



Steve Blackbird (TardoFreak)

PIERIN PIC18: SCHEDA OPEN SOURCE PER IMPARARE E SPERIMENTARE CON I MICROCONTROLLORI

1 June 2013

Al giorno d' oggi ci sono una miriade di microcontrollori e relativi sistemi di sviluppo. Si va dal piccolo micro in contenitore a 6 pin fino al bestione a 32 bit che fa cose strabilianti. Tale grande offerta se da un lato è un bene, dall' altro può creare confusione in coloro che vorrebbero avvicinarsi a questo mondo e che poco ne sanno. In effetti è difficile per un neofita scegliere il primo sistema di sviluppo. In questo articolo ne descrivo uno che è stato pensato per chi vuole iniziare con il piede giusto, cioè non vuole sprecare il proprio tempo a studiare cose incomprensibili o inutili.

Caratteristiche

Un buon sistema di sviluppo didattico o per la prototipazione veloce è comunque un compromesso fra tanti fattori. Diciamo che dovrebbe avere almeno queste caratteristiche.

- Semplice da utilizzare.
- Abbastanza potente per non trovarsi in breve tempo ad avere bisogno del "fratello maggiore" o comunque dover cambiare micro.
- Che si possa usare per un gran numero di applicazioni, da quelle più semplici a quelle più complesse.
- Sistema di sviluppo gratuito.
- Poter essere programmato in assembly ma anche e soprattutto in C.

Ora, in commercio ne esistono tantissimi. Alcuni però utilizzano micro obsoleti, oppure poco potenti, o utilizzano linguaggi di programmazione non standard e quindi inutili per una conoscenza generale dei microcontrollori, altri costano parecchio o i sistemi di sviluppo nelle versioni lite sono troppo limitati, altri sono macchinosi ecc.. La scelta del microcontrollore diventa quindi una cosa cruciale.

Il **PIERIN**

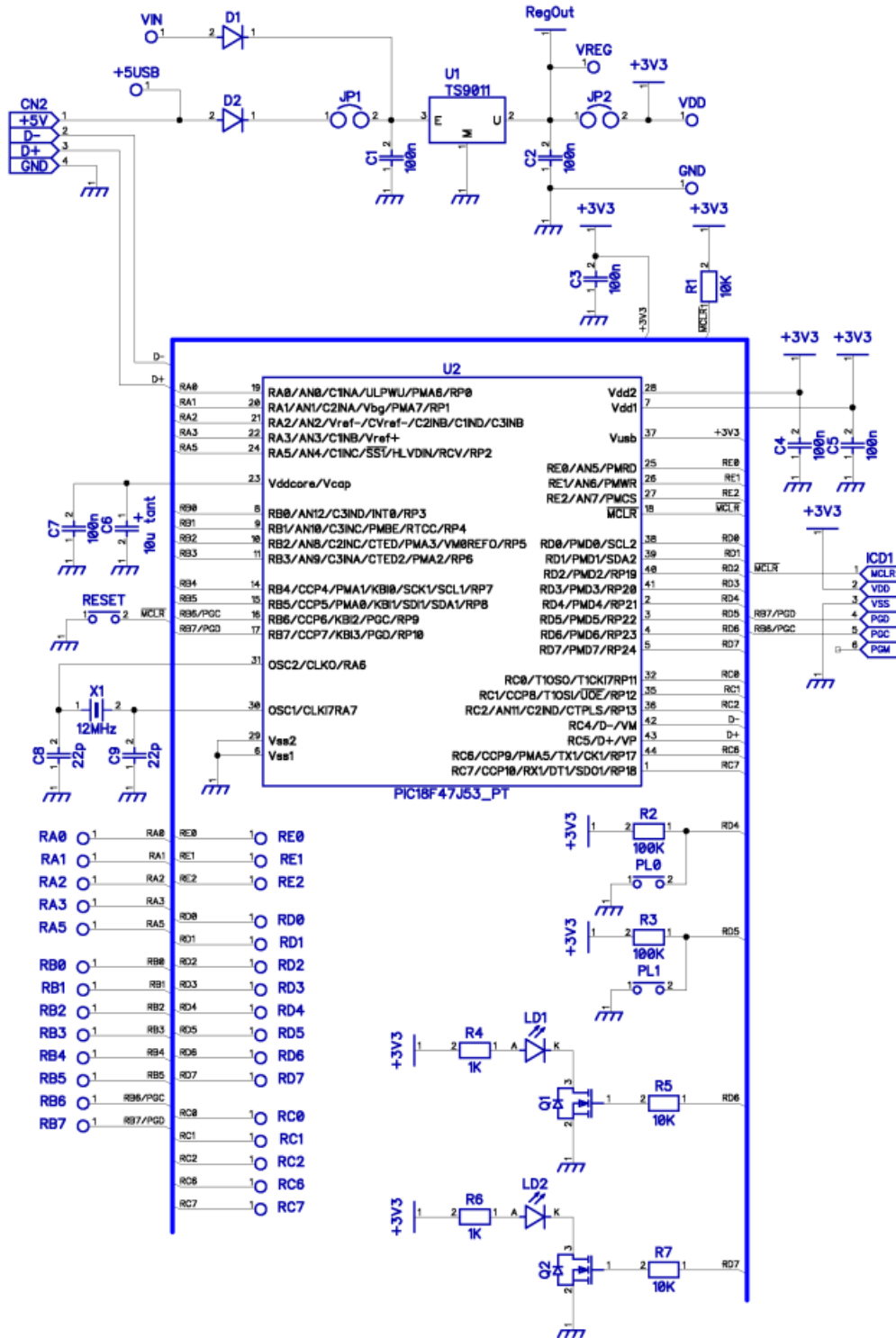
E' uno schedino che definirei "piccolo e cattivo" infatti monta un microcontrollore che, a mio avviso, è un gioiellino: [il PIC18F47J53](#) che ha queste caratteristiche:

