

Pensato per essere collegato a qualcosa

La scheda Genuino UNO nasce per essere collegata facilmente a tantissime cose, talmente tante che ad oggi nessuno le ha ancora individuate tutte

Rispetto al mondo dei personal computer, Genuino UNO - come i suoi predecessori - ha una marcia in più: quella di poter gestire in ingresso e in uscita segnali non solo digitali, ma anche analogici con grande immediatezza e semplicità grazie al software di programmazione e alle funzioni disponibili. Con queste possibilità, diventa quasi una passeggiata collegare a Genuino UNO praticamente qualsiasi cosa che sia "compatibile" con l'elettricità. Ci sono ovviamente dei limiti e delle modalità di collegamento per ogni dispositivo ed è evidente che il funzionamento a bassa tensione della scheda non permetta di pilotare direttamente dei dispositivi di potenza (lampade, motori e tutto quello che ha bisogno di più di 5V), ma con l'elettronica e l'elettricità ci sono sempre delle soluzioni per risolvere questo tipo di problematiche ricorrendo a qualche componente aggiuntivo e un circuito accessorio specifico.

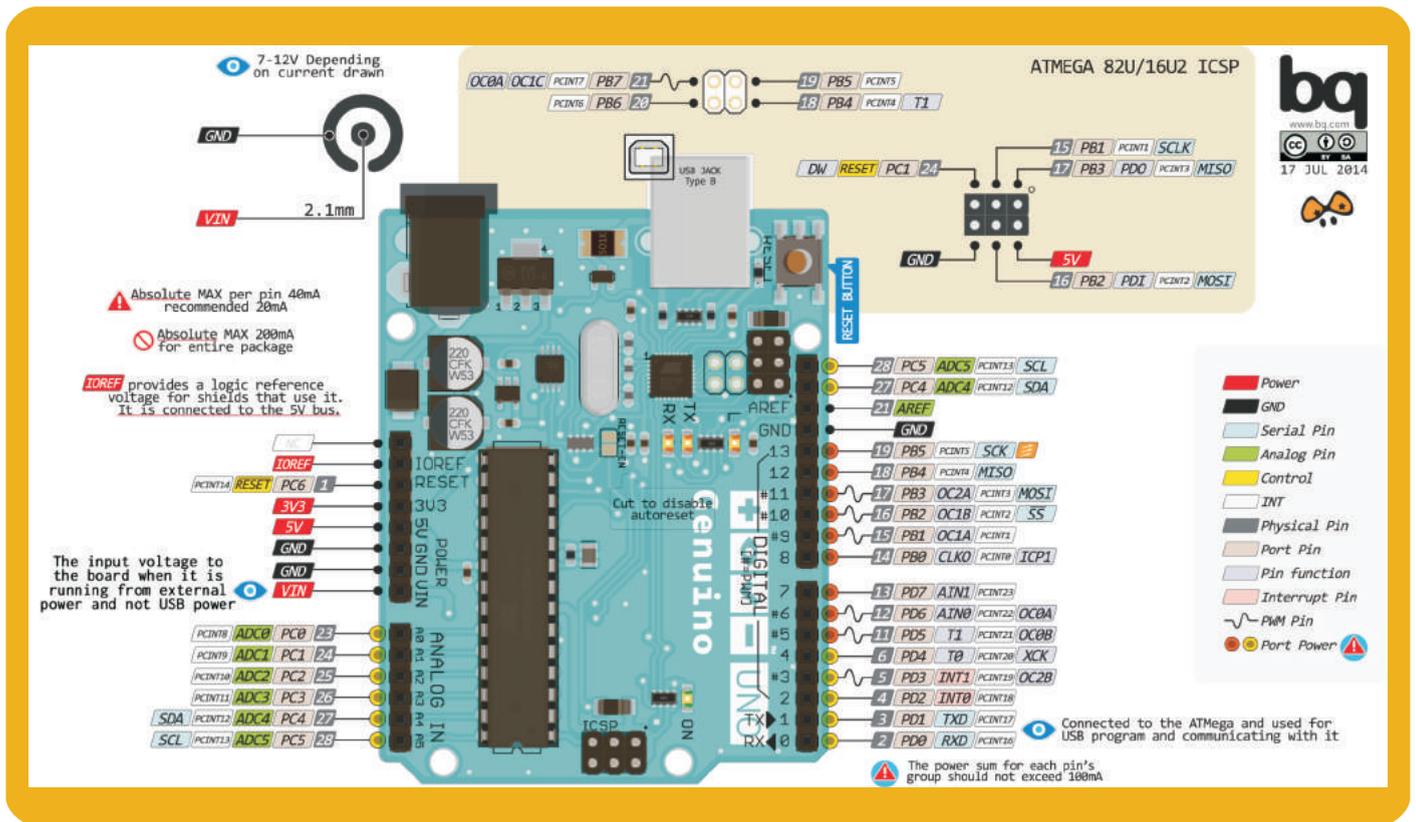
Input e Output, Analogico e Digitale: le quattro parole chiave

Se da una parte la flessibilità è uno degli obiettivi dei progettisti di microcontrollori, dall'altra la poca chiarezza può diventare un incubo per il neofita. Il microcontrollore Atmel scelto per Genuino UNO - il tipo ATmega328P - brilla per flessibilità, ma

Come i pin di ATmega328 diventano pin di Genuino UNO.

