

BUS I2C

Il Bus I2C (abbreviazione di Inter Integrated Circuit) è un sistema di comunicazione bifilare (2 fili più il riferimento di zero) tra più dispositivi.

Normalmente nel protocollo è previsto un dispositivo Master ed più dispositivi Slave. E' però possibile incontrare anche sistemi multi-master.

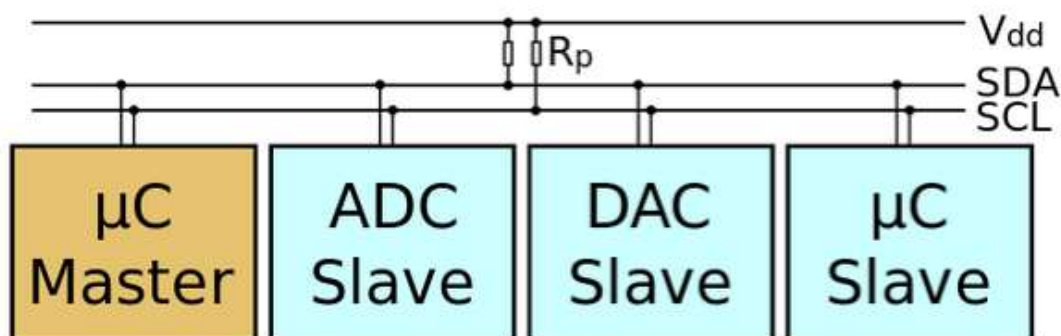
Il protocollo prevede l'utilizzo di 2 linee:

- SDA: linea in cui vengono trasmessi o ricevuti i dati
- SCL: linea di clock

Come già precedentemente accennato esiste una terza linea che corrisponde al riferimento di zero (normalmente chiamata GND).

I dispositivi collegati in I2C prevedono nelle linee SDA e SCL degli stadi open collector e quindi sono in grado di portare a 0 la linea dati o il clock ma non sono in grado di riportarlo al livello logico alto; per fare ciò è quindi necessario inserire delle resistenze di pull-up sulle 2 linee (tipicamente 4,7K Ohm) per garantire che queste possano assumere il valore logico alto.

Di seguito riportiamo un esempio di collegamento tra dispositivi attraverso il protocollo I2C:



Il dispositivo Master per poter leggere o scrivere da uno slave, dovrà prima di tutto inviare l'indirizzo dello slave da contattare, successivamente inviare l'indirizzo del registro da interrogare e, nel caso della lettura riceverà il dato, mentre nel caso della scrittura dovrà fornire anche il dato da scrivere.

La DM Board ICS non prevede al suo interno le 2 resistenze di pull-up e quindi se anche i dispositivi a cui viene connessa non la prevedono, devono essere aggiunte.

DM Board ICS può funzionare sia da master che da slave.

Utilizzo del protocollo I2C con DMDesign

DMDesign prevede tutte le istruzioni per il funzionamento della DMBoard ICS come Master o come Slave.

Nel caso si voglia utilizzare la DM Board ICS come master è necessario eseguire le seguenti operazioni:

- Settare l'I2C in Master mode
- Scrivere o leggere il dato

Entrambe le operazioni corrispondono ad un’istruzione in DMDesign. Per la lettura o la scrittura è necessario fornire l’indirizzo della periferica e del registro da leggere. Questi valori devono essere passati all’istruzione attraverso delle variabili numeriche.

Riportiamo di seguito un esempio di scrittura di un dato lato master:

The screenshot displays the DMDesign software interface, divided into two main sections: **I2C function** (left) and **User code** (right).

I2C function section:

- Set I2C bus in master mode**
- Set I2C bus in slave mode**
I2C slave address=In8_1
Rln_1=I2C state
- Write data to I2C (Master mode)**
Rln_1=peripheral address
Rln_2=register address
Rln_3=value to write
- Read data from I2C (Master mode)**
Rln_1=peripheral address
Rln_2=register address
Rln_3=readed value
- Read data from I2C register**
Rln_1=readed value
Rln_2=index of I2C register
- Write data to I2C register**
Rln_1=value to write
Rln_2=index of I2C register
- Store I2C register to flash**
MemOut_1=flash address
- Load I2C register from flash**
MemIn_1=flash address

User code section:

- VARIABLES**
 - Define numeric variable: IndirizzoPeriferica
 - Define numeric variable: IndirizzoRegistro
 - Define numeric variable: ValoreDa Scrivere
- OUTPUTS**
 - Set I2C bus in master mode
 - Store constant to register: IndirizzoPeriferica=18
 - Store constant to register: IndirizzoRegistro=0
 - Store constant to register: ValoreDa Scrivere=10
 - Write data to I2C (Master mode)
IndirizzoPeriferica=peripheral address
IndirizzoRegistro=register address
ValoreDa Scrivere=value to write
- JUMPS**

Per quel che riguarda la lettura la procedura è analoga ma in questo caso dovremo fornire un registro nel quale verrà riposto il dato letto:

The screenshot displays the DM Design software interface, divided into two main sections: 'I2C function' and 'User code'.