- E' un microcontrollore ad 8 bit quindi semplice da utilizzare e programmare anche in asseby (ottimo per lo studio)
- Ha ben 128KB di FLASH che permette di fare veramente tante cose. Insomma ha 4 volte la FLASH di un PIC18F4550 o di un ATmega328. Permettetemi di dire che con 128KB di FLASH si riescono a fare cose strabilianti. Nel momento in cui si arriva a riempire la memoria del micro significa che non si avranno problemi ad utilizzare qualsiasi altro microcontrollore.
- Ha una RAM di tutto rispetto: 4KB
- ADC con risoluzione 12 bit.
- Timer ad 8 e 16 bit in abbondanza, PWM, cattura e quant' altro.
- SPI, I2C, USART e chi più ne ha più ne metta.
- Mappatura degli I/O, caratteristica particolarmente gustosa.
- interfaccia USB integrata. Avere l' USB permette di scrivere applicazioni veramente gustose. Si va dalla semplice tastira/mouse a lettori di SD-CARD alle periferiche audio.
- Il sistema di sviluppo è fornito dalla Microchip ed il compilatore C nella versione lite, anche se limitata nell' ottimizzazione, funziona benissimo. Tutti i progetti applicativi della Microchip funzionano alla perfezione anche con la versione lite.
- MPLAB-X è un sistema multiplatforma quindi i linux-ari e i mac-chettari sono serviti.
- Se proprio ci si vuole sbizzarrire ci sono altri sistemi di sviluppo gratuiti come l' SDCC.
- COSA IMPORTANTISSIMA: la Microchip fornisce gratuitamente un' enorme libreria di codice e di applicazioni dalla quale si può imparare tantissimo!
- I sistemi di sviluppo non sono macchinosi da utilizzare.
- La scheda ha un bootloader USB per programmato che non necessita di driver.
- Il software del bootloader lato PC è multiplatforma.
- Lo scheda può funzionare stand-alone semplicemente collegandola alla USB oppure in un circuito. Non ci sono pin del microcontrollore impegnati per qualcosa di specifico se non i due pin della USB.
- Può veramente sostituire il micro nella prototipazione diventando così uno strumento per lo sviluppo dei propri prodotti, prodotti seri.

Direi che si può considerare un buon compromesso ed un buon punto di partenza per chi vuole imparare in modo serio ad usare questi bellissimi oggetti.

Lo schema elettrico

Questo è lo schema elettrico del [PIERIN PIC18](#).