Per capire come il lavoro del team di sviluppo abbia semplificato la vita di chi non è un esperto di microcontrollori, basta guardare come ciascun pin del microcontrollore ATmega328 sia stato gestito assegnandogli una funzione specifica. Questo non impedisce di utilizzare i vari pin anche nelle altre modalità, ma permette di avere all'inizio un quadro semplificato e più accessibile.

Arduino function				Arduino function	
reset	(PCINT14/RESET) PC6	1	28	PC5 (ADC5/SCL/PCINT13)	analog input 5
digital pin 0 (RX)	(PCINT16/RXD) PD0	2	27	PC4 (ADC4/SDA/PCINT12)	analog input 4
digital pin 1 (TX)	(PCINT17/TXD) PD1	3	26	PC3 (ADC3/PCINT11)	analog input 3
digital pin 2	(PCINT18/INT0) PD2	4	25	PC2 (ADC2/PCINT10)	analog input 2
digital pin 3 (PWM)	(PCINT19/OC2B/INT1) PD3	5	24	PC1 (ADC1/PCINT9)	analog input 1
digital pin 4	(PCINT20/XCK/T0) PD4	6	23	PC0 (ADC0/PCINT8)	analog input 0
VCC	VCC	7	22	GND	GND
GND	GND	8	21	AREF	analog reference
crystal	(PCINT6/XTAL1/TOSC1) PB6	9	20	AVCC	VCC
crystal	(PCINT7/XTAL2/TOSC2) PB7	10	19	PB5 (SCK/PCINT5)	digital pin 13
digital pin 5 (PWM)	(PCINT21/OC0B/T1) PD5	11	18	PB4 (MISO/PCINT4)	digital pin 12
digital pin 6 (PWM)	(PCINT22/OC0A/AIN0) PD6	12	17	PB3 (MOSI/OC2A/PCINT3)	digital pin 11 (PWM)
digital pin 7	(PCINT23/AIN1) PD7	13	16	PB2 (SS/OC1B/PCINT2)	digital pin 10 (PWM)
digital pin 8	(PCINT0/CLKO/ICP1) PB0	14	15	PB1 (OC1A/PCINT1)	digital pin 9 (PWM)

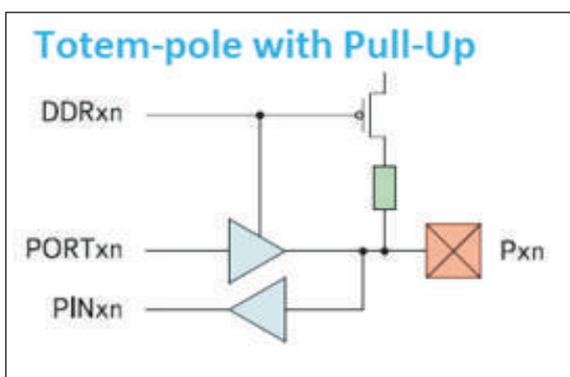


Qui sopra uno schema molto dettagliato di come i singoli pin della scheda siano collegati al microcontrollore o abbiano altre funzioni specifiche. Sotto lo schema delle porte di I/O digitali con la resistenza di pull-up attivabile via software.

allo stesso tempo potrebbe sollevare qualche perplessità in chi non ha grande esperienza nell'utilizzo di questi dispositivi. Anche questa volta, quindi, le scelte sono state fatte già dal team di sviluppo e la scheda si presenta con una ordinata serie di pin che dichiarano in modo evidente la loro funzione. Le tipologie di collegamento che ci offre Genuino UNO sono ingresso e uscita, per segnali digitali o analogici. Mentre le definizioni di ingresso e uscita sono chiare, dal punto di vista tecnico è il caso di spendere qualche parola per chiarire di cosa si tratti.

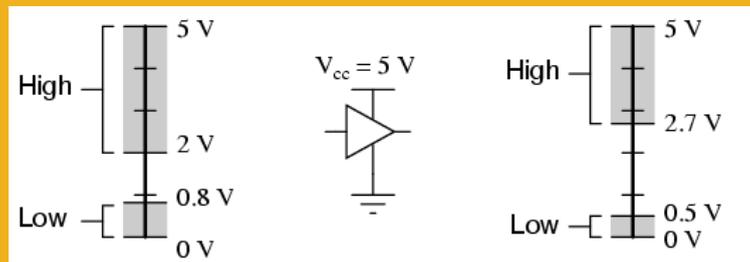
I/O Digitale

I pin digitali sono progettati per essere bidirezionali, ovvero in grado di passare dalla modalità di ingresso a quella di uscita attraverso la scrittura di opportuni valori nei registri di controllo (aree di memoria speciali interne al microcontrollore). Quando sono in modalità Output, possono assumere non solo valori logici 0 e 1 - corrispondenti a 0 volt e 5 volt, ma anche un terzo stato (tri-state) ad alta impedenza che di fatto li rende "ininfluenti" per qualsiasi circuito logico collegato. Non sono né 0 né 1 e allo stesso tempo non influiscono sullo stato di altre porte in uscita collegate. Questo stato risulta utile quando si vogliono collegare più dispositivi con un unico filo comune (o bus) per scambiare informazioni: chi vuole ricevere mette il pin in ingresso, chi vuole comunicare trasmette sul pin in modalità output e chi non deve far nulla va in modalità tri-state rendendosi neutro. In ingresso, con le porte ad alta impedenza, basta poco perché venga rilevato il livello logico 1 e per evitare incertezze il microcontrollore dispone di resistenze di *pull-up*, attivabili su ciascun pin via software. Se non è collegato nulla, la resi-



