

# Interfacciare Arduino

Il termine **interfaccia** ha diversi significati; in generale sta a identificare qualcosa che si interpone fra due sistemi, ovvero agisce da tramite, allo scopo di permettere una forma di interazione.

Il termine ricorre in vari contesti e, dall'uso inizialmente limitato al campo tecnologico / informatico, è poi passato al linguaggio comune, sempre indicando un elemento che si presta a favorire i contatti fra due diverse realtà.

Nel caso descritto in questa pagina faremo invece riferimento a qualcosa di specifico, da utilizzarsi in abbinamento alla ormai notissima piattaforma **Arduino**. Chi usa tale sistema, sa bene che, in conseguenza del software utilizzato, si ottengono sui pin di uscita determinati stati logici o "livelli", che possono essere, come noto, livelli di tipo **L** (cioè **Low**) oppure di tipo **H** (cioè **High**).

Con questi livelli logici è senz'altro possibile comandare dispositivi esterni, come relè, motori, sistemi di illuminazione e quant'altro; occorre tuttavia considerare che il collegamento fra i pin di Arduino e i dispositivi esterni non può essere diretto, tranne pochi casi, poichè la corrente che i pin possono erogare è molto limitata.

Vedremo allora come realizzare questo collegamento dal punto di vista elettrico, in modo da far funzionare Arduino come un direttore di orchestra in grado di gestire qualsiasi processo.

## I pin di Arduino

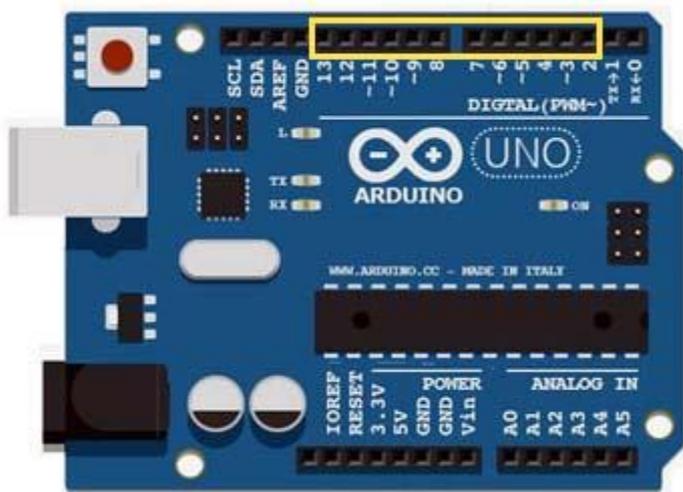


figura 1

Come ben sa chi si diverte a programmare la scheda Arduino, fra i vari pin di IN/OUT ve ne sono dodici che funzionano come pin digitali; ciò significa che ciascuno di essi può assumere un livello L, cioè una tensione pari a zero, oppure un livello H, cioè una tensione di circa 5 V.

Tali pin, che in figura 1 appaiono all'interno di un riquadro giallo, sono proprio quelli che permettono di controllare dispositivi esterni del tipo acceso/spento. Poichè la piattaforma Arduino si basa su di un microcontrollore ATmega, i pin che possiamo utilizzare possiedono le stesse caratteristiche di tale integrato; ad esempio, la massima corrente che ciascun pin può erogare è di 40 mA.

Una corrente di tale valore può sicuramente accendere un led, ma decisamente non può far funzionare dei carichi che presentano un maggior assorbimento: ecco allora che diventa necessario ricorrere ad un'interfaccia, ovvero ad un semplice circuito che,

