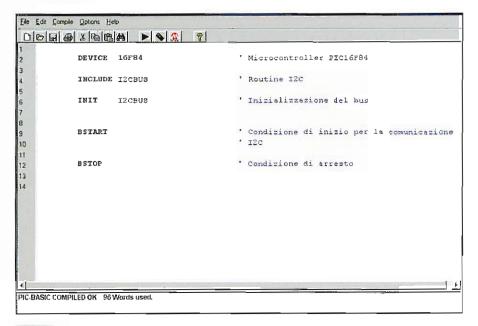
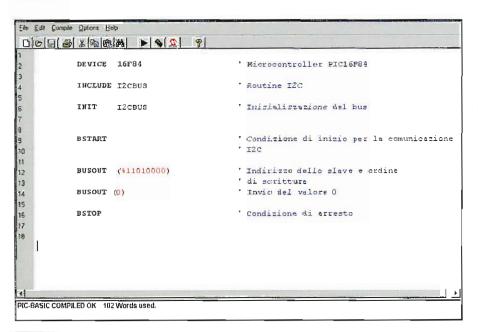
Basic per PIC



L'ultima istruzione sarà quella della condizione di stop.



Il primo BSOUT serve per indirizzare lo "slave".

poco dopo quella di stop, quindi deve esistere un sistema per fare in modo che i "master" non cerchino di realizzare una comunicazione se nel frattempo il bus è già occupato da un altro.

Configurazione del canale

Così come succedeva con le istruzioni per la comunicazione seriale con un computer,

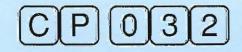
quelle di comunicazione con il bus I2C sono progettate per il lavoro con il PIC16F84. Questo PIC non dispone di risorse hardware dedicate a guesto scopo, di conseguenza la comunicazione viene fatta via software. I comandi INCLUDE e INIT serviranno per includere le routines di funzionamento, in questo caso del bus I2C, e per inizializzarlo. Bisogna ricordare che le istruzioni 12C utilizzano piedini fissi, che sono il bit 0 della porta A come linea SDA e il bit 1 della porta A come linea SCL, Queste linee sono configurate automaticamente come uscite. Se la porta A si utilizza nello stesso programma per la comunicazione con il protocollo RS-232, il comando INIT I2CBUS deve essere successivo a INIT SERIAL.

Istruzione di inizio

Dopo aver inizializzato il canale, per iniziare la comunicazione la prima istruzione sarà la condizione di inizio BSTART. Come si può vedere dalla figura, questa istruzione non ha parametri, quindi il suo utilizzo è molto semplice.

Istruzione di stop

L'istruzione per la condizione di stop è simile alla precedente. È sufficiente porre BSTOP, senza parametri come mostrato nella figura. Verrà terminata in questo modo la comunicazione I2C. Per riattivarla bisogna nuovamente generare una condizione di inizio.



Basic per PIC

Invio dei dati

L'istruzione per l'invio dei dati è BSOUT. L'unico parametro di cui ha bisogno, posto fra parentesi, può essere un numero, una variabile o un'espressione. Il primo byte che si invia corrisponde all'indirizzo dello "slave" con cui si vuole realizzare la comunicazione, e il bit meno significativo indica se si vuole scrivere (0), oppure leggere (1).

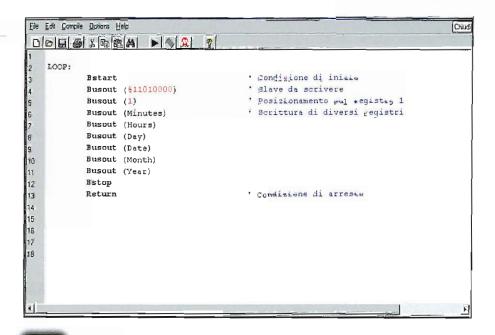
La quantità di byte inviati, il loro significato e il loro numero dipenderanno dal dispositivo con cui si sta realizzando lacomunicazione.

Ricezione dei dati

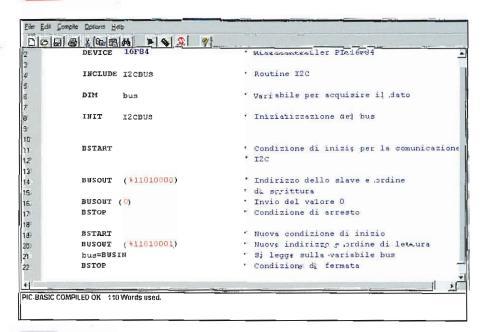
Con l'istruzione BUSIN si riceve un dato del bus. È necessario definire una variabile sulla quale scrivere questo dato per il suo successivo trattamento. Inoltre, in precedenza, è stato inviato un byte per indirizzare lo "slave" e informarlo che si vuole realizzare una lettura.

possono essere diverse istruzioni di inizio e di stop, di fatto sarà necessario chiudere una comunicazione e tornare ad aprirla quando si desidera passare da scrittura a lettura o viceversa.

In uno stesso programma ci



Il significato dei byte dipenderà dal dispositivo.



Esempio di ricezione dei dati.