I livelli logici TTL

L'acronimo TTL - Transistor-Transistor Logic - definisce la tensione dei segnali utilizzati per comunicare. Un altro tipo di circuito, il CMOS, utilizza livelli diversi. Nel mondo digitale, però, non abbiamo a che fare con valori esatti e sempre come da specifiche e per questo sono state definite le aree entro le quali una tensione viene considerata come LOW (0), HIGH (1) o non definita. Nello schema qui sotto sono riportati i valori ritenuti accettabili; a sinistra i valori in ingresso e a sinistra quelli in uscita.



stenza di pull-up (una resistenza da circa 35k ohm verso il positivo) porta a 1 il valore letto, senza però che i 35k ohm possano creare problemi nel caso di un livello logico 1 generato da un dispositivo collegato.

Dal punto di vista squisitamente elettronico, i pin configurati in ingresso non rappresentano un carico di cui preoccuparsi e quindi anche sensori e dispositivi a bassa potenza possono generare i livelli logici necessari. Da 0 a 2,5 volt avremo una lettura di uno stato logico 0, mentre oltre 2,5 volt fino a 5 volt saranno considerati come stato logico 1. La tensione è ovviamente riferito alla massa.

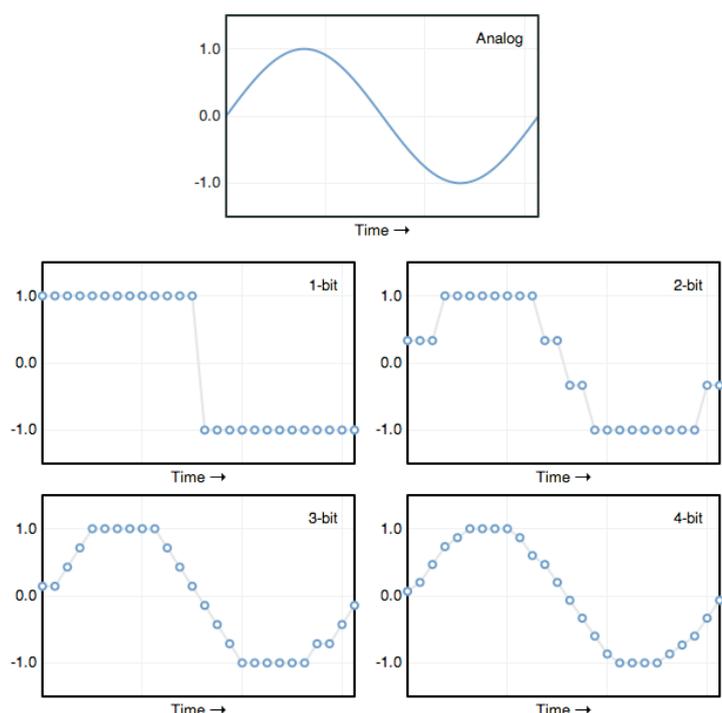
In uscita, i pin di Genuino UNO hanno dei circuiti di pilotaggio definiti come simmetrici, ovvero in grado di generare dei livelli 0 e 1 non solo con carichi ad alta impedenza (e quindi bassissimo assorbimento), ma anche su carichi che assorbono un massimo di 40 mA, fino a raggiungere un totale complessivo per tutti i vari pin di 150 mA. Ogni pin digitale non dovrebbe essere fatto lavorare con assorbimenti superiori a 20 mA per non sottoporre il microcontrollore a carichi fuori dalle specifiche di funzionamento ottimali. In pratica, potete collegare direttamente qualche LED, un buzzer e altri dispositivi senza problemi. La corrente è anche sufficiente ad alimentare dei sensori, che possono quindi essere accesi e spenti in base alle necessità, economizzando il consumo energetico.

Ingresso Analogico

Il comportamento delle schede Genuino con i segnali analogici non è simmetrico fra ingresso e uscita, ovvero non possiamo generare segnali analogici con la stessa precisione che abbiamo in ingresso e il tipo di segnale analogico non è neppure un vero livello di tensio-

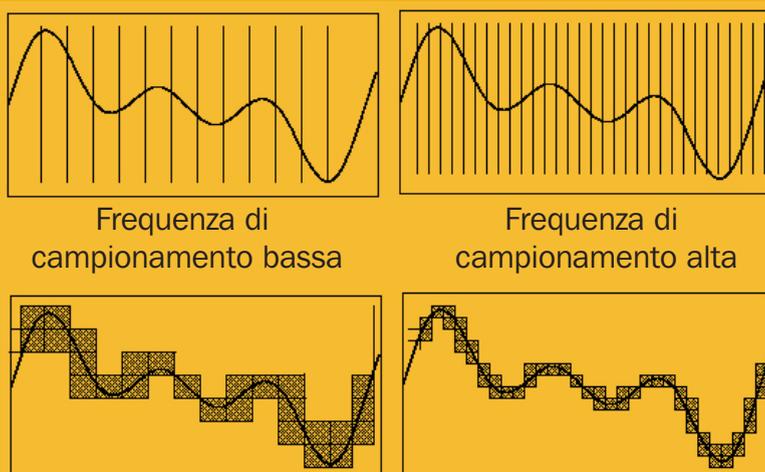
Qui sotto gli effetti del campionamento a 1, 2, 3 e 4 bit di un segnale analogico.

