

Comunicazione I2C con dispositivi (I)

Continueremo il discorso della comunicazione seriale, parlando del secondo modo fornito dal modulo MSSP dei PIC, che fornisce vantaggi in più rispetto al modo SPI già visto.

Differenza tra i modi SPI e I2C

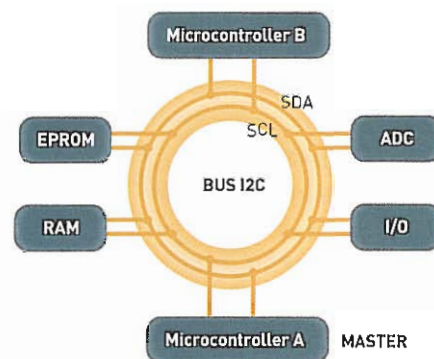
Il principale vantaggio del modo I2C (protocollo sviluppato da Philips per rispondere in un primo momento alle sue necessità di un'elevata interconnessione fra circuiti) rispetto al modo SPI è dato dall'enorme diffusione sul mercato ottenuta dal primo; questo fa sì che esistano moltissime periferiche capaci di lavorare con questo protocollo,

ad esempio memorie, controller, clock in tempo reale, convertitori, ecc. La comunicazione continua a essere di tipo seriale e, come già avveniva con il modo SPI, questo tipo di comunicazione permette a un PIC di comunicare con un dispositivo esterno o con un altro PIC.

In questo capitolo ci dedicheremo alla prima opzione.

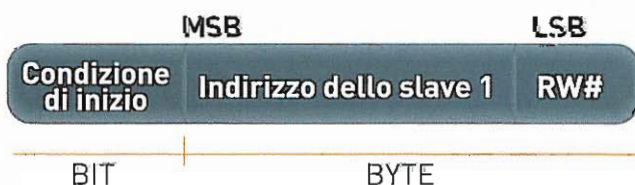
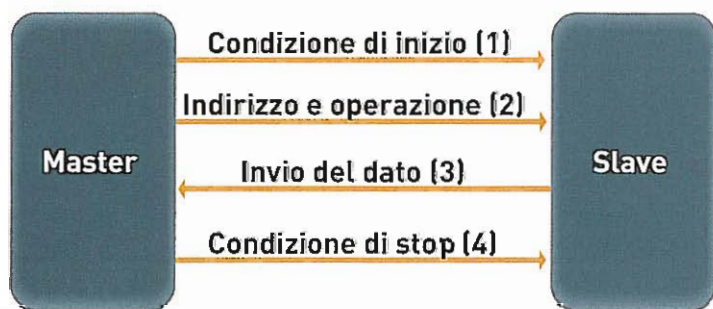
Concetto del I2C

Al posto delle tre linee, di cui aveva bisogno il modo SPI, il modo I2C lavora solamente con due, rappresentate come una struttura a forma di anello dove una delle linee è dedicata a supportare i dati (SDA) e l'altra porta gli impulsi di clock per la sincronizzazione (SCL). Questi impulsi sono generati dal



Schema dei collegamenti dei dispositivi in modo I2C.

master che dirige la comunicazione, mentre al bus si "collegano" i diversi slave. Nelle specifiche di ogni dispositivo sono stabiliti i vari parametri di comunicazione che bisogna seguire, inoltre queste specifiche sono strutturate in modo da prevedere una compatibilità che permetta a tutti di essere collegati allo stesso bus, rendendo nel contempo possibili espansioni e modifiche. In questo modo ogni dispositivo si identifica con un unico indirizzo, e sarà in grado di trasmettere e/o ricevere dati a seconda della sua natura, anche se sarà sempre gestito dalle richieste del master del sistema. Una novità rispetto al modo SPI è che I2C è un bus multi-master, in altre parole sullo stesso bus possono esistere più master contemporaneamente, che dovranno lavorare in modo da non generare collisioni fra dati. Questi "master" di solito sono microcontroller, però un microcontroller in questo sistema può funzionare anche come slave.



Ordine di trasmissione dei dati.



Fasi della comunicazione

Prendiamo in esame un primo caso in cui un PIC master comunica con uno slave. Esso potrà sia inviare che ricevere i dati, ma in entrambi i casi si dovrà prima indirizzare lo slave, poi inviare il dato o attendere di riceverlo e in ultimo terminare il trasferimento. Essendo un sistema multi-master, in un altro momento lo stesso PIC potrà funzionare come slave, inoltre dato che più master potrebbero iniziare la comunicazione contemporaneamente, è stato previsto un sistema di arbitraggio del bus. Tutti i trasferimenti di dati iniziano con ciò che viene chiamato "Condizione di Inizio" e terminano con la "Condizione di Stop". Il bus sarà occupato all'interno di queste due condizioni. Tramite la linea SDA il master invia un byte (8 bit) seguito da un bit di riconoscimento

(ACK), che è obbligatorio ed è generato dal ricevitore. Se il ricevitore non può mantenere la comunicazione, ad esempio perché è già occupato in un'altra comunicazione, genera un bit NACK, per fare in modo che il master generi la Condizione di Stop e annulli questa comunicazione. Il master genererà anche il bit ACK dopo la ricezione di ogni byte, ad eccezione dell'ultimo, dopo il quale manderà la Condizione di Stop. Fra byte e byte ci potrà essere un tempo di attesa. Dopo la Condizione di Inizio, si deve inviare l'indirizzo dello slave in questione, e comunicare se l'operazione dovrà essere di lettura o di scrittura, cosa che si fa con un solo byte. L'ordine è riportato nella figura allegata, insieme ai byte inviati dal master per un procedimento di richiesta di dati. Gli indirizzi possibili e il loro significato sono specificati all'interno del protocollo; quando arriva un indirizzo tramite il bus,

ogni slave lo compara con il proprio, per vedere se è chiamato in causa. Una parte di questo indirizzo è fissa, a seconda del tipo di dispositivo, e l'altra è programmabile, in modo che possano coesistere diversi dispositivi dello stesso tipo sullo stesso bus. Questo è anche regolato da Comitato I2C.

Alcuni dispositivi necessitano di più di un byte di indirizzo.

Registri utilizzati

Per il lavoro in modo I2C, è necessario, oltre al registro di controllo SSPCON e al registro di stato SSPSTAT, già visti nel modo SPI, un secondo registro di controllo (SSPCON2). SSPBUF sarà il registro dove collocare il dato da trasmettere, o dove si ricevono i dati a seconda dell'operazione, e SSPADD il registro di indirizzo, quando si svolge la funzione di slave.

REGISTRO SSPSTAT						
SMP	CKE	D/A#	PS	R/W#	UA	BF
7						0

Registri di controllo e stato utilizzati nel modo I2C.

REGISTRO SSPCON							
WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0
7							0

REGISTRO SSPCON 2							
GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
7							0

GCEN: Solo in Modo Slave
ACKSTAT=1. È stato ricevuto il bit ACK dallo Slave
ACKDT=0. Il Master ha trasmesso il bit ACK
ACKEN=1 inizia la sequenza di generazione della Condizione di Riconoscimento
RCEN=1 abilita il modo di ricezione del Master
PEN=1 genera la Condizione di Stop
RSEN=1 inizia la ripetizione della Condizione di Inizio
SEN=1 inizia la Condizione di Inizio

