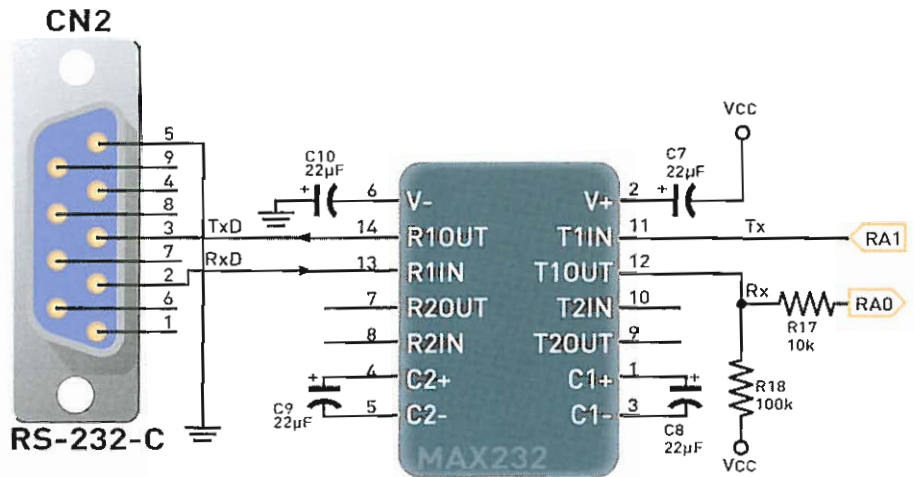


Comunicazione seriale dei dati

I microcontroller PIC sono progettati per lavorare come sistemi chiusi, cioè non hanno bisogno di comunicare con altri sistemi per realizzare il loro lavoro, ma solamente con i dispositivi di ingresso e di uscita.

Tuttavia, a volte, può essere interessante che il PIC si relazioni con altri sistemi "intelligenti". In questo capitolo lavoreremo con una di queste forme di comunicazione: la comunicazione seriale.



Collegamenti del MAX232 al PIC e al connettore DB9.

Perché la comunicazione seriale?

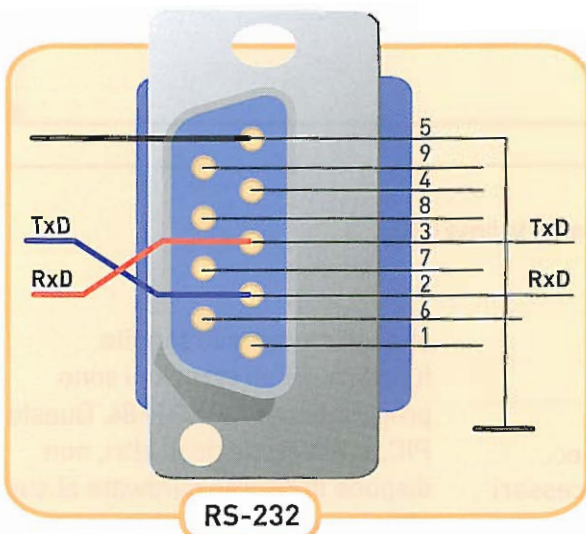
La comunicazione seriale fra sistemi e circuiti integrati è un modo semplice e pratico di trasferire dati digitali. Semplice perché il LetPicBasic possiede le istruzioni necessarie per fare in modo che il programmatore debba solo preoccuparsi dei dati che

vuole trasferire; pratico, perché con sole due o tre linee si possono ampliare le possibilità di funzionamento del microcontroller.

La comunicazione seriale, quindi, può essere un buon metodo per trasferire dati fra un PIC e un computer, oppure fra un PIC e delle memorie, dei convertitori, ecc.

Protocollo di comunicazione

Anche se esistono diversi tipi di comunicazione seriale, le istruzioni che ci fornisce il LetPicBasic sono specifiche per la comunicazione seriale dei byte mediante il protocollo RS-232, senza parità e con un bit di stop, 8-N-1. Se si sta lavorando con un PIC a 4 MHz la velocità di trasferimento sarà di 9.600 baud, e questa verrà raddoppiata a 19.200 se il quarzo è da 8 MHz. Dato che è il linguaggio di programmazione a stabilire i parametri, per lavorare con esso dovremo adattare le caratteristiche del dispositivo con cui stiamo comunicando perché siano compatibili.



Cavo di connessione con linee incrociate.

Comunicazione con il PC

Se l'altro dispositivo è un computer, non sarà sufficiente realizzare il programma, ma dovremo anche aggiungere una parte di hardware. Questo è dovuto al fatto che i livelli elettrici con cui lavora il PIC non sono gli stessi di quelli che usa il computer, sarà necessario quindi un circuito integrato per fare questa "traduzione". Questo circuito è il MAX232, e nella figura della pagina precedente è indicato come collegarlo. Il PIC, secondo lo schema, trasmetterà i dati tramite la linea RA1. Ogni segnale passerà al MAX232, tramite il piedino 11 con livello TTL, che sarà trasformato dal circuito e trasmesso dal piedino 14 con livello RS-232, per passarlo al computer. I dati in arrivo dal computer passeranno per il piedino 13 con livelli RS-232, che saranno trasformati a livello TTL, e usciranno dal piedino 12 verso il PIC, entrando in RA0.

Cavo di comunicazione

Se un dispositivo riceve un dato è perché un altro lo ha trasmesso, e viceversa. Dato che esistono due linee di dati indipendenti, questo fa sì che le linee di comunicazione debbano essere incrociate, così come è mostrato nella figura. In essa possiamo vedere anche il connettore DB9, nel caso in cui la comunicazione sia con un PC; per quanto riguarda il resto dei dispositivi, l'incrocio di cavi deve essere effettuato nello stesso modo.

```
File Edit Compile Options Help
PIC-BASIC COMPILER

1
2 DEVICE 16F84
3
4 INCLUDE SERIAL
5 DEFINE PORTA=00000001          * Configurazione delle linee come I/O
6
7 SYMBOL Rin=A.0                 * Linea di ricezione
8 SYMBOL Rout=A.1                * Linea di trasmissione
9 SYMBOL Dtr=A.2
10
11 INIT SERIAL Rin, Rout, Dtr    * Inizializzazione
12
13
14
15
16
17
18
19
20
21
22
23
PIC-BASIC COMPILED OK 49 Words used.
```

Comandi di configurazione del canale.

```
File Edit Compile Options Help
PIC-BASIC COMPILER

1
2 DEVICE 16F84
3
4 INCLUDE SERIAL
5 DEFINE PORTA=00000001          * Configurazione delle linee come I/O
6
7 SYMBOL Rin=A.0                 * Linea di ricezione
8 SYMBOL Rout=A.1                * Linea di trasmissione
9
10 INIT SERIAL Rin, Rout         * Inizializzazione
11
12
13
14
15
16
17
18
19
20
21
22
23
PIC-BASIC COMPILED OK 47 Words used.
```

Configurazione del canale senza la linea DTR.

Configurazione del canale

Come sempre, inizieremo mostrando i comandi necessari per fare in modo che le istruzioni

che utilizzeremo in seguito funzionino. Tali istruzioni sono progettate per il PIC16F84. Questo PIC, a differenza degli altri, non dispone di risorse hardware al suo interno per realizzare la