Da notare come il segnale venga approssimato su scalini che dipendono dal numero di bit usati.



La frequenza di campionamento

Oltre al numero di bit, nella conversione da analogico a digitale conta molto anche la velocità con cui si esegue il campionamento: più questa è elevata e più sarà la possibilità di catturare in digitale le frequenze più alte grazie al miglior dettaglio sull'asse del tempo. I CD audio hanno una frequenza di campionamento di 44,100 Hz a 16 bit. I bit definiscono il dettaglio sull'asse del livello.



ne fra un minimo e un massimo a meno che non si usi un circuito d'integrazione. Questo mostra chiaramente come ci siano dei limiti imposti dal costo contenuto dell'hardware, limiti ai quali si è posto un rimedio "software".

Innanzitutto vale la pena capire meglio come un dispositivo digitale gestisca la lettura di un segnale analogico, attraverso il processo della conversione o campionatura, svolto dal convertitore analogico digitale (A/D).

Il convertitore A/D è un circuito che misura la tensione in ingresso ogni volta che

viene interrogato. Si tratta di una misurazione istantanea che viene poi convertita in un valore binario. L'intervallo di tensioni misurabile viene suddiviso in tanti livelli discreti - o scalini - quanti sono i valori esprimibili con i bit a disposizione. Se il convertitore è a 8 bit, potrà esprimere 256 valori; se è a 16 bit ne esprimerà 65536. Il microcontrollore utilizzato ha un convertitore A/D a 10 bit e quindi può esprimere 1024 valori diversi.

È importante comprendere come il numero di scalini (valori) gestiti dal convertitore non abbia un rapporto diretto con l'estensione dei valori stessi: questa è infatti legata all'intervallo di misura che va da 0 alla tensione di alimentazione/riferimento del convertitore (5V per Genuino UNO). I bit di risoluzione ci permettono solo di avere una lettura più o meno precisa. In pratica, senza intervenire con circuiti accessori, con Genuino UNO possiamo leggere valori analogici fra 0 e 5 volt, con 1024 livelli discreti.

La lettura è lineare e quindi il valore 512 corrisponde a 2,5V, mentre 128 corrisponde a 0,625 volt e così via. Un singolo livello è pari a 0,0048 volt e questa di

fatto è la risoluzione teorica del convertitore della nostra scheda Genuino UNO. Oltre alla risoluzione, è anche importante la frequenza con cui la lettura viene ripetuta - o frequenza di

Frequenza di campionamento	Qualità audio	Banda passante
11.025 Hz	Radio AM di bassa qualità	0 - 5.512 Hz
22.050 Hz	Radio FM di scarsa qualità	0 - 11.050 Hz
32.000 Hz	Radio FM di buona qualità	0 - 16.000 Hz
44.100 Hz	Qualità CD Audio	0 - 22.050 Hz
48.000 Hz	Qualità DVD	0 - 24.000 Hz
96.000 Hz	Audio ad Alta Definizione DVD	0 - 48.000 Hz

campionamento -, specie se si vogliono leggere segnali che variano molto rapidamente, come ad esempio un segnale audio. Diciamo subito che Genuino UNO può arrivare al massimo a registrare della voce con una fedeltà limitata, mentre per la musica è necessario uno shield dedicato. La frequenza massima di lettura senza usare "trucchi" è di circa 9 kHz e questo permette di campionare frequenze fino a 4,5 kHz. Il convertitore A/D di Genuino UNO è disponibile su cinque pin e questo permette di dotare la scheda di più sensori, non necessariamente uguali fra loro, per avere una lettura del mondo esterno sotto forma di valori analogici.

Uscita analogica - PWM

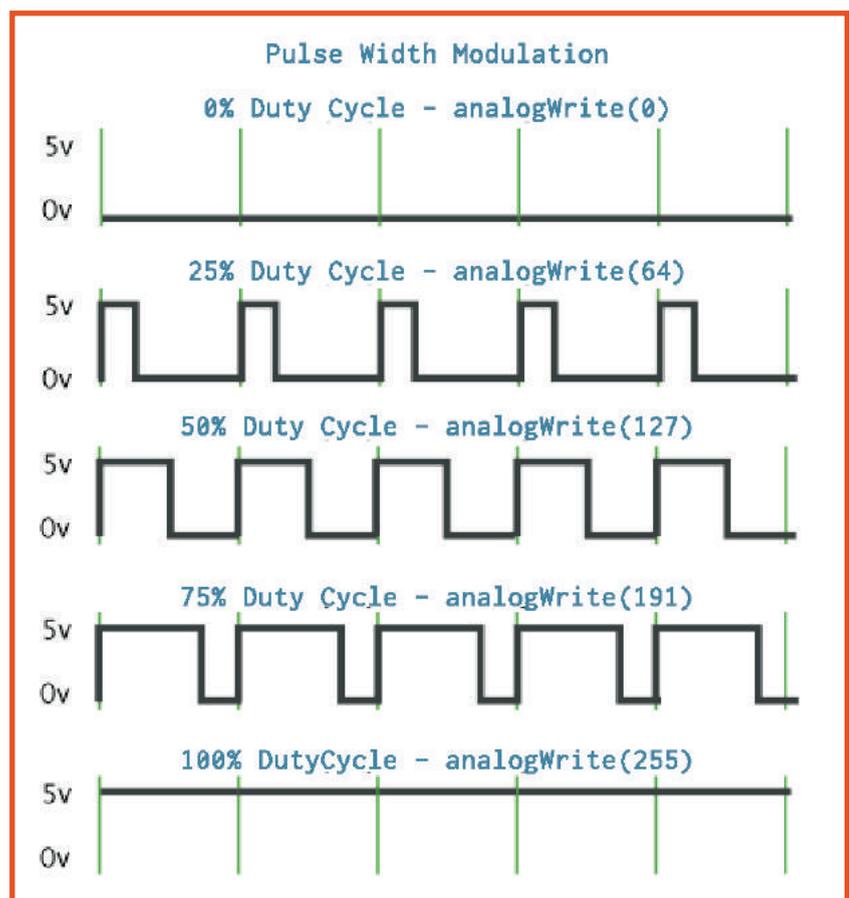
Le uscite analogiche di Arduino UNO non sono realmente in grado di produrre una tensione variabile - sempre a step discreti - da 0 a 5 V, ma sfruttano la tecnica del PWM per ottenere un effetto in molti casi equivalente. PWM è l'acronimo di Pulse Width Modulation - modulazione dell'ampiezza d'impulso - e da qualche anno è la tecnica utilizzata per modulare l'intensità dei LED.