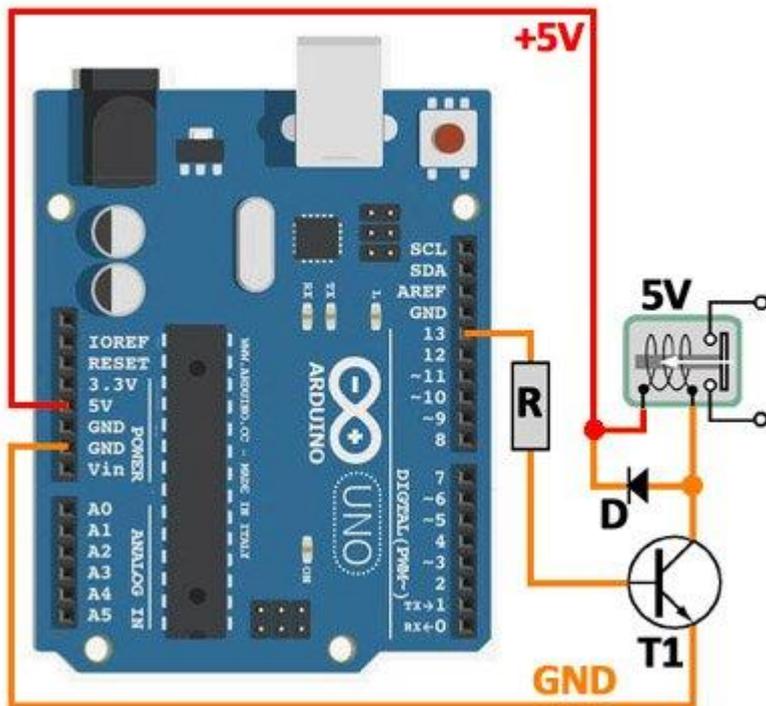


figura 2

comandato dalla debole corrente disponibile sui pin della scheda, si faccia carico di erogare ad un carico esterno tutta la corrente che questo richiede.

## Semplici circuiti di interfaccia

Vedremo adesso alcuni esempi pratici che hanno lo scopo di illustrare le situazioni più comuni.

**Primo caso:** immaginiamo di voler comandare un relè, in modo che si attivi (cioè che la sua bobina venga alimentata) quando sul pin di Arduino è presente un livello **H**; nell'esempio si è raffigurato il pin 13, ma potrebbe essere qualsiasi altro pin digitale. Se si utilizza un relè la cui bobina funziona a 5 V, lo schema può essere quello di figura 2.

Il relè risulta inserito nel circuito di collettore di T1, un transistor npn [BC547](#) o equivalenti, e viene da questo pilotato; la linea di massa (linea in colore arancio) viene derivata da uno dei pin "GROUND" della scheda, così come l'alimentazione a 5 V (linea in colore rosso). Tramite la resistenza R, del valore di circa 4,7 kohm, la base di T1 è collegata al pin della scheda;

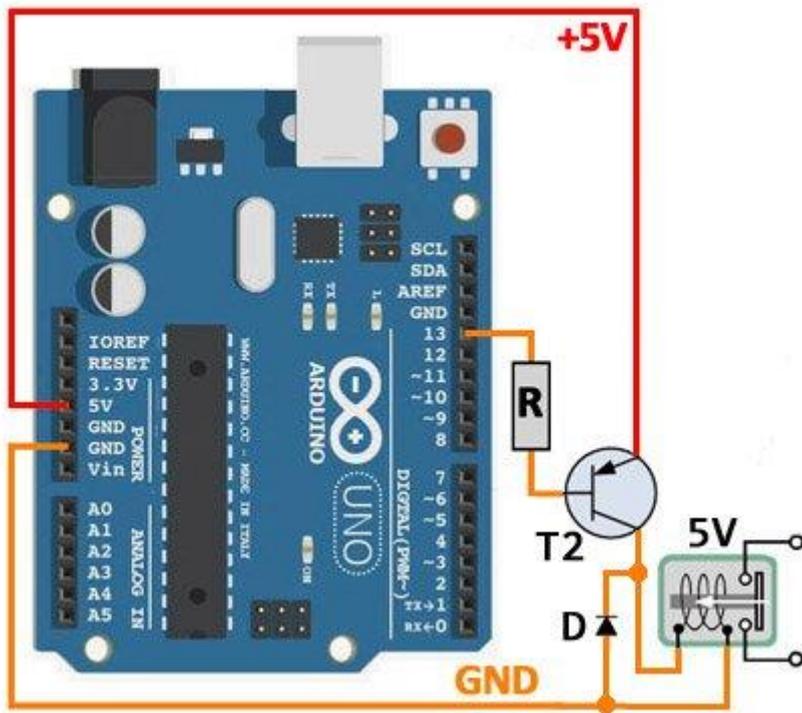


figura 3

quando il pin va a livello H, il transistor passa in conduzione.

Come sempre, il diodo D (1N4001 o simili) ha il compito di sopprimere le extratensioni causate dalla bobina del relè. Con un transistor come il BC547 è possibile comandare carichi che richiedono correnti fino ad un massimo di 100 mA; tale corrente è sicuramente sufficiente per pilotare un relè a 5 V.

