

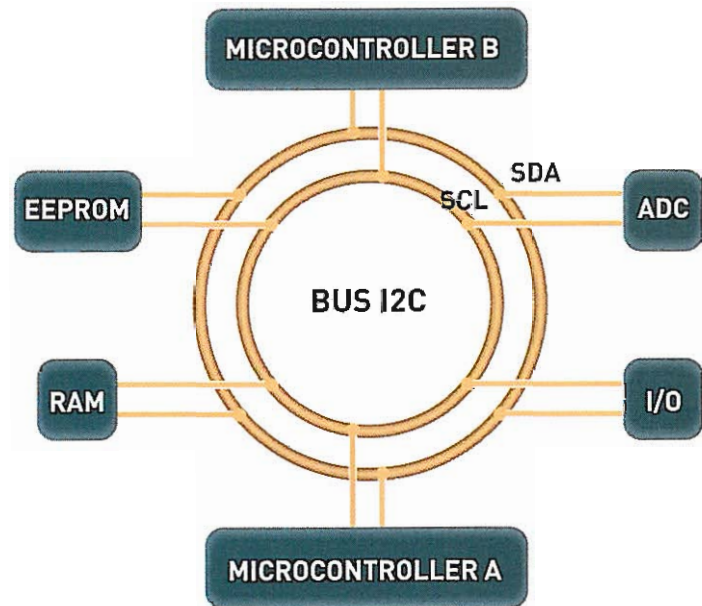
Comunicazione mediante il bus I2C

Se la comunicazione mediante il protocollo RS-232 vi è sembrata interessante, aspettate di vedere ciò che vi presentiamo in questo capitolo.

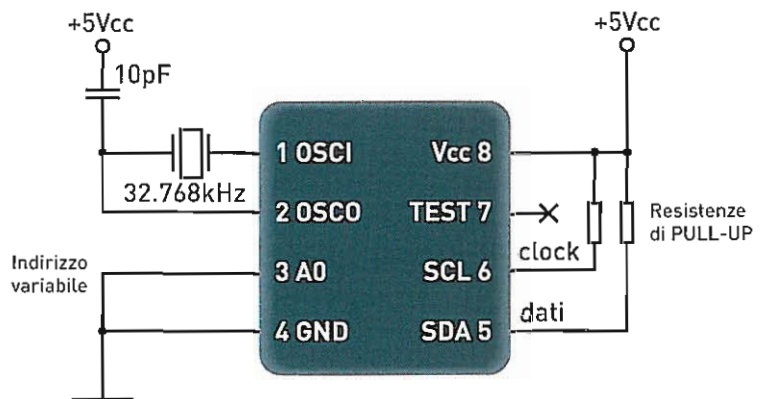
Utilizzeremo le possibilità che offre il LetPicBasic per mettere in comunicazione il PIC con una moltitudine di dispositivi tramite due sole linee. Questo tipo di comunicazione seriale è più diffuso del precedente e l'utilizzo dell'uno o dell'altro sarà condizionato dal dispositivo con cui si vuole realizzare questa comunicazione.

Caratteristiche del bus I2C

Questo protocollo fu sviluppato dalla Philips per la comunicazione fra grandi quantità di circuiti integrati di propria produzione, come memorie, clock in tempo reale, convertitori, microprocessori, ecc. La sua caratteristica più interessante consiste nell'utilizzare unicamente due linee del PIC, per i collegamenti con tutti i dispositivi con cui comunica. Una delle linee chiamata SDA, supporterà i dati in entrambe le direzioni (da e verso il PIC), e l'altra SCL, servirà per veicolare gli impulsi di clock che rendono possibile la sincronizzazione.



Sistema multimaster con diversi dispositivi collegati al bus.



Clock in tempo reale con un bit di indirizzo variabile.

Gli impulsi di clock sono generati da un PIC denominato "master", che è quello che controlla la comunicazione, decidendo in ogni momento con quale dispositivo bisogna

comunicare e se si inviano o si ricevono dati. Ogni elemento ha un indirizzo che lo identifica, ed è considerato uno "slave". In un bus, tuttavia, si possono trovare più di un "master".



Passi della comunicazione

Il microcontroller "master" è l'unico che può iniziare la comunicazione.

A questo proposito, per prima cosa, genererà una condizione chiamata "di inizio" dopo di che dovrà avvenire il processo di indirizzamento, che consiste nell'invviare un codice per selezionare un determinato "slave".

Gli "slaves" collegati al bus confrontano l'indirizzo inviato con il proprio, e se coincidono sanno di essere stati selezionati. Normalmente un indirizzo è formato da 7 bit, una parte dei quali fissi, e l'altra

programmabile dall'utente, in modo da poter collegare dispositivi uguali sullo stesso bus, cambiando l'indirizzo variabile degli stessi. Secondo il numero di bit variabili si potranno collegare più o meno dispositivi dello stesso tipo.

Il clock-calendario in tempo reale della figura della pagina precedente ha un unico bit variabile, di conseguenza sullo stesso bus, ne potranno coesistere solamente due, uno con l'indirizzo 0 e l'altro con 1. Nell'esempio è stato configurato con indirizzo 0 collegando a massa la relativa linea. Notate, inoltre, che le due linee di comunicazione, SDA e SCL, sono collegate a 5 V mediante una resistenza di pull-up.

Questo è dovuto a una specifica del protocollo I2C.

Esiste un indirizzo speciale per "chiamare" tutti i dispositivi nello stesso tempo.

Il dato successivo a quello

```
File Edit Compile Options Help
[Icons]
1
2     DEVICE 16F84           * Microcontroller PIC16F84
3
4     INCLUDE I2CBUS        * Routine I2C
5
6     INIT I2CBUS           * Inizializzazione del bus
7
PIC-BASIC COMPILED OK 94 Words used.
```

Comandi di configurazione del canale.

```
File Edit Compile Options Help
[Icons]
1
2     DEVICE 16F84           * Microcontroller PIC16F84
3
4     INCLUDE I2CBUS        * Routine I2C
5
6     INIT I2CBUS           * Inizializzazione del bus
7
8
9     BSTART                * Condizione di inizio per la comunicazione
10                    * I2C
11
```

La prima istruzione sarà quella della condizione di inizio.

dell'indirizzamento indicherà l'operazione da realizzare. Se il "master" vuole inviare dati, lo farà in questo momento, se invece li deve ricevere si metterà in stato d'attesa.

Il "master" sarà anche il dispositivo che termina la comunicazione inviando la "condizione di stop".

Il bus si considera occupato fra la condizione di inizio e sino a