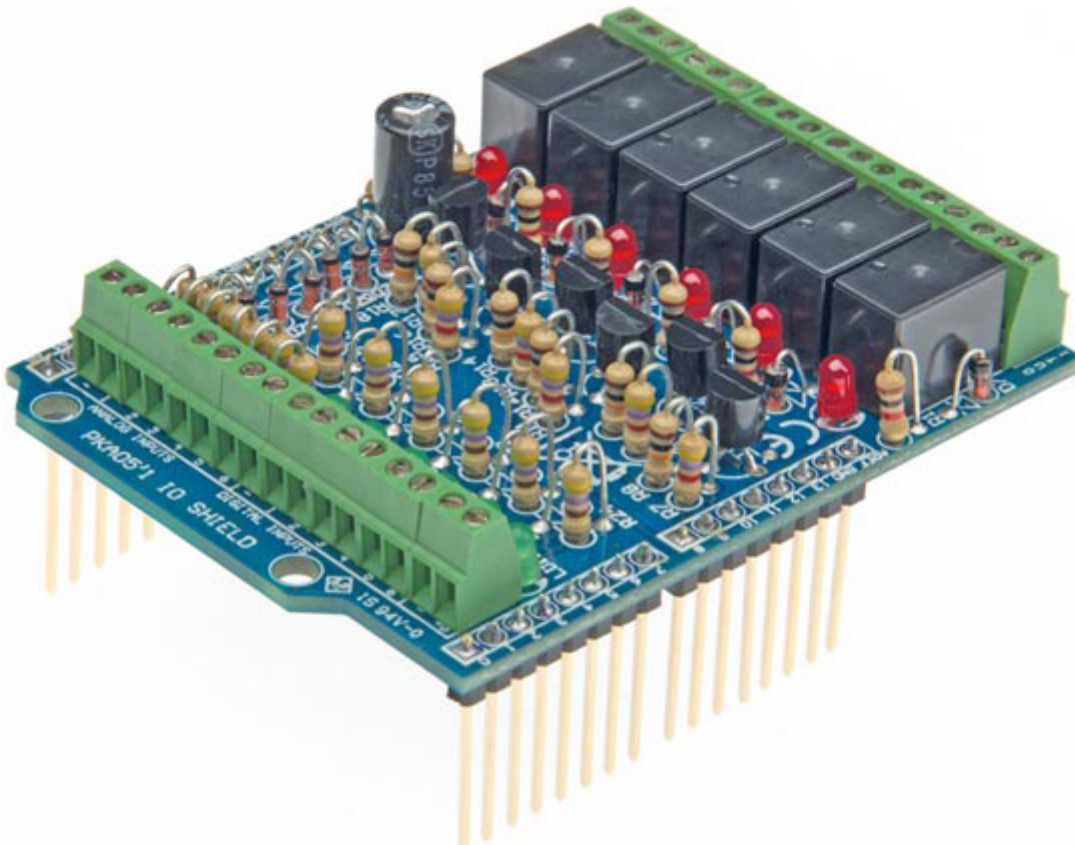


Shield IN/OUT per Arduino - 6 relè, 6 ingressi analogici e digitali

Prezzo: 24.18 €

Tasse: 5.32 €

Prezzo totale (con tasse): 29.50 €



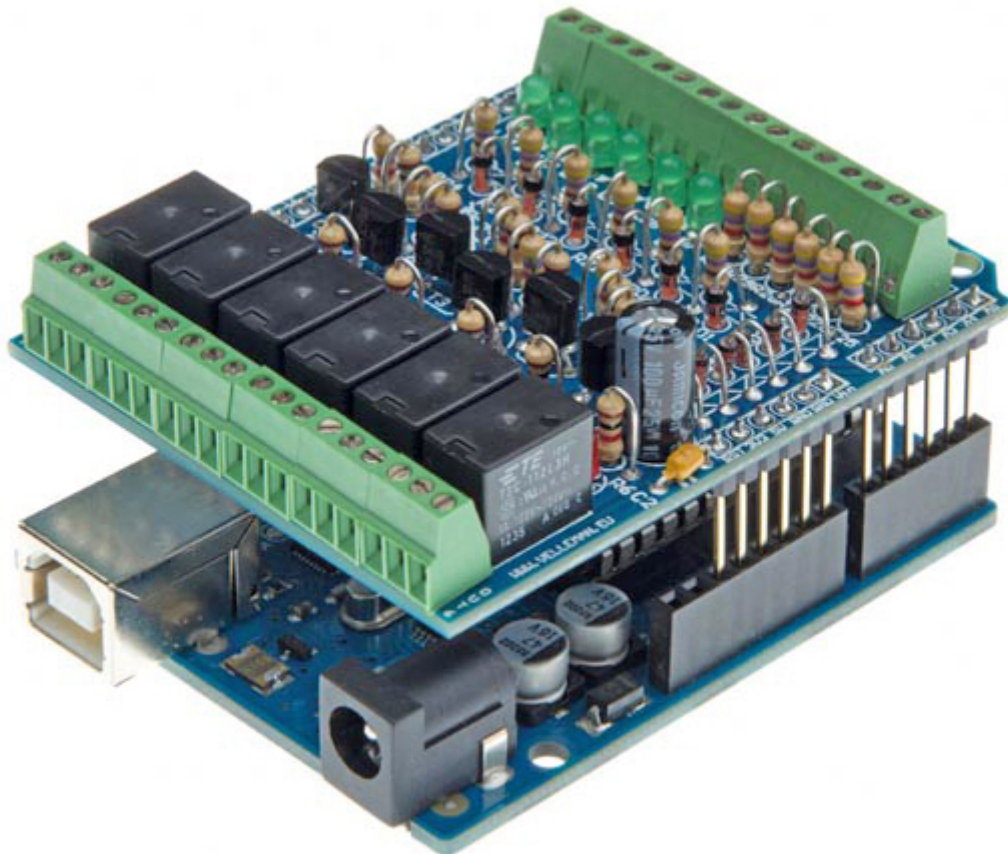
WHAODA

EXCITING ELECTRONICS

Lo Shield In/Out per Arduino è un modulo che consente ad Arduino di controllare 6 dispositivi utilizzando relè e di acquisire lo stato di 6 ingressi analogici e 6 digitali. Ogni ingresso digitale e uscita a relè è dotata di un LED che indica lo stato corrente. Le linee di input/output sono collegate ai pin corrispondenti di Arduino utilizzando un connettore a passo 2,54 mm. La scheda viene alimentata direttamente dal modulo Arduino, che fornisce i 5 volt necessari tramite i contatti 5V e GND. Tuttavia, poiché i mini-relè dello Shield funzionano a 12 volt, è necessario alimentare il modulo Arduino con un alimentatore esterno in grado di fornire questa tensione. Questo Shield può essere utilizzato in molte applicazioni e offre molteplici possibilità di utilizzo.

Lo shield alloggiato sulla scheda Arduino

[caption id="attachment_96957" align="aligncenter" width="600"]



Il nostro shield montato sulla scheda Arduino (*)[/caption]

* Attenzione! La scheda Arduino va acquistata separatamente * Lo shield viene fornito da montare