Sostanzialmente il segnale prodotto varia solo fra i due livelli logici 0 e 1, pari a 0 e 5 V, ma il rapporto fra i livelli viene variato all'interno del singolo ciclo per simulare una variazione di tensione. Voglio fornire 0,5 V al mio dispositivo? Genero un segnale PWM con il 10% del ciclo positivo. Voglio fornire 2,5V? Allora il ciclo positivo è del 50%.

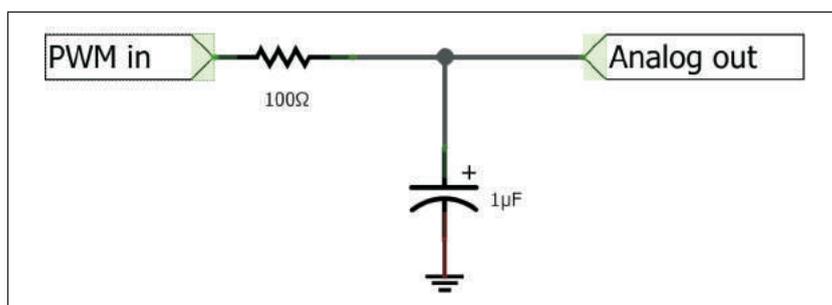
Se facciamo una media nel tempo della tensione, otterremo effettivamente una

serie di livelli variabili fra 0 e 5V pari agli step disponibili per il duty cycle del PWM. Nel caso di Genuino il PWM è a 8 bit, ovvero ha 256 livelli. Nel caso specifico dei LED, il PWM determina la maggiore o minore luminosità percepita gestendo il tempo in cui il led stesso è acceso. Una volta che si supera una certa frequenza nel lampeggio, l'occhio umano non percepisce più la "vibrazione" e vede solo un LED con luminosità più o meno forte. Variando la durata del rapporto acceso/spento si ottiene quindi una luce più o meno intensa, come se si pilotasse il LED con una

Le uscite analogiche di Genuino UNO utilizzano la modulazione d'ampiezza dell'impulso (PWM) per simulare 256 livelli fra 0 e 5 volt.



tensione variabile. Poiché la tensione varia sempre fra 0 e 5 volt, è importante ricordare che ci sono dispositivi che non sopportano bene i 5V e quindi vanno protetti con resistenze di limitazione della corrente o altri accorgimenti. Una media di 2,5V corrisponde a 5V per il 50% del tempo. In pratica, quindi, predisponete il circuito per il suo funzionamento "sicuro" con 5V, dopodiché con il PWM riducete la forza o l'intensità senza il problema di evitare la massima potenza per non bruciare il componente.



Da PWM a tensione

Se è proprio indispensabile avere una tensione e non un segnale PWM, si può usare un semplice circuito che fa la media nel tempo degli impulsi e li trasforma quindi in una vera tensione. Si tratta

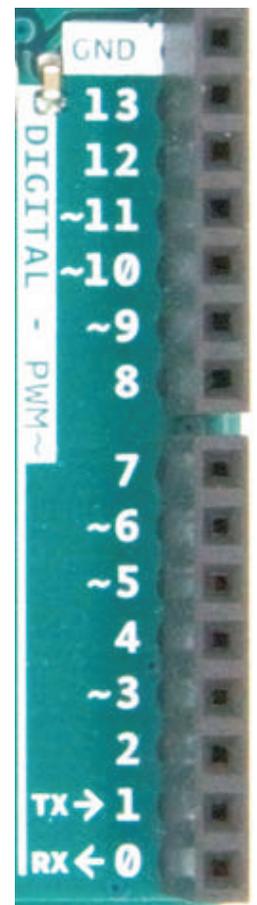
di un filtro passa basso composto da una resistenza e da un condensatore.

