di valore. In questo caso non fa differenza se la periferica è collegata a livello alto o basso, dato che si attende un cambio di livello da alto verso il basso o viceversa. Come parametro dell'istruzione si può utilizzare il numero di pin all'interno della porta, o la sua definizione simbolica.

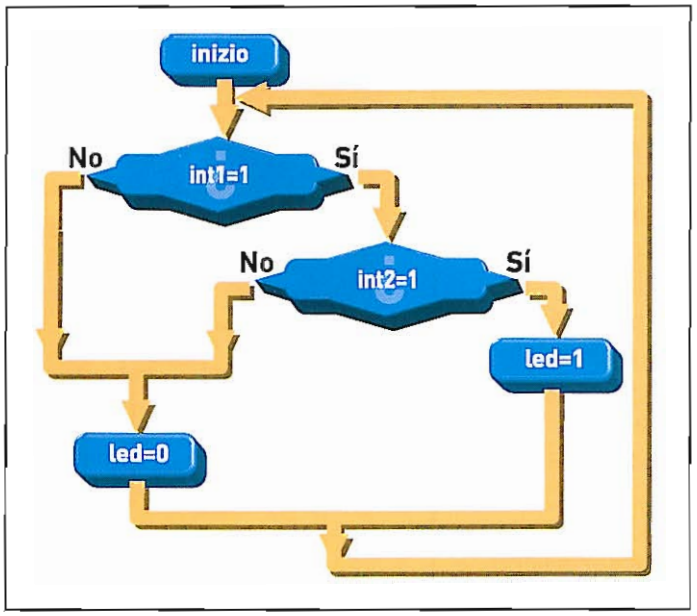
```
File  Edit  Compile  Options  Help

        Device 16F84                    ' si definisce il PIC16F84

        Define PORTB=%11111111          ' Si dichiara la PortaB come ingresso
        Define PORTA=%00000000          ' Si dichiara la PortaA come uscita

        Symbol LED=A.0                  ' Si assegna un nome al bit 0 della portaA

If PORTB&1=1 then Set LED               ' Se la condizione si compie
                                        ' il bit 0 della portaA si pone a 1

If PORTB&2=1 then Set A.1               ' Se la condizione si compie
                                        ' il bit 1 della portaA si pone a 1

If PORTB&8=1 then Clear A.4             ' Se la condizione si compie
                                        ' il bit 4 della portaA si pone a 0

END

PIC BASIC COMPILED OK  35 Words used.
```

Verifica di diversi interruttori con struttura IF...THEN.

Richiesta di vari bit individuali però consecutivi

I programmi trattati sinora in questo capitolo sono semplici, però non capita spesso di avere un solo interruttore o un solo pulsante collegati, di solito ce ne sono diversi, rendendo necessaria la verifica del valore di ognuno di essi sia separatamente che insieme. Analizziamo la figura che mostra un programma in cui si ricerca un valore di diversi interruttori, mediante l'istruzione "button". Credete sia corretto? Se avete capito bene come funziona questa istruzione vi renderete conto che non è neccessario, per far sì che una delle azioni si compia, attendere che cambino i valori dei bit precedenti.

Normalmente la verifica viene fatta per ogni pin di ingresso, e ogni verifica è inserita nella propria struttura IF...THEN, che realizza l'azione se la condizione si compie, oppure passa oltre non eseguendo alcuna azione in caso contrario; in ogni caso la sequenza del programma viene eseguita. Se invece di considerare ogni pin di ingresso come portatore di un valore indipendente, si desidera che fra loro formino un codice, per poter

