

Comunicazione I2C fra PIC (I)

In un capitolo precedente abbiamo già parlato dei passaggi necessari per far comunicare un microcontroller PIC con diversi dispositivi mediante il suo modulo MSSP nel modo I2C. Però nel caso ci fossero diversi PIC che tendono ad avere il controllo della comunicazione, oppure se il PIC comunica con un altro PIC invece che con un dispositivo? Vediamo come bisogna fare.

Il PIC come slave

Trattandosi di I2C, uno slave è un dispositivo indirizzato da un master. Secondo questa definizione, se un PIC è indirizzato da un altro questo PIC si sta comportando come uno slave, almeno nel momento particolare di quella comunicazione; quindi mentre il master deve sempre essere un microcontroller, lo slave

può essere un microcontroller o un dispositivo che accetta la comunicazione I2C. Il concetto di comunicazione è lo stesso in entrambi i casi e si continuano a utilizzare le regole che sono state definite. È il PIC master che inizia la comunicazione e genera il segnale di clock, quindi sia il master che lo slave possono inviare e ricevere dati.

Passaggi nella comunicazione

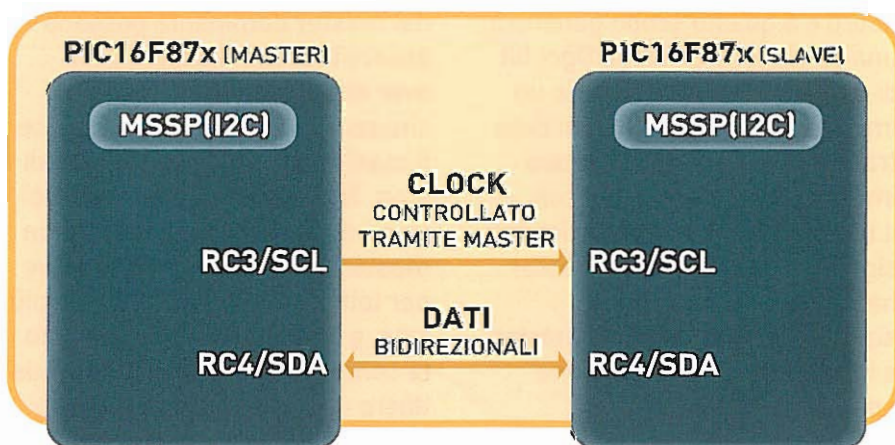
Nel modo slave, in un primo momento sia la linea dei dati (SDA) che quella del clock (SCL) devono essere configurate come ingressi, sarà il modulo MSSP stesso che modificherà la linea dei dati come uscita quando questo sarà necessario per trasmettere. Quando viene identificato il proprio indirizzo, che sarà il valore contenuto nel registro SSPADD,



Passaggi nella ricezione dei dati da parte di uno slave.

