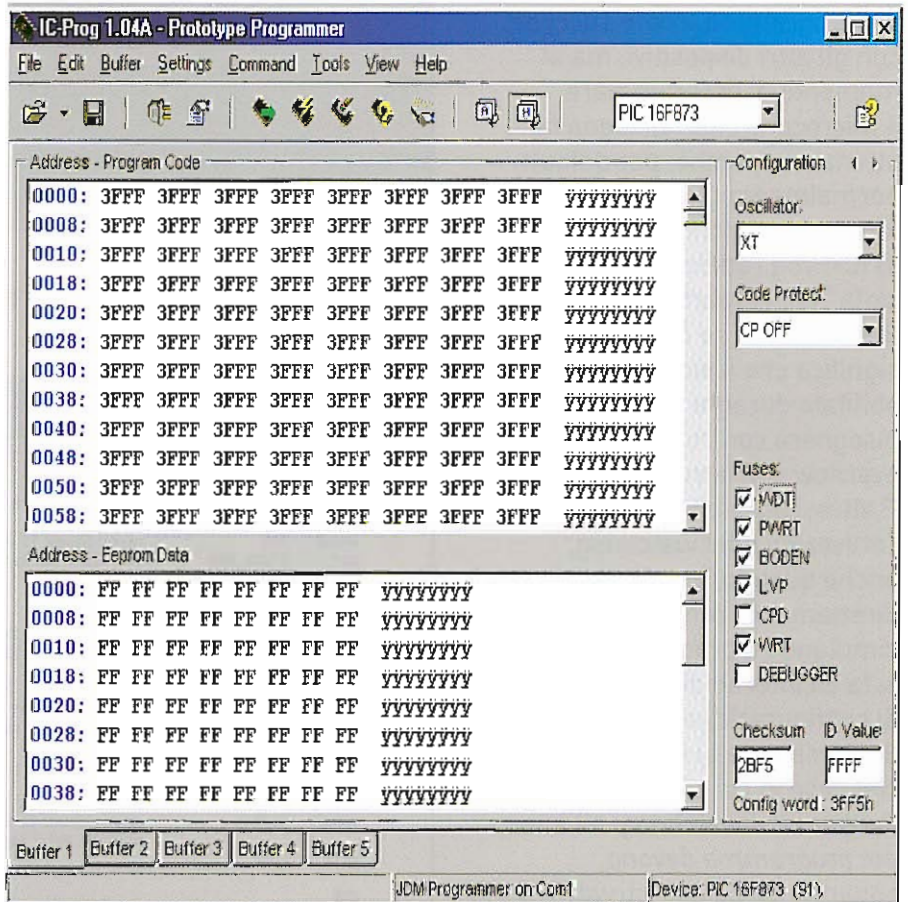
A questo scopo il watchdog si incrementa e aspetta, prima di arrivare al suo valore massimo (FF) e ricominciare, che fra le istruzioni del programma ne vengano eseguite anche alcune "per lui", nello specifico la "clrwdt", che il programmatore dovrà inserire in modo opportuno fra il normale codice del programma. In un programma ben fatto e con una esecuzione corretta, ogni volta che si passa su di una istruzione "clrwdt" il watchdog inizia nuovamente da 0 e ricomincia il suo conteggio dall'inizio. Mediante queste istruzioni bisogna controllare che non arrivi mai a overflow (passaggio da FF a 00), dato che altrimenti produrrà un reset del sistema. L'unico caso in cui si deve produrre un reset è se realmente esiste qualche problema. Se, ad esempio il PC ha trovato un valore sbagliato, non verrà eseguita alcuna istruzione "clrwdt", quindi arriverà il momento in cui il watchdog andrà in overflow e resetterà

MNEMONICO	PARAMETRI	SIGNIFICATO
clrw		Imposta a valore 0 il registro di lavoro W.

Utilizzo dell'istruzione "clrw".

MNEMONICO	PARAMETRI	SIGNIFICATO
clrwdt		Imposta a valore 0 il watchdog

Utilizzo dell'istruzione "clrwdt".



Il watchdog si abilita e disabilita al momento della scrittura.

