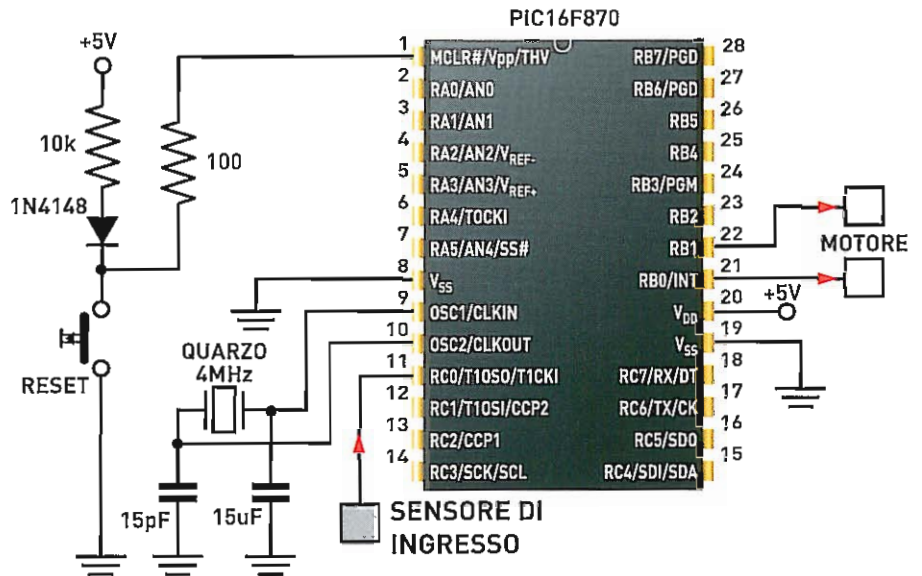


IL TMR1 come contatore

Così come il TMR0, anche il TMR1 può lavorare in modo contatore utilizzando gli impulsi applicati a uno dei piedini del PIC. Questo tipo di funzionamento è conosciuto come modo "contatore asincrono", poiché non si conosce quando si verificano gli impulsi, in quanto non sono generati a intervalli costanti (in modo sincrono) ma dipendono da fattori esterni, ad esempio sensori, o dall'utente stesso che sta utilizzando il sistema. Questo è il caso che tratteremo nel presente capitolo; il terzo modo di funzionamento riportato nella figura per il TMR1, non è un modo tipo contatore, e lo studieremo in un capitolo successivo.



Schema elettronico dell'esercizio proposto.

Differenze con il contatore del TMR0

Oltre al fatto che il TMR0 è un temporizzatore da 8 bit mentre il TMR1 è un temporizzatore da 16,

quando il TMR0 lavora in modo contatore e il TMR1 come contatore asincrono, fra loro esiste solo una differenza: mentre il primo può essere configurato in modo che l'incremento sia prodotto o con il fronte di salita o con il fronte di

discesa, con il TMR1 è possibile lavorare solamente con il fronte di salita. Questa limitazione tuttavia non comporta alcun problema nella stesura dei nostri programmi.

Definizione del problema

Bisogna controllare la capienza di un locale in modo che la porta rimanga aperta sino a che non si raggiunge un numero limite di persone, e si chiuda quando questo limite viene raggiunto. Il problema non è un caso del tutto reale, dato che normalmente se qualcuno abbandona il locale bisognerebbe diminuire il conteggio e questo non è possibile con i temporizzatori, i quali incrementano solamente.

TMR0 (8 bits)	CONTATORE (fronte di salita/discesa)	
	TEMPORIZZATORE	
TMR1 (16 bits)	CONTATORE (fronte di salita)	SINCRONO
	TEMPORIZZATORE	ASINCRONO
TMR2 (8 bits)	TEMPORIZZATORE	

Tuttavia ce ne serviremo per vedere il modo con cui si lavora con i contatori in assembler, perché quando abbiamo visto il TMR0 utilizzavamo le istruzioni Basic.