**Secondo caso:** vogliamo che il relè scatti quando sul pin di uscita, che anche in questo caso è il numero 13, è presente un livello L. Per ottenere questo tipo di funzionamento occorre utilizzare un transistor pnp, come il [BC557](#) o equivalenti. Lo schema diventa quello della figura 3. Il relè si trova sul circuito di collettore del transistor T2: quando il pin 13 passa a livello L, attraverso la resistenza R porta in conduzione il transistor e quindi fa scattare il relè. Anche in questo caso, naturalmente, il relè deve avere una bobina a 5 V, bypassata dal solito diodo D. La resistenza R può avere lo stesso valore del caso precedente. Il transistor BC557, come già detto per il BC547,

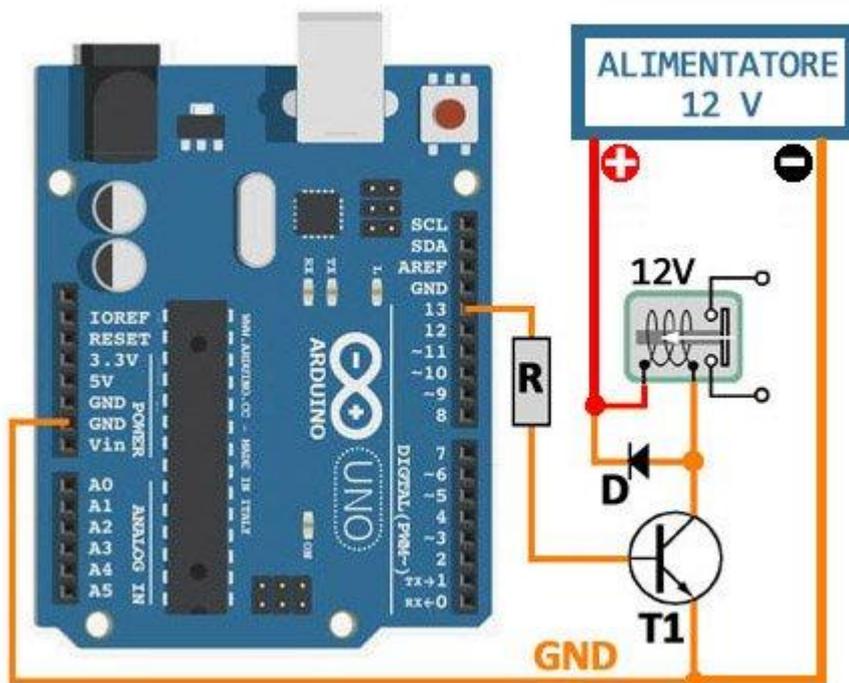


figura 4

può alimentare carichi fino ad un massimo di 100 mA.

**Terzo caso:** come nel primo caso, vogliamo comandare un relè perchè scatti quando sul pin 13 è presente un livello H; questa volta, però, useremo un relè con bobina funzionante a 12 V.

Poichè la scheda Arduino dispone solo di 5 V, occorre che la tensione di 12 V, necessaria al funzionamento del relè, sia fornita da un circuito esterno; nell'esempio di figura 4 si è ipotizzato che tale tensione venga fornita da un apposito alimentatore.

Osserviamo che la linea a zero volt della scheda Arduino, derivata come sempre dal pin GND, risulta collegata al negativo dell'alimentatore a 12 V, così da costituire una massa unica.

Il circuito di collettore di T1, che pilota il relè, farà capo, in questo caso, al polo positivo dell'alimentatore, così da ricevere la tensione di 12V.

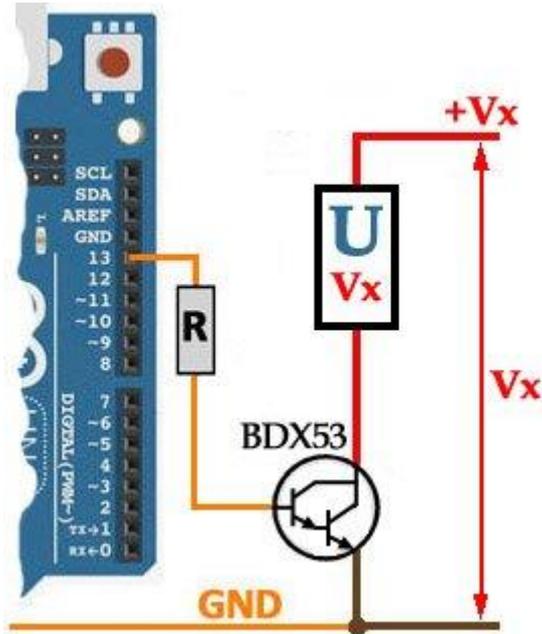


figura 5

**Comandare un carico di potenza:** Col circuito di figura 4 è naturalmente possibile pilotare non solo un relè, ma qualsiasi altro carico funzionante a tensione diversa da quella della scheda Arduino. Nel caso di carichi (ovvero utilizzatori) che richiedano forti correnti, dell'ordine di vari ampere, occorre sostituire il transistor BC547 con un transistor di potenza, meglio se di tipo darlington.