A seconda dei valori di ingresso, del carico collegato in uscita e della frequenza del segnale PWM, vanno selezionati opportunamente i valori di R e C, ricordando che comunque questo tipo di filtro ha una propria velocità di risposta alle variazioni nel segnale PWM e inoltre lascia passare come "ronzio" di fondo la portante del segnale PWM (ovvero la frequenza dell'onda quadra di cui varia il rapporto on/off).

Diamo un senso a tutte le scritte: cosa posso fare con ciascuno dei PIN della scheda

Ora che è chiaro come funzionino input, output, analogico e digitale, puoi guardare la scheda Arduino UNO con maggiore confidenza, iniziando a immaginare quello che puoi collegare.

Sul lato destro della scheda si trovano quattordici pin definiti come "Digital" e affiancati in alcuni casi dal segno "~". Questi pin possono essere utilizzati indifferentemente come ingressi o uscite tramite le funzioni `pinMode()`, `digitalWrite()`, e `digitalRead()`. Ciascun pin digitale può fornire o assorbire **20 mA** e fornisce 5V in uscita quando attivo. Come indicato in precedenza, se impostato su input, ciascun pin



può essere tenuto a livello alto tramite una resistenza di pull-up. Se non viene fornito qualche comando specifico in fase di inizializzazione (setup), tutti i pin digitali sono in input e le resistenze di pull-up sono scollegate. Alcuni pin digitali hanno delle etichette di testo specifiche e svolgono opzionalmente funzioni speciali, in base allo sketch caricato e al setup.

Porta Seriale: pin 0 (**RX**) and pin 1 (**TX**). **Permette** di dialogare in seriale - con livelli TTL - con altri dispositivi ricevendo sul pin 0 e trasmettendo sul pin 1. I due pin sono collegati ai pin corrispondenti del chip ATmega8U2 USB-to-TTL.

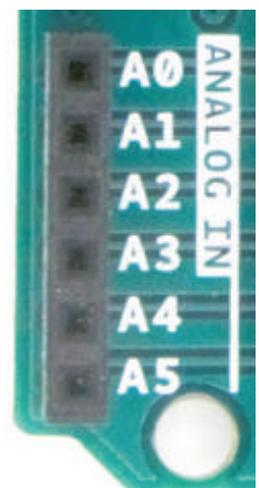
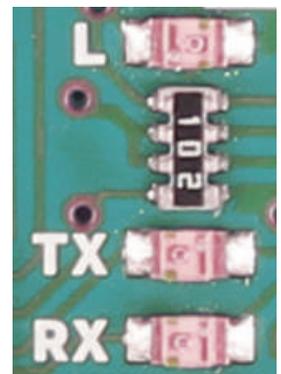
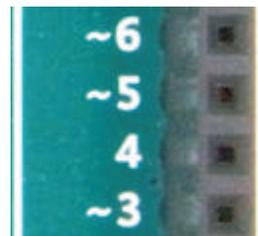
Interrupt Esterni: pin 2 e pin 3. Questi due pin permettono di attivare un interrupt - particolare situazione in cui il microcontrollore viene portato ad eseguire del codice specifico, se previsto - quando il livello subisce una variazione secondo una delle quattro modalità previste. Queste sono LOW (valore basso), CHANGE (tutte le volte che il pin digitale cambia stato), RISING (l'interrupt avviene quando il pin passa da 0 a 1) e FALLING (il pin passa da 1 a 0). La funzione con cui gestire l'interrupt è `attachInterrupt()`.

PWM (~): pin 3, 5, 6, 9, 10, e 11. Questi pin supportano la generazione di segnali PWM direttamente gestiti dall'hardware ATmega; la frequenza dell'onda quadra generata dipende da alcuni parametri che possono essere modificati, ma di default è circa 490 Hz. Il valore del PWM varia da 0 a 255 con 8 bit di risoluzione. La funzione specifica per gestire il PWM su questi pin è `analogWrite()`.

SPI: pin 10 (**SS**), 11 (**MOSI**), 12 (**MISO**) e 13 (**SCK**). Questi pin supportano il protocollo di comunicazione seriale sincrono SPI - Serial Peripheral Interface - che viene utilizzato per comunicazioni su brevi distanze verso una periferica o un altro microcontrollore. La gestione di SPI avviene tramite una apposita libreria inclusa nell'Arduino Software (IDE).

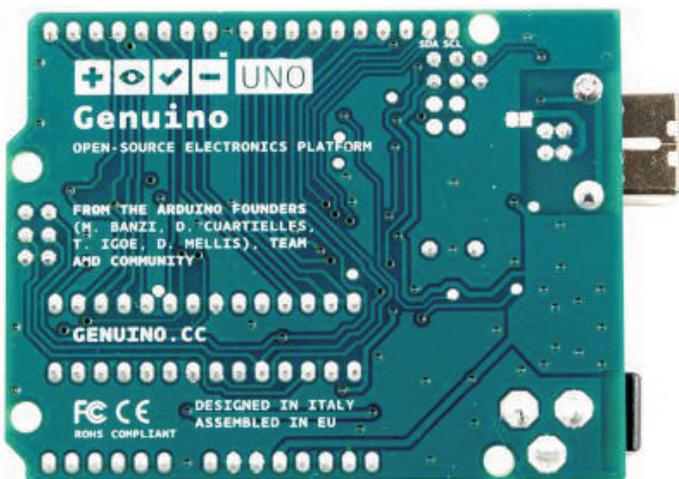
LED: pin 13. Dato che far lampeggiare un LED è fra le prime attività consigliate al neofita della piattaforma Arduino, il team degli sviluppatori ha pensato di inserire direttamente sulla scheda un LED collegato opportunamente al pin digitale 13. Per accenderlo basta impostare il pin 13 su HIGH (`digitalWrite(13, HIGH);`), mentre per spegnerlo il valore deve essere LOW (`digitalWrite(13, LOW);`).