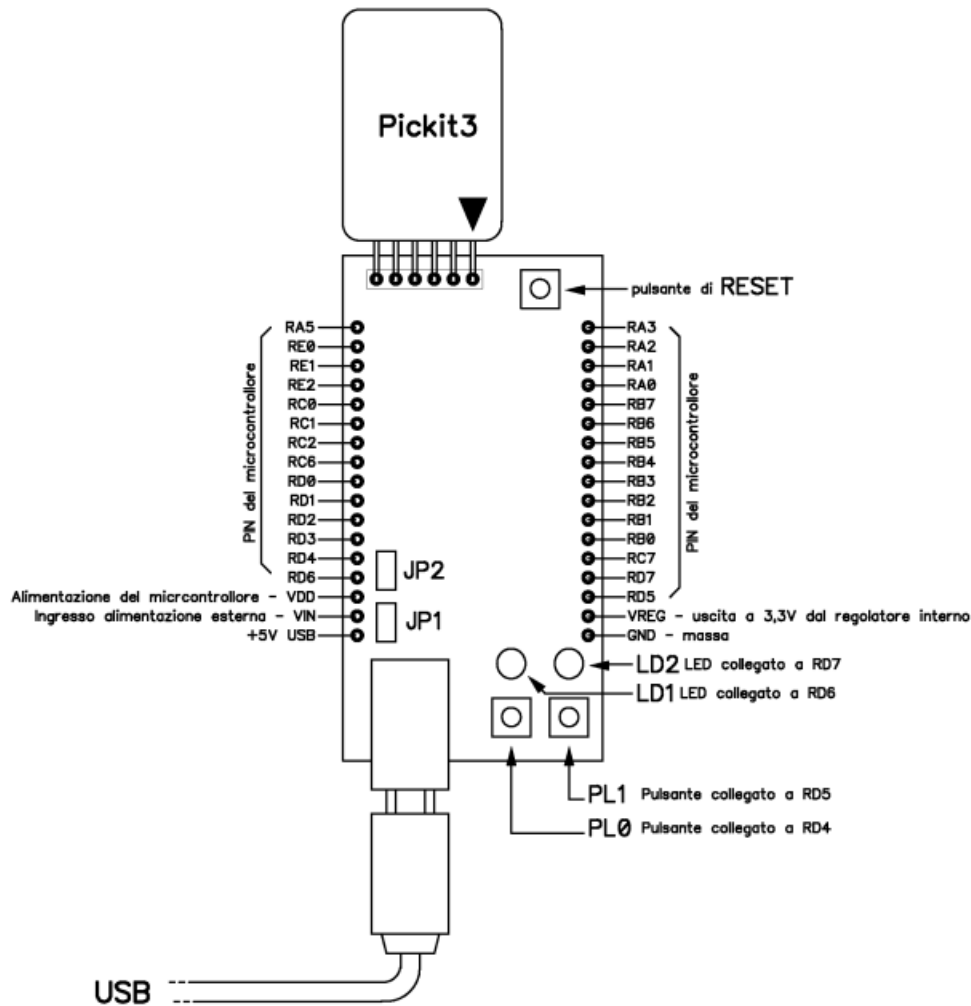
pierin_pic18_sch.png

In alto a sinistra troviamo il connettore USB CN2. Dal pin 1 di CN2 si prelevano i 5V forniti dall' USB che vengono portati all' esterno tramite il terminale +5USB.

Attraverso D2 e JP1 i 5V raggiungono il regolatore di tensione U1. Il regolatore può essere anche alimentato da una sorgente esterna il cui punto d' ingresso è il terminale VIN. La tensione regolata a 3,3V è disponibile sul terminale VREG (in alto a sinistra) e, attraverso JP2 va ad alimentare tutto il circuito del microcontrollore. Con JP2 disinserito il circuito può essere alimentato tramite il terminale VDD. I condensatori C3, C4 e C5 servono per il bypass della tensione di alimentazione mentre i condensatori C6 e C7 (sulla sinistra) servono al regolatore interno che genera la tensione di funzionamento dell' interfaccia USB. Sempre sulla sinistra troviamo il pulsante di RESET, il quarzo a 12 MHz X1 ed i due condensatori C8 e C9 per l' oscillatore. Sulla destra troviamo il connettore per il Pickit3. Procedendo verso il basso incontriamo i due pulsanti PL0 e PL1 muniti di resistenza di pull-up e collegati rispettivamente ai pin RD4 e RD5 del microcontrollore. In basso a destra ci sono i due LED. Questi non vengono pilotati direttamente dai pin del microcontrollore ma attraverso i due mosfet Q1 e Q2. Questo sistema permette di non aggiungere un carico direttamente sulle uscite del microcontrollore.