Schema elettronico

Come avevamo visto all'epoca, il TMR0 utilizza sempre lo stesso piedino per acquisire gli impulsi, cioè RA4. Quando il TMR1 lavora in modo contatore ha assegnato il piedino RC0. Oltre a questa linea, che dovrà essere definita come ingresso, e alla parte indispensabile di circuito, che ha sempre lo stesso collegamento, nello schema sono riportate anche due linee, le quali faranno girare un motore in un verso o nell'altro per aprire o chiudere la porta. Normalmente sarà necessario un dispositivo chiamato driver che farà da intermediario fra il PIC e il motore, per amplificare il segnale fra entrambi. Il tipo di driver dipenderà dal motore. Nello schema i segnali del motore sono stati collegati ai piedini RB0 e RB1, anche se avremmo potuto operare una scelta diversa. Con la linea RB0 si attiverà e disattiverà il motore e con RB1 se ne cambierà il verso di rotazione, così come mostrato nella tabella della figura.

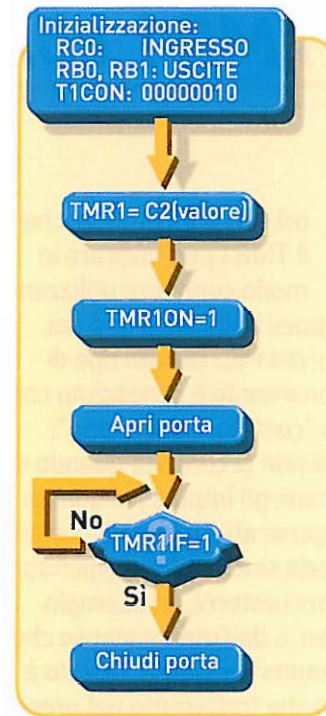
RB0	RB1	MOTORE
0	0	Disattivato
0	1	Disattivato
1	0	Attivato verso A
1	1	Attivato verso B

Tabella di funzionamento del motore.

Organigramma e programma commentato

Il programma della figura rappresenta i passi che dovremmo fare per costruire il programma in assembler. Saranno definiti i registri di ingresso e uscita.

Il registro di configurazione del TMR1, il T1CON, verrà caricato con il valore che è indicato, per fare in modo che si comporti come contatore di impulsi esterni. La tabella riportata in basso spiega con quale criterio si genera il valore da caricare sul T1CON. Il TMR1 deve essere caricato con il complemento a due del valore che si vuole contare. Si procede in questo modo perché c'è un bit, il TMR1IF, che viene impostato a 1 quando il temporizzatore va in overflow, cioè quando arriva al suo valore massimo (FFFF) e poi ritorna a 0. Quindi il conteggio non può iniziare da 0. Attiveremo il TMR1 e apriremo la porta, che consisterà nell'attivare il motore in uno dei due sensi per un certo periodo. Quando si rileva che il bit di overflow è stato attivato, significherà che il locale sarà pieno e quindi dovremo chiudere la



Organigramma dell'esercizio proposto.

porta, ovvero tornare ad attivare il motore in senso contrario a quello precedente, per un determinato tempo. I tempi di apertura e chiusura della porta non sono stati sviluppati, si potrebbe trattare di un'altra routine con un contatore manuale o con un altro temporizzatore, ad esempio il TMR0.

-	-	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC#	TMR1CS	TMR1ON
7	T1CON						0
<ul style="list-style-type: none"> T1CKPS1-0: 00. Valore minimo del divisore di frequenza [1/1] T1OSCEN: 0. Il TMR1 si comporta come contatore di impulsi asincroni T1SYNC#: 0. Ininfluenza se il TMR1 funziona come contatore di impulsi asincroni. TMR1CS: 1. Utilizzo di clock esterno come contatore TMR1ON: 0. Temporizzatore scollegato; dovremo impostarlo a 1 quando vogliamo che inizi a contare. 							

Selezione di un valore sul registro T1CON.