Nella parte bassa a sinistra si trovano i cinque pin di ingresso analogici, denominati A0, A1... A5. Su di essi si può leggere una tensione compresa fra 0 e 5 V riferiti alla massa della scheda. I 1024 valori - pari a 10 bit di risoluzione - sono corrispondenti all'intervallo di lettura. Per rendere il livello massimo della scala diverso



da 5V è possibile usare sia le sorgenti interne che una sorgente esterna collegata al pin AREF - che si trova in alto a destra, sopra il pin digitale 13 - e la funzione `analogReference()`. Usare un limite superiore più basso, come `INTERNAL` da 1,1 volt, permette di avere una maggiore risoluzione sulle variazioni di una tensione debole in quanto ogni gradino rilevabile è pari a $\text{Riferimento}/1024$. La tensione di riferimento NON PUO' superare i 5V o essere inferiore a 0V, altrimenti si può danneggiare il microcontrollore.

Anche su questi pin sono presenti delle funzionalità per applicazioni particolari:

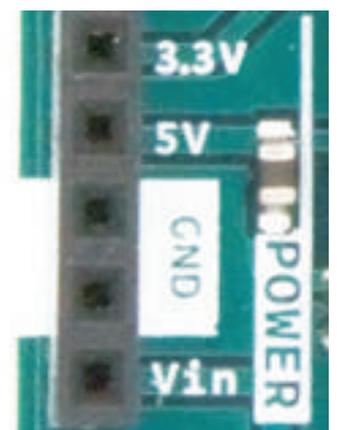


TWI: pin A4 (**SDA**) e pin A5 (**SCL**) - disponibili come nuovi collegamenti sopra il pin AREF e marchiati SDA e SCL sulla parte rame della scheda Rev. 3 - per comunicare con i dispositivi I2C e TWI che supportano la comunicazione con due soli fili. Per gestire questo protocollo Genuino UNO dispone di una libreria standard dedicata, chiamata `Wire`.

Sempre a sinistra, sopra il connettore degli ingressi analogici si trova la serie di collegamenti legati all'alimentazione.

VIN. Genuino UNO può essere alimentato in vari modi. Quando è collegato al computer tramite USB preleva da lì i 5V, altrimenti li può ottenere da un'alimentazione fra 7 e 12 volt fornita tramite un alimentatore esterno collegato alla presa di fianco a quella USB, oppure tramite questo pin. Da notare che la presa e questo pin sono collegati fra loro e quindi se l'alimentatore esterno fornisce per esempio 9 volt, su VIN troveremo 9 volt.

5V. L'alimentazione regolata e stabilizzata, usata per il microcontrollore e la circuiteria presente sulla scheda. Questa tensione arriva o dal regolatore presente sulla scheda a cui è fornita una tensione fra 7 e 12 volt, o dalla porta USB, alla quale si può collegare un computer che fornisce l'alimentazione o anche un semplice alimentatore con porta USB, purché di qualità.



3V3. Una tensione di 3,3 volt generata dal regolatore della scheda. La corrente massima disponibile è di 50 mA.

GND. La massa della scheda (0 volt).

Reset. Il pin da mettere a massa per generare il segnale di reset, in alternativa a quello generato con il pulsante presente in alto a destra. Viene usato per montare un tastino di reset su queglii shield che per la loro forma rendono inaccessibile il tastino di reset originale.

IOREF permette agli shield di adattarsi alla tensione di alimentazione fornita dalla scheda. Si tratta di un nuovo pin della Rev. 3. Sopra IOREF si trova un ulteriore nuovo pin che al momento è riservato per impieghi futuri.

ICSP (In-Circuit Serial Programming) è un connettore da sei pin presente in centro nella parte bassa della scheda ed è utilizzabile per programmare direttamente - attraverso dell'hardware specifico - il microcontrollore.

