

GESTIONE INGRESSI ED USCITE DIGITALI NEL MICROCONTROLLORE PIC

Nel programma qui a lato, possiamo vedere una semplice gestione di bit digitali come ingresso (su PORTA) e come uscita (su PORTB).

Dopo le direttive per il compilatore, troviamo il codice del programma scritto nella memoria del microcontrollore.

La prima parte serve per definire la direzione dei pin delle due porte PORTA e PORTB, tramite i due registri TRISA e TRISB.

Per poter scrivere in TRISA e TRISB, devo prima spostarmi nel banco di memoria 1, mettendo ad uno il bit RP0 di STATUS.

Poi per scrivere in PORTA e PORTB devo tornare nel banco di memoria 0, mettendo a zero il bit RP0 di STATUS, il bit RP1 del registro STATUS rimane sempre a 0.

File Address	File Address	File Address	File Address
Indirect addr.(*) 00h	Indirect addr.(*) 80h	Indirect addr.(*) 100h	Indirect addr.(*) 180h
TMR0 01h	OPTION_REG 81h	TMR0 101h	OPTION_REG 181h
PCL 02h	PCL 82h	PCL 102h	PCL 182h
STATUS 03h	STATUS 83h	STATUS 103h	STATUS 183h
FSR 04h	FSR 84h	FSR 104h	FSR 184h
PORTA 05h	TRISA 85h	WDTCON 105h	
PORTB 06h	TRISB 86h	PORTB 106h	TRISB 186h
PCLATH 0Ah	PCLATH 8Ah	PCLATH 10Ah	PCLATH 18Ah

STATUS: ARITHMETIC STATUS REGISTER (ADDRESS 03h, 83h, 103h, 183h)

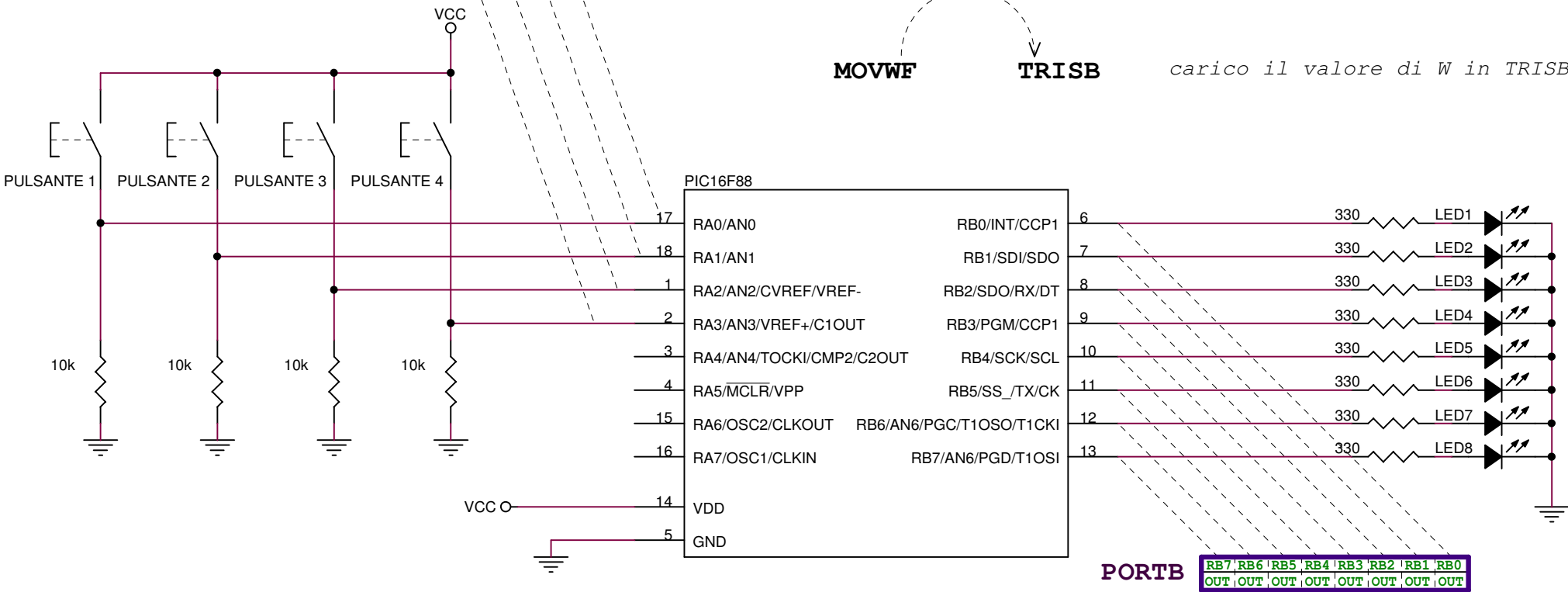
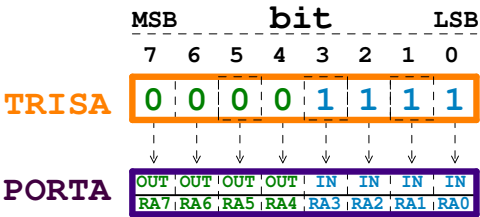
R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	TO	PD	Z	DC	C
bit 7							bit 0