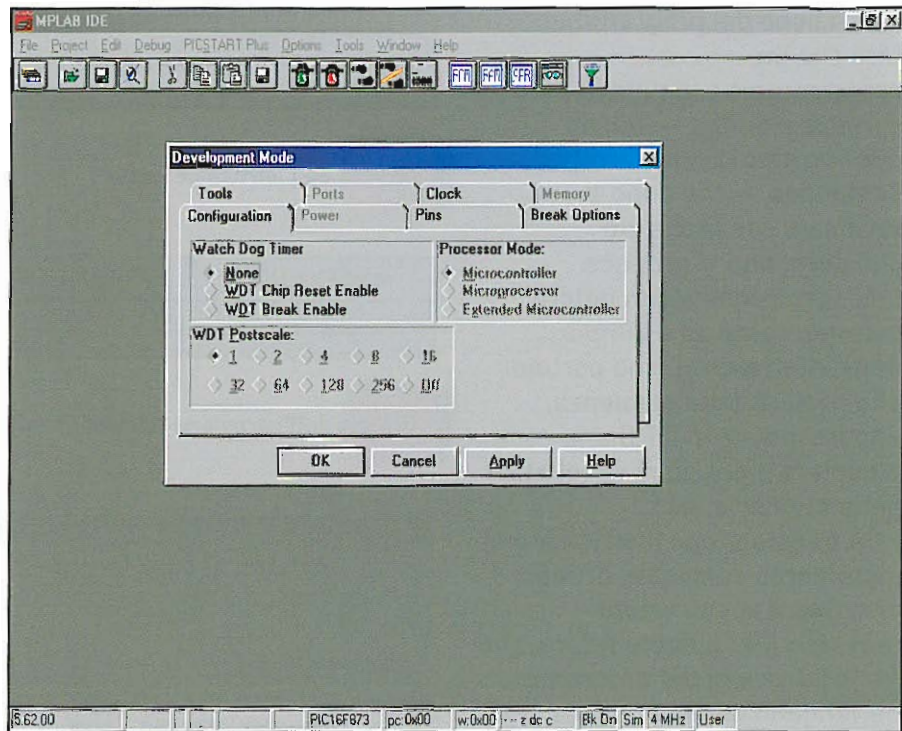
il sistema, in modo che il programma ritorni alla sua esecuzione normale.

Per controllare la velocità di conteggio del watchdog

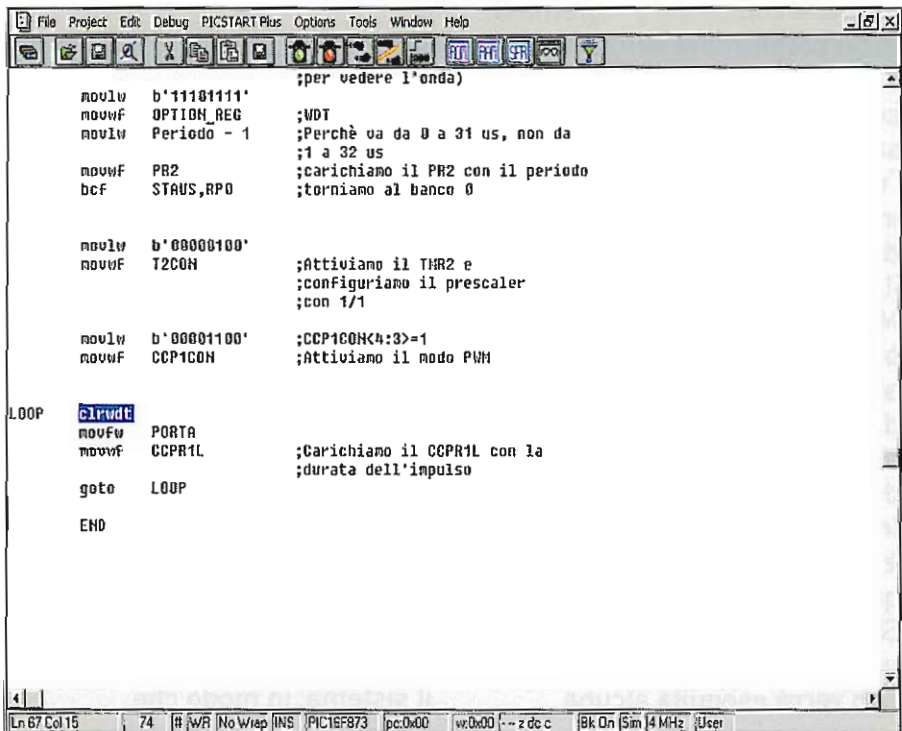
è possibile assegnare il divisore di frequenza, cosa che si fa nel registro OPTION, che abbiamo già visto parlando del TMR0.

Abilitazione e disabilitazione del watchdog e suo controllo

Anche se l'utilizzo del watchdog è molto utile nei sistemi a microcontroller, in cui è necessario non perdere mai il controllo del programma, viste le conseguenze pericolose che questo potrebbe supporre, è possibile disabilitarlo. Tuttavia questo non si fa all'interno del programma, come succede con gli altri dispositivi, ma al momento di programmare il microcontroller. Bisogna far attenzione perché, per default, normalmente gli scrittori tengono il watchdog attivato. In IC-Prog l'opzione appare con la sigla "WDT" (parte destra della finestra) che, se è selezionato significa che il watchdog verrà abilitato durante la scrittura, bisognerà controllare il suo overflow con istruzioni "clrwdt". È altresì possibile simulare l'attivazione del watchdog, anche quando, invece di lavorare direttamente con il PIC, si sta simulando il programma. Questo si fa all'interno di una finestra di configurazione delle opzioni dell'ambiente di sviluppo. Nel caso in cui sia stato abilitato l'utilizzo del watchdog, all'interno del programma devono apparire istruzioni "clrwdt" in modo periodico. Normalmente i cicli di temporizzazione o di attesa di valori contengono questo tipo di istruzioni. Se dopo averlo abilitato decidiamo di non utilizzarlo, dobbiamo nuovamente programmare il microcontroller con l'opzione corrispondente.



Anche nella simulazione è possibile scegliere se utilizzare o meno il watchdog.



Programma che utilizza il watchdog.