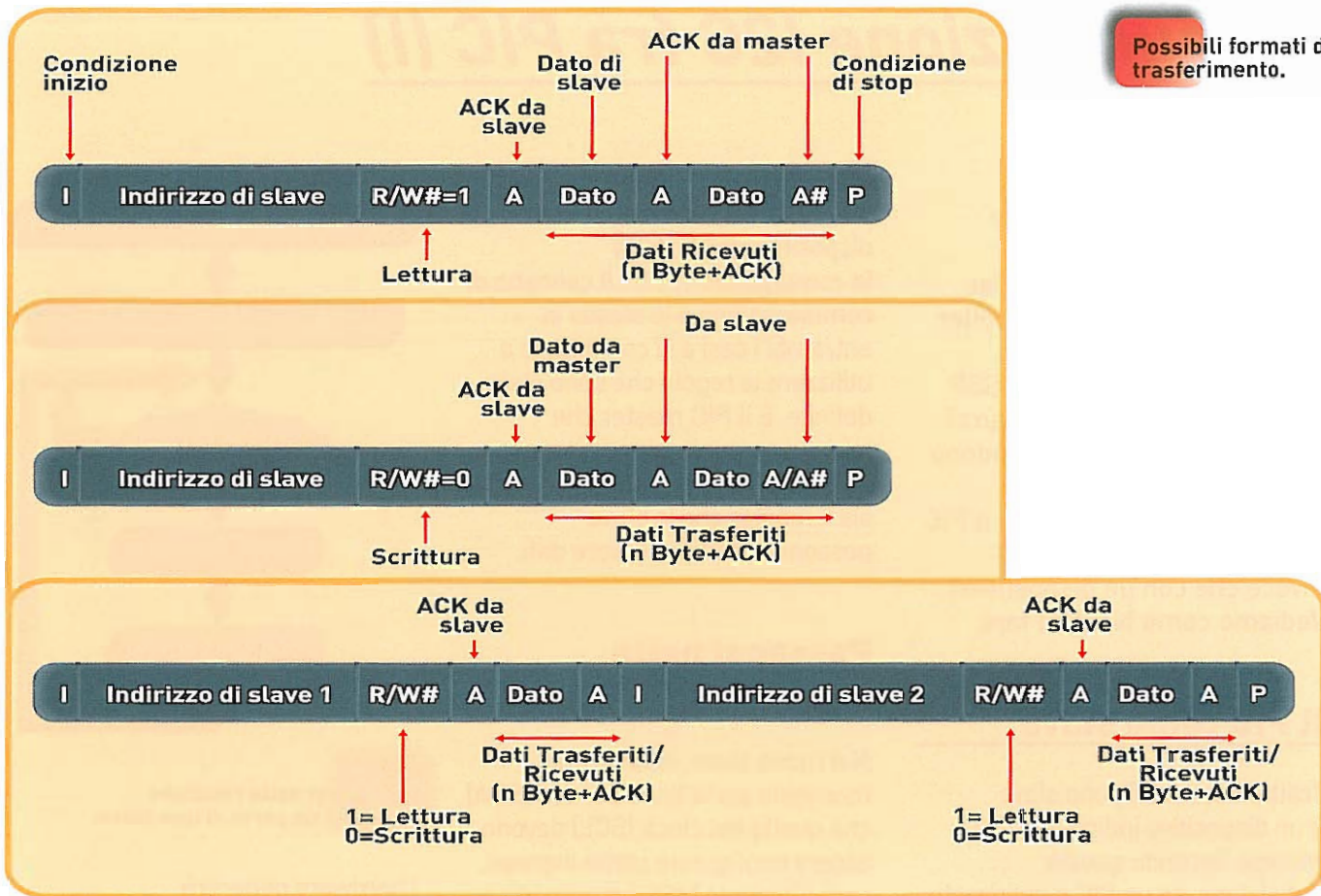
l'hardware genererà automaticamente l'impulso di riconoscimento (ACK). Esiste la possibilità di lavorare con indirizzi da 10 bit, in questo caso sarà necessario attendere di ricevere due byte di indirizzo.

Il microcontroller si può programmare per fare in modo che all'arrivo di ogni byte generi un interrupt, grazie al quale sapremo se c'è un nuovo dato da trattare. Un'altra opzione è di creare un ciclo in cui si verifica quando il bit corrispondente (SSPIF) avvisa di quanto successo, passando a valore 1. Il registro SSPBUF serve sia per ricevere il dato che per trasmetterlo, perciò quando al master arriva un ordine di lettura sarà questo registro che viene caricato con il valore richiesto, inoltre in una comunicazione con dei dispositivi, il significato di ogni



Lo slave può essere un dispositivo oppure un altro PIC.

Possibili formati di trasferimento.



byte trasmesso sarà definito e noto. Invece quando sono in comunicazione due PIC sarà il programmatore dell'applicazione che dovrà determinare quale informazione arriva con ogni byte e quindi quale significato abbia per uno slave un byte ricevuto o che byte vada trasmesso. I registri utilizzati sono gli stessi che si utilizzerebbero nel caso dei PIC master.

Sistemi multimaster

Un'altra opzione nella comunicazione fra dispositivi è che in un sistema ci siano più microcontroller dello stesso tipo, cioè che esista più di un master; si tratta dei sistemi multimaster.

La definizione I2C contempla questa possibilità con una tecnica che evita conflitti sul bus e perdite di informazione.

Un master può iniziare una comunicazione solo se il bus è libero e a questo scopo genererà una condizione di inizio. Ogni bit di dato va accompagnato da un impulso di clock. Se il bit di dato trasmesso, insieme con il suo impulso di clock coincide con il livello logico della linea dei dati, significherà che questo master ha il controllo della linea, sia perché è stato l'unico master a iniziare una comunicazione oppure perché il sistema di arbitraggio gli ha concesso l'utilizzo del bus. La comparazione per determinare l'arbitraggio può aver bisogno di

diversi bit, sino a quando esiste un fattore di coincidenza con gli altri master. Dopo l'arbitraggio i master "perdenti" si devono porre immediatamente in modo slave, dato che i dati inviati dal master dominante possono essere indirizzati a loro. Dopo aver assegnato il bus, questa situazione non cambia sino a che il master genera la condizione di stop. Tuttavia come mostrato nei primi due esempi della figura un master può indirizzare uno slave per lettura o scrittura di uno o più byte, e terminare a questo punto la comunicazione lasciando il bus libero oppure generare un'altra condizione di start per indirizzare un nuovo slave senza una condizione di stop precedente come nel terzo caso della figura.