RP1 – bit 6 di STATUS	RP0 – bit 5 di STATUS	Banco di memoria accessibile
1	1	Bank3 (180 Hex – 1FF Hex)
1	0	Bank2 (100 Hex – 17F Hex)
0	1	Bank1 (80 Hex – FF Hex)
0	0	Bank0 (0 Hex – 7F Hex)

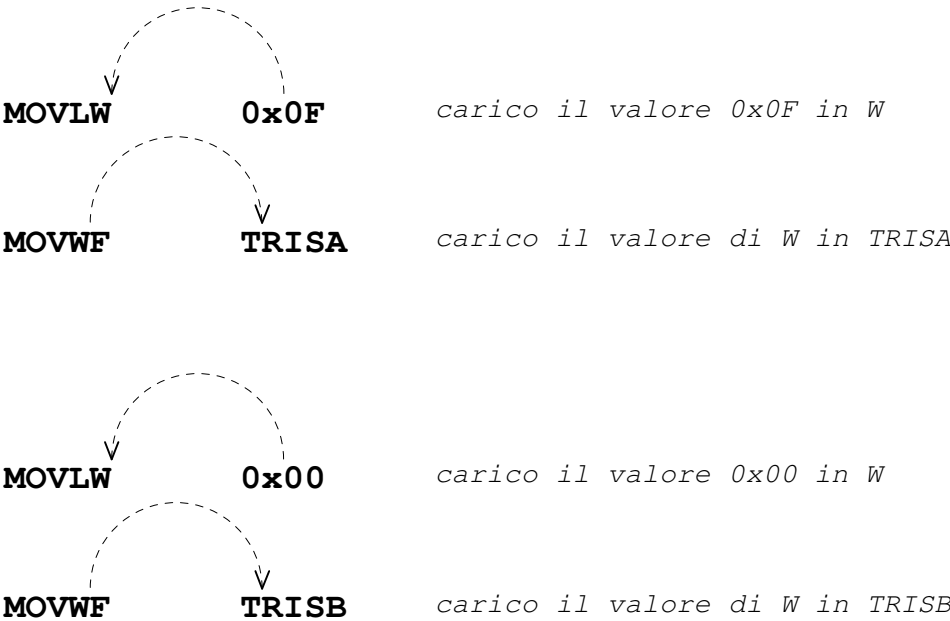
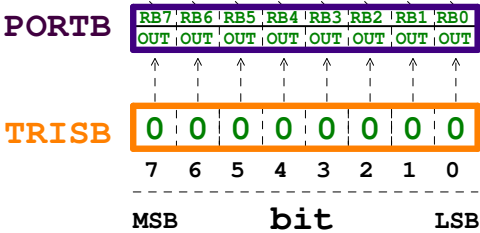
```
*****  
;DIRETTIVE PER IL COMPILATORE/ASSEMBLATORE  
  
#include "P16F88.INC" ;definizione del tipo di microcontrollore  
;viene incluso un file che contiene le  
;associazioni tra indirizzo fisico e nome  
;delle porte e dei bit del microcontrollore  
  
;DEFINIZIONE DI COSTANTI  
  
LED EQU 0 ;si assegna alla label LED1 la corrispondenza  
;al valore 0. LED non è un variabile.  
  
PULS EQU 0 ;si assegna alla label PULS la corrispondenza  
;al valore 0. PULS non è un variabile.  
  
*****  
;PROGRAMMA CONTENUTO NELLA FLASH  
  
ORG 0x00 ;inizio del programma, il microcontrollore  
;inizierà a decodificare le istruzioni  
;partendo dalla locazione 0x00 della FLASH  
  
GOTO Inizio ;prima istruzione, salto alla locazione  
;indicata dalla label "Inizio"  
  
ORG 0x100 ;locazione di memoria dove è scritto il programma  
  
Inizio ;label inizio, posizionata alla riga 0x100  
;scelta del banco di memoria dove scrivere, BANK1  
  
BSF STATUS,RP0 ;metto ad 1 il bit RP0 del registro STATUS  
BCF STATUS,RP1 ;metto ad 0 il bit RP1 del registro STATUS  
MOVLW 0x0F ;carico il valore 0x0F in W  
MOVWF TRISA ;copio il contenuto di W nel registro TRISA  
MOVLW 0x00 ;carico il valore 0x00 in W  
MOVWF TRISB ;copio il contenuto di W nel registro TRISB  
  
;scelta del banco di memoria dove scrivere, BANK0  
BCF STATUS,RP0  
BCF STATUS,RP1  
  
;loop del programma  
MainLoop ;label che identifica il loop del programma  
BTFSC PORTA,PULSANTE ;test bit0 di PORTA se 0 salto prossima riga  
GOTO LOOP1 ;salto alla label SetToZero  
BCF PORTB,LED ;metto ad uno il bit 0 di PORTB  
GOTO MainLoop ;resetto il bit 0 di PORTB  
;salto alla label MainLoop  
  
LOOP1  
  
END ;fine codice
```