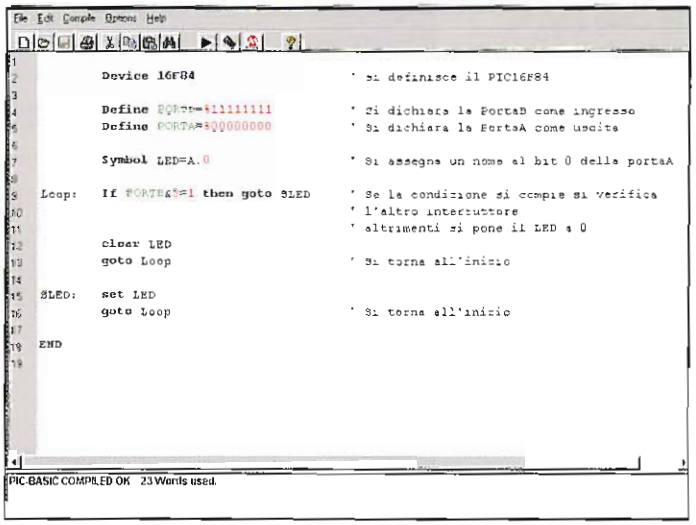


Organigramma per gestire diversi interruttori allo stesso tempo.

```
File  Edit  Compile  Options  Help

        Device 16F84                    ' si definisce il PIC16F84

        Define PORTB=%11111111          ' Si dichiara la PortaB come ingresso
        Define PORTA=%00000000          ' Si dichiara la PortaA come uscita

        Symbol LED=A.0                  ' Si assegna un nome al bit 0 della portaA

Loop:   If PORTB&1=1 then goto INT2     ' Se la condizione si compie si verifica
                                        ' l'altro interruttore
        goto NOINT                      ' altrimenti si pone il LED a 0

INT2:   If PORTB&2=1 then goto SLED     ' Se la condizione si compie si pone
                                        ' il LED a 1
                                        ' altrimenti si pone il LED a 0
NOINT:  clear LED
        goto Loop                       ' Si torna all'inizio

SLED:   set LED
        goto Loop                       ' Si torna all'inizio

END

PIC BASIC COMPILED OK  32 Words used.
```

Programma per gestire diversi interruttori contemporaneamente con strutture IF...THEN.

avere così un range maggiore di valori, si può procedere in diversi modi. Uno di essi è rappresentato dall'organigramma della figura e si trasformerà in strutture IF...THEN annidate. Come si può vedere, con due interruttori il programma inizia a essere più complesso. Di solito, soprattutto quando i valori di ingresso fanno parte della stessa porta, si utilizzano le già note maschere, per eseguire delle verifiche con i possibili valori.

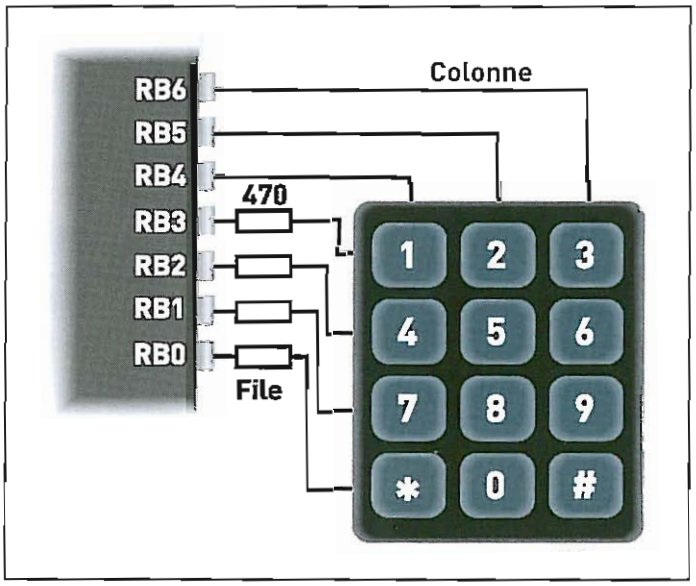


Programma per gestire diversi interruttori contemporaneamente utilizzando le maschere.

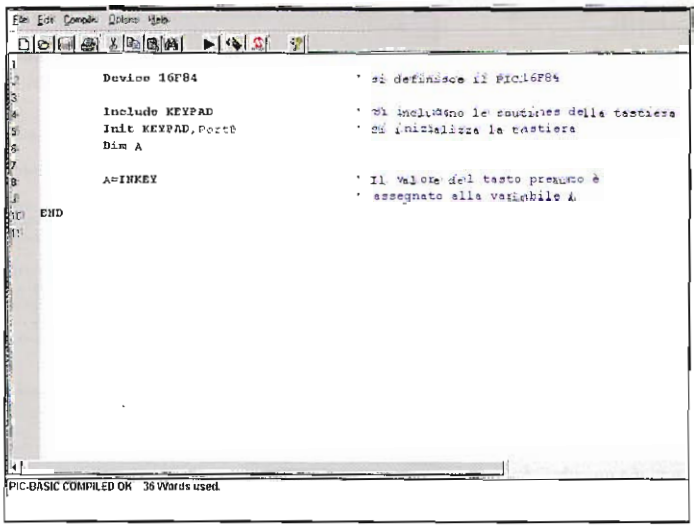
Un caso speciale di ingresso

Che cosa succede se come periferica di ingresso abbiamo una tastiera? Una tastiera non è altro che un insieme di pulsanti, posti in forma matriciale e collegati in un modo particolare. Per una tastiera a 16 pulsanti, se venissero collegati come pulsanti normali sarebbero necessarie 16 linee di ingresso, cosa impossibile nel PIC16F84. Invece, se si collegano in forma di file e di colonne, le linee necessarie sono il numero di file più il numero di colonne: per una tastiera da 16 tasti si convertiranno così in 8 linee, e per una tastiera da dodici, come quella della figura, sono sufficienti 7 linee.

Per poter utilizzare la tastiera, è necessario includere il pacchetto di routine dedicato a questa periferica, e inizializzarlo assegnandolo a una porta. Con l'istruzione "INKEY" viene portato su una variabile il valore del pulsante premuto. Utilizzeremo questa periferica in programmi che vedremo più avanti.



Collegamento di una tastiera matriciale al PIC.



Esempio di un'acquisizione di dati dalla tastiera.