Come collegarlo

Qui di sotto riporto lo schema dei collegamenti.



pierin_conn.png

Partendo dall' alto troviamo il posto per il connettore a 6 pin per il collegamento con l' emulatore/debugger Pickit3 che serve per la programmazione del microcontrollore ed il debugging del programma. Tuttavia, tramite il bootloader, è possibile fare a meno del programmatore perché il microcontrollore può essere programmato mediante il bootloader e collegato al PC tramite la porta USB (in basso nel disegno). Particolare importanza hanno i jumper JP1 e JP2 che servono per configurare l' alimentazione. Qui di seguito vediamo i vari casi anche se le funzioni dei jumper sono facilmente comprensibili guardando lo schema elettrico.

JP1=ON JP2=ON

Funzionamento in modalità stand-alone. L' alimentazione arriva attraverso la USB e sono a disposizione per l' eventuale circuito su cui è montato le tensioni di 5V provenienti direttamente dalla USB sul terminale +5USB e quella da 3,3V regolata sui terminali VDD e VREG.

JP1=OFF JP2=ON

La scheda ha bisogno di un' alimentazione esterna collegata al terminale VIN per funzionare poichè non riceve più la tensione di alimentazione dalla USB. Il regolatore interno però funziona, alimenta il microcontrollore e può anche alimentare un circuito esterno. La tensione da 3,3V regolata è disponibile sui terminali VDD e VREG.

JP1=ON JP2=OFF

Il regolatore di tensione è alimentato dalla USB ma non fornisce tensione al microcontrollore che per funzionare ha bisogno di una alimentazione esterna da collegare al terminale VDD. La tensione da 3,3V regolata è disponibile sui terminali VREG.

JP1=OFF JP2=OFF

Il regolatore non riceve la tensione dalla USB e non fornisce tensione al microcontrollore. Il microcontrollore per funzionare ha bisogno di una alimentazione esterna da collegare al terminale VDD.

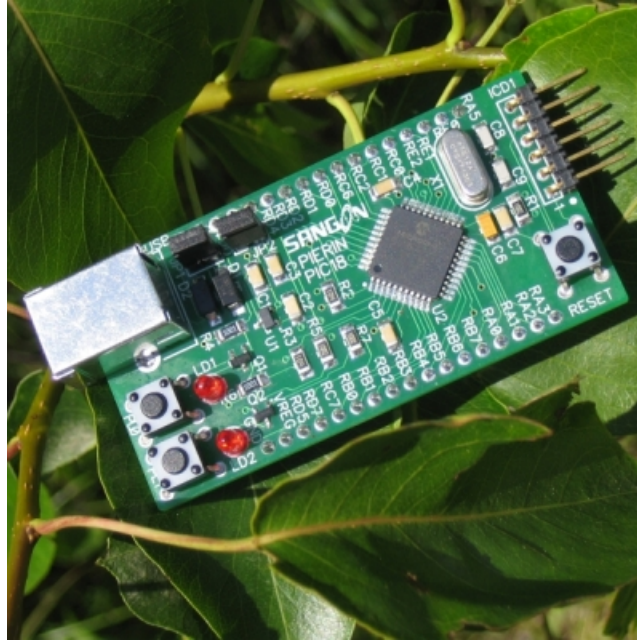
Conclusioni

Bene, ho spiegato la filosofia che sta dietro a questo tipo di soluzione didattica e di prototipazione veloce ma una cosa ci tengo a sottolineare: **non è per forza necessario avere fisicamente una di queste schede per sperimentare ed imparare i microcontrollori. Chi è pratico di elettronica la può realizzare da se**, e se non possiede un PicKit3 può cercare un amico o conoscente che gli programmi dentro il bootloader.

Allegato a questo articolo c'è il file **pierinpic18.rar** che contiene i pdf dello schema elettrico e dello schema di collegamento **ed il file .hex del bootloader**. Quindi lo si prende, lo si programma nel micro e da quel momento in poi non si avrà più bisogno del PicKit3!

Auguro a tutti una **BUONA SPERIMENTAZIONE!**.

Il file contenente la documentazione ed il bootloader lo si può scaricare da [questo link](#)



Il Pierin PIC18

Estratto da "<http://www.electroyou.it/mediawiki/index.php?title=UsersPages:Tardofreak:pierin-pic18-per-imparare-e-sperimentare-con-i-microcontrollori>"