Il darlington [BDX53](#) di figura 5, per esempio, può controllare sul collettore correnti oltre 5 A, per una potenza totale di circa 60W.

Come nei casi precedenti, la linea GND viene derivata da uno dei pin GND della scheda; il transistor [BDX53](#) viene comandato dal pin 13 tramite una resistenza R che può avere un valore di varie decine di kohm, mentre nel circuito di collettore sarà inserito l'utilizzatore **U**. La tensione  $V_x$  richiesta dall'utilizzatore **U** sarà fornita da un'apposita sorgente esterna.

### Comandare la rotazione di un motore

Con la scheda Arduino è possibile, tra l'altro,

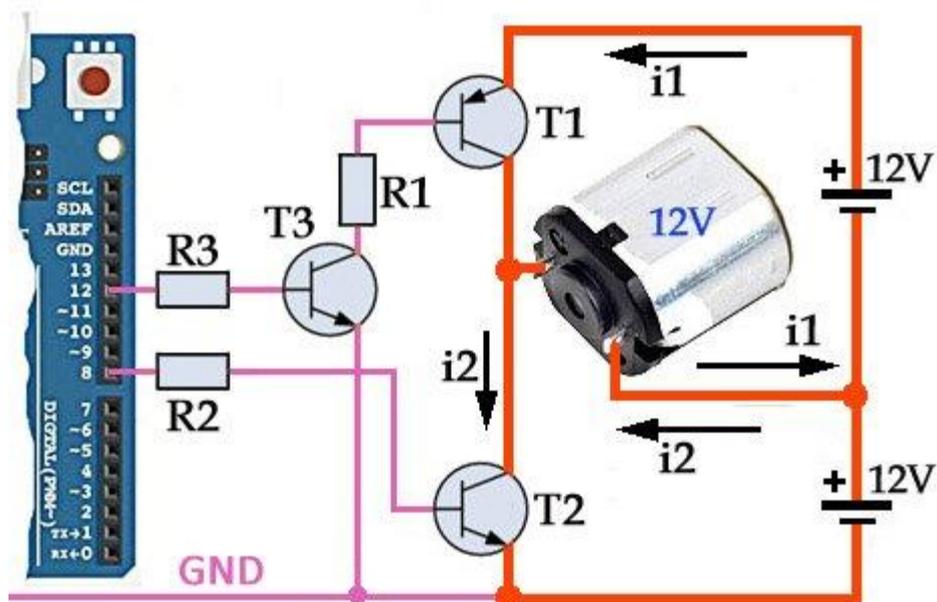


figura 6

comandare a piacere la rotazione di un motorino in corrente continua, semplicemente invertendo i livelli logici su alcuni pin.

Un esempio è riportato nella figura 6, dove si suppone di voler controllare un motorino funzionante a 12 Vcc. Per realizzare tale schema occorre prevedere una tensione di alimentazione con zero centrale: se, come nell'esempio, il motorino necessita di 12V, sarà necessaria un'alimentazione 12-0-12. Per "zero centrale" s'intende il punto di unione fra la tensione positiva e quella negativa.

La rotazione viene comandata, in questo caso, dai pin 8 e 12. Quando tali pin sono a livello L (low) il motore è fermo. Quando il pin 8 passa a livello H (high) il transistor T2 va in conduzione ed il motore viene attraversato dalla corrente  $i_2$ . Quando passa a livello H il pin 12, T3 va in conduzione, ed insieme ad esso passa in conduzione anche T1; nel motore passa adesso la corrente  $i_1$ , che, scorrendo in senso opposto rispetto alla  $i_2$ , fa ruotare il motore nell'altro verso.

Le tre resistenze possono avere un valore di 4,7 kohm; i transistor T2 e T3 sono degli npn tipo BC547, mentre T1 è un pnp, tipo [BC327](#) o equivalenti.

Naturalmente il software deve impedire che i due pin di controllo (8 e 12 in questo caso) siano contemporaneamente a livello H.

Fonte: <http://www.raffaeleilardo.it> di Raffaele Ilardo