I2C function section:

- Set I2C bus in master mode**
- Set I2C bus in slave mode**
I2C slave address=In8_1
RIn_1=I2C state
- Write data to I2C (Master mode)**
RIn_1=peripheral address
RIn_2=register address
RIn_3=value to write
- Read data from I2C (Master mode)**
RIn_1=peripheral address
RIn_2=register address
RIn_3=readed value
- Read data from I2C register**
RIn_1=readed value
RIn_2=index of I2C register
- Write data to I2C register**
RIn_1=value to write
RIn_2=index of I2C register
- Store I2C register to flash**
MemOut_1=flash address

User code section:

VARIABLES

- Define numeric variable: IndirizzoPeriferica
- Define numeric variable: IndirizzoRegistro
- Define numeric variable: ValoreLetto

OUTPUTS

- Set I2C bus in master mode
- Store constant to register: IndirizzoPeriferica=18
- Store constant to register: IndirizzoRegistro=0
- Read data from I2C (Master mode)
IndirizzoPeriferica=peripheral address
IndirizzoRegistro=register address
ValoreLetto=readed value

JUMPS

Se vogliamo invece utilizzare la DM Board ICS come dispositivo Slave, dobbiamo prima di tutto capire come viene gestita questa modalità dalla scheda.

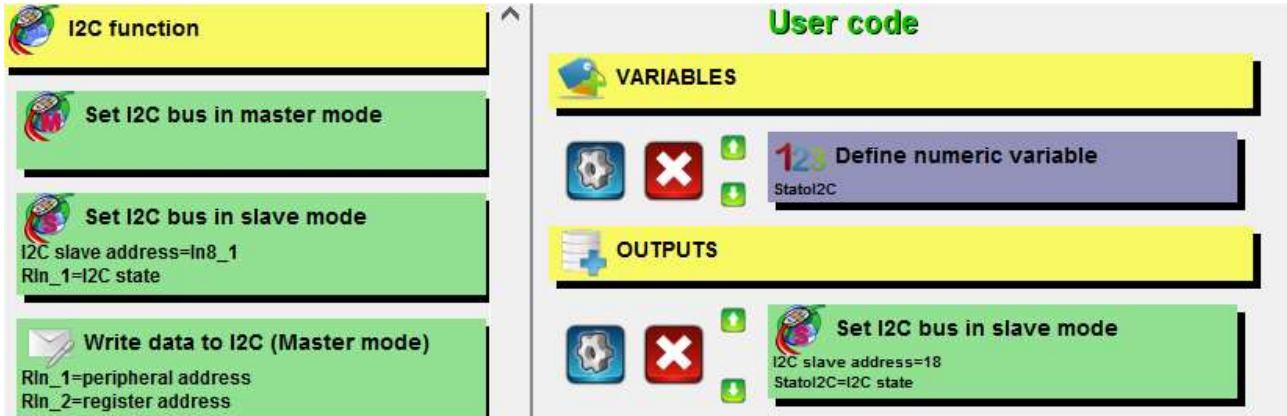
In fase di attivazione della modalità slave, verrà chiesto l'indirizzo della periferica ed una variabile in cui verrà salvato lo stato dell'I2C.

La DM Board ICS, una volta attivato l'I2C in modalità slave, attiva una periferica hardware che si occupa della ricezione o trasmissione automatica dei dati richiesti. I dati richiesti vengono presi da un registro formato da 255 celle a 8 bit dedicato a questa periferica. In DM Design, per poter interagire con questo particolare registro, esistono 4 diverse istruzioni:

- Istruzione per la lettura di una cella del registro I2C
- Istruzione per la scrittura di una cella del registro I2C
- Istruzione per il salvataggio in flash del registro I2C
- Istruzione per il caricamento dalla flash del registro I2C

Infine per sapere quando il Master ha letto o scritto un dato nel registro I2C è necessario controllare la variabile di stato dell'I2C definita all'attivazione della modalità slave. Questa variabile verrà messa a 1 nel caso in cui venga letto un dato dal Master, mentre verrà messa a 2 nel caso in cui il Master scriva un dato.

Quindi per il funzionamento della modalità slave, basta attivarla come in figura seguente:



L'indirizzo dello slave viene in questo caso passato come costante.

Potranno poi essere utilizzate le istruzioni "Read data from I2C register" o "Write data to I2C register" per leggere o modificare il registro I2C della scheda.