Di seguito la descrizione del funzionamento dei registri TRIS, e la descrizione delle istruzioni per gestire in ingresso ed uscita i singoli ingressi digitali nel linguaggio assembler del microcontrollore PIC.

Scivendo 000111 in binario oppure 0F in esadecimale sul registro TRISA, imposto i 4 bit meno significativi di PORTA come ingressi



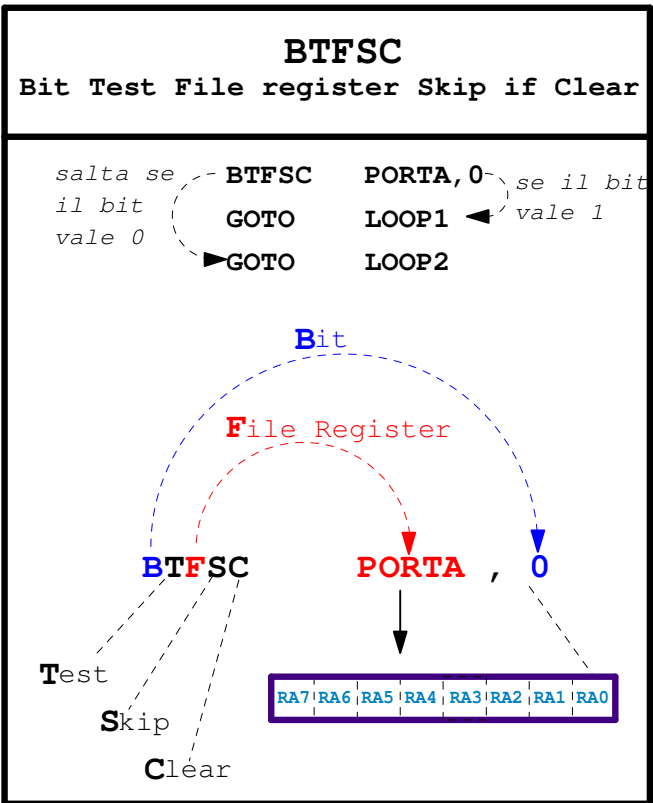
Scivendo 0000000 sul registro TRISB, imposto gli 8 bit meno significativi di PORTB come uscite



TEST DI UN INGRESSO

In questo esempio ci sono 4 pulsanti collegati, supponiamo di voler testare lo stato del PULSANTE 1 collegato su RA0.

Per testare un bit di un qualsiasi registro si può utilizzare una delle due istruzioni qui a fianco.



SET E RESET DI UN USCITA

In questo esempio ci sono 8 LED collegati, supponiamo di voler accendere o spegnere il LED1 collegato ad RB0

Per settare o resettare un bit si può utilizzare una delle due istruzioni qui a fianco.