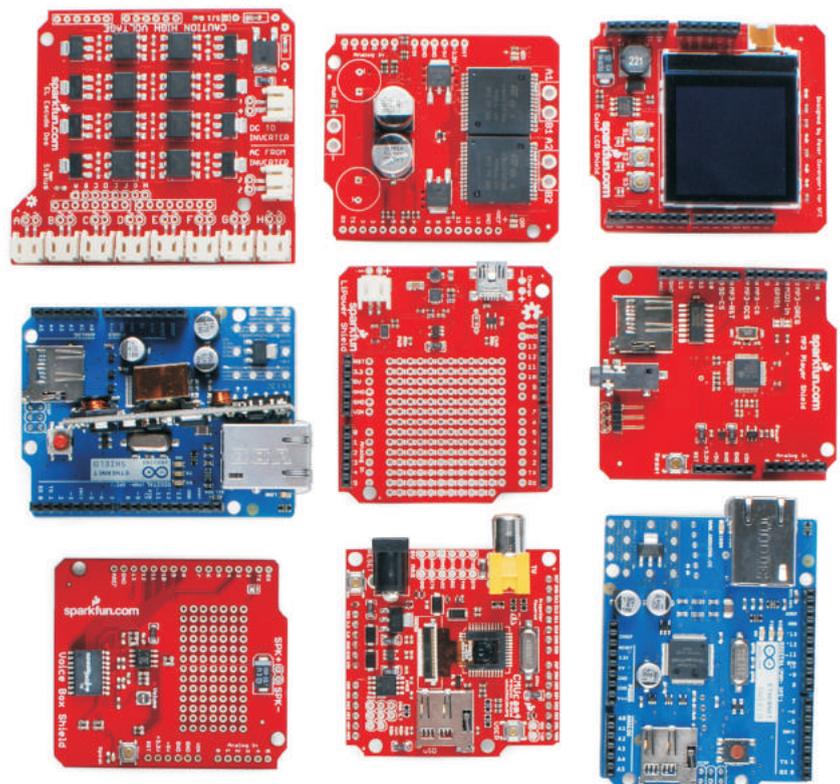


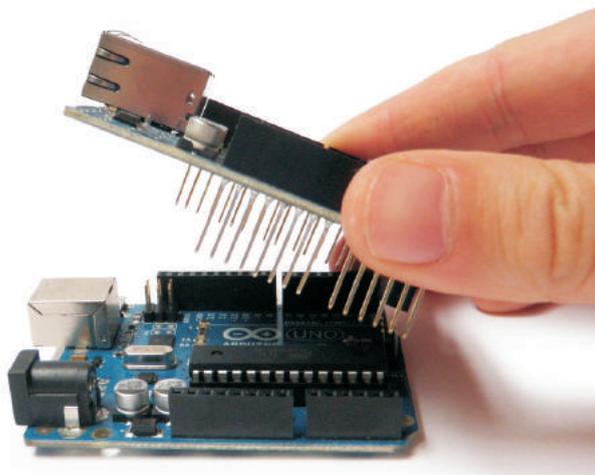
Gli shield

Una delle caratteristiche peculiari del progetto Arduino è la disponibilità di hardware accessorio - denominato shield - che permette con una semplicità estrema, pari a quella dei mattoncini Lego, l'espansione delle funzionalità della scheda di base.

La presenza dei connettori sui bordi della scheda Genuino UNO e la disponibilità degli schemi elettrici ha permesso a tutti coloro che ne hanno sentito il bisogno di sviluppare dell'hardware aggiuntivo con la ragionevole certezza di un collegamento immediato e la compatibilità con le varie versioni di scheda.

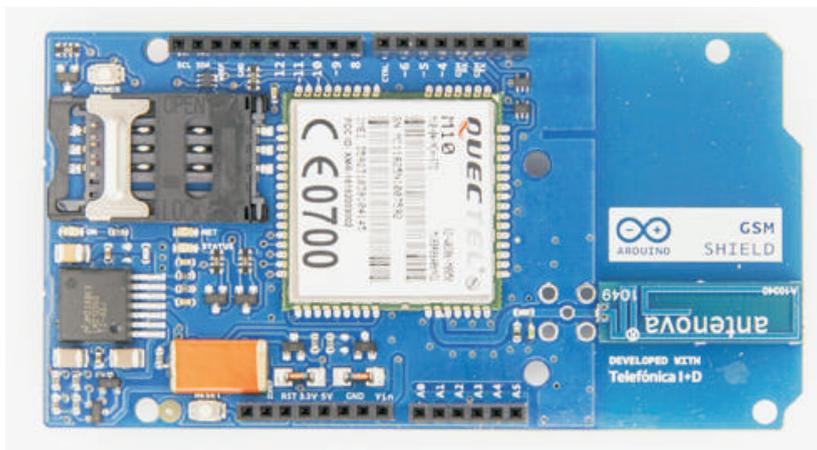
Tutti i vari segnali/pin che abbiamo visto nel capitolo precedente sono infatti quello che serve per aggiungere altri dispositivi mantenendone il pieno con-





wireless, della gestione dei motori per la robotica, dell'interfacciamento con LED e matrici, della gestione audio ad alta fedeltà, della prototipazione e della sperimentazione.

Dietro ad ogni shield c'è spesso una libreria dedicata e quindi ogni nuova caratteristica diventa disponibile al software tramite funzioni che si comportano come quelle standard che hai già imparato ad utilizzare.



Ufficiali e non

Il passaggio dalla vecchia alla nuova produzione ha creato qualche cambiamento nell'assortimento dei prodotti e quindi con il marchio Genuino, al momento in cui scriviamo (gennaio 2016), non ci sono shield in vendita, ma vale la pena ricordare quello WiFi annunciato in America - Arduino WiFi Shield 101 - che dispone di un nuovo chip con la

crittografia integrata per semplificare le connessioni IoT (Internet of Things) protette e il sempreverde GSM Shield realizzato assieme a Telefonica e tutt'ora cavallo di battaglia per molti progetti.

La facilità con cui è possibile connettere sensori, attuatori e dispositivi anche di una certa complessità, assieme alla crescente integrazione di dispositivi a bordo della scheda stessa - si veda la nuova Genuino 101 - stanno un po' cambiando l'atteggiamento degli sviluppatori di hardware.

Da qualche tempo è nata la tendenza a non realizzare più uno shield, ma di integrare nell'intero circuito dedicato a una funzione specifica anche la circuiteria di Genuino. Così è successo per alcune schede per stampante 3D e per dei