Usare lo shield I/O

La scheda può essere utilizzata per collegare Arduino ad utilizzatori che devono essere galvanicamente separati da essa (ad esempio piccoli motori elettrici) e leggere segnali di sensori o contatti che possono presentare disturbi impulsivi o picchi di tensione tali da danneggiare il microcontrollore ATmega senza un'adeguata protezione; i relé possono pilotare servo relé di maggiore potenza per gestire carichi che assorbono correnti molto intense o lavorano ad alta tensione. La gestione dei relé e l'acquisizione dei segnali digitali e delle tensioni degli input analogici possono essere effettuate utilizzando semplici routine software o firmware generici già pronti; è anche possibile procedere da Personal Computer, infatti l'IDE di Arduino dispone di un emulatore di terminale (una sorta di Hyper Terminal di Windows) che permette l'invio e la ricezione di dati seriali alla scheda e quindi la gestione diretta dal PC dei relé e degli input. Inviando tramite seriale (impostata per 9.600, 8, N, 1, ossia velocità di 9.600 bps, 8 bit, nessuna parità e un bit di stop) il comando **O** seguito dal numero del canale, si attiva il relé corrispondente; ad esempio, con **O2** si attiva il relé 2. Il funzionamento previsto per le uscite è di tipo bistabile (toggle) quindi ogni comando inverte lo stato del relativo relé. Col comando **Ix** (dove *x* è il numero del canale) si richiede lo stato di un ingresso digitale, mentre con **Ax** si interroga un ingresso analogico. Ad ogni comando, il modulo Arduino risponde all'emulatore di terminale con un messaggio che segnala la condizione dell'uscita dovuta al comando stesso, ovvero lo stato dell'ingresso oggetto del comando. Ad esempio, impartendo il comando **O2** si può ottenere "Out 2 = 1". Analogamente, per gli ingressi possiamo avere messaggi del tipo "In 3 = 0" se trattasi di input digitale o Analog 1=126 se si parla di input analogico; in quest'ultimo caso il valore indicato dopo il simbolo di uguale è il corrispondente dell'A/D converter. È anche possibile gestire gli I/O caricando nel microcontrollore di Arduino il firmware FIRMATA, scaricabile dalle pagine web <http://arduino.cc/en/Reference/Firmata> oppure http://firmata.org/wiki/Main_Page. Con esso potete gestire gli ingressi e le uscite direttamente da un'interfaccia grafica, grazie a Processing; quest'ultimo è un linguaggio di programmazione Java Based e open-source di cui abbiamo parlato nella quinta puntata del corso su Arduino, pubblicata nel fascicolo numero 151 (novembre 2010). Processing è un linguaggio che permette di realizzare moltissime applicazioni (sketch con Arduino) e in questo caso lo possiamo usare per elaborare lo sketch *firmata*, che si trova direttamente tra gli esempi dell'IDE di Arduino e che ci servirà a gestire input e relé della nostra scheda. Questo sketch va modificato leggermente, semplicemente per "dire" ad Arduino quali pin sono collegati alle uscite e quali agli ingressi; poi tutto funzionerà senza difficoltà.

Il linguaggio processing

Si tratta di un linguaggio di programmazione basato su [Java](#), che consente di sviluppare diverse applicazioni come giochi, animazioni e contenuti interattivi. Basandosi su Java, ne eredita completamente la sintassi, i comandi e il paradigma di programmazione orientata agli oggetti; inoltre mette a disposizione numerose funzioni ad alto livello per gestire facilmente l'aspetto grafico e multimediale. È distribuito sotto licenza [Open Source](#) ed è supportato dai sistemi operativi [GNU / Linux](#), [Mac OS X](#) e Windows. Il pacchetto, scaricabile gratuitamente dal sito ufficiale <http://processing.org>, mette a disposizione un ambiente di sviluppo integrato (IDE), e le varie creazioni (sketch) vengono organizzate in uno sketchbook e possono essere esportate come [Java applet](#). Le funzionalità di processing possono essere espanse tramite delle librerie aggiuntive: ad esempio, è possibile utilizzare in Processing protocolli di comunicazione Seriale, TCP/IP, UDP, RDP, OSC, riuscendo in questo modo a gestire una moltitudine di applicazione hardware esterne. Lo stesso ambiente di sviluppo Software di Arduino è derivato da Processing, con il quale condivide sia la struttura che il linguaggio di programmazione.

Caratteristiche

- 6 uscite relè: max 0.5A, 30V
- 6 ingressi analogici
- 6 ingressi digitali
- Design stackable: lo shield può essere impilato con altri shield
- Ampia comunità di utenti
- Richiede 1 Arduino UNO™ (non incluso)

Specifiche tecniche

- Carico massimo di 1A per uscita
- Tensione massima di 30V per uscita
- Utilizza i pin 8-13 come uscite, i pin A0-A5 come ingressi analogici e i pin 2-7 come ingressi digitali
- Dimensioni: 68 x 53mm

Download

- [Esempio sketch Arduino](#)