



KIT D'AVVIAMENTO ECCELLENTE

TUTORIAL FOR UNO

Introduzione

La nostra società

Costituita nel 2011, Elegoo Inc. è un produttore ed esportatore professionale che si occupa di progettazione, sviluppo, produzione e commercializzazione di Arduino, stampanti 3D, Raspberry Pi e STM32. Ci si trova a Shenzhen che è conosciuta come la Silicon Valley cinese. Tutti i nostri prodotti rispondono alle norme internazionali di qualità e sono molto apprezzate in una varietà di mercati differenti in tutto il mondo.

Sito ufficiale: <http://www.elegoo.com>

IT Amazon negozio: <http://www.amazon.it/shops/A1780XYQ9DFQM6>

Il Nostro Tutorial

Questo tutorial è orientata per i principianti. Imparerete tutte le informazioni di base su come utilizzare Arduino controllore di bordo, sensori e componenti. Se si vuole studiare Arduino in modo più approfondito, si consiglia di leggere la Arduino libro di cucina scritto da Michael Margolis.

Il nostro servizio

Se avete qualsiasi domanda, commento o suggerimento sulla nostra azienda, prodotto o tutorial, non esitate ad inviarci una email EUservice@elegoo.com. Siamo impegnati a un continuo miglioramento in modo che il commenti sono insostituibile.

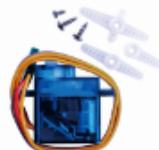
Elenco dei componenti

www.elegoo.com

Motore passo-passo



Servo (SG90)



IR Receiver
Module
1PCS



5V Relay
1PCS



Elegoo UNO R3
microcontrollore



LCD1602 display



Uln2003 modulo
del driver
motore
passo-
passo



modulo di
alimentazione
di breadboard



Scheda di espansione
del prototipo



Contattaci: EUservice@elegoo.com

www.elegoo.com

4 -cifre -7-Segmento Display



Esposizione di segmento



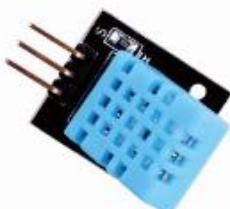
Interruttore palla



6v DC motore



DHT11 modulo



Modulo sensore ad ultrasuoni



Contattaci: EUservice@elegoo.com

 www.elegoo.com

Interruttori a pulsante
5 pezzi



Potenziometro



Passivo Ronzatore



batteria e
connettore 9V



Ronzatore attivo



Modulo di Joystick



USB fili



F-M fili 10 pezzi



F-F fili 65 pezzi



IR remote



Breadboard

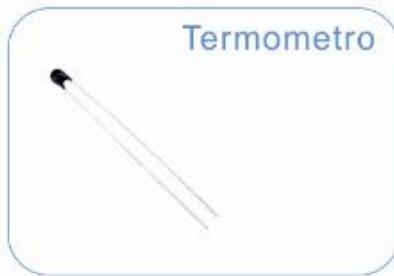


74HC595 IC



Contattaci: EUservice@elegoo.com

 www.elegoo.com



Contattaci: EUservice@elegoo.com

Catalogo dei corsi

Lezione 0: Installazione di IDE	8
Lezione 1: Add librerie.....	16
Lezione 2: Lampeggio	20
Lezione 3: LED	27
Lezione 4: RGB LED	33
Lezione 5: Ingressi Digitali	39
Lezione 6: Otto LED con 74HC595.....	43
Lezione 7: Il Monitor Seriale.....	49
Lezione 8: Fotocellula.....	55
Lezione 9: Facendo I Suoni.....	59
Lezione 10: Passivo Ronzatore	63
Lezione 11: Sfera Interruttore	66
Lezione 12: Relay.....	69
Lezione 13: 74HC595 e Display Segmento	72
Lezione 14: 4 -cifre -7-Display Segmento	77
Lezione 15: Servo	81
Lezione 16: LCD Display.....	84
Lezione 17: Termometro	89
Lezione 18: Modulo sensore ad ultrasuoni	93
Lezione 19: Temperatura e umidità di sensore di DHT11	97
Lezione 20: Modulo di joystick analogico	102
Lezione 21: Modulo di IR Ricevitore.....	105
Lezione 22: DC Motori.....	109
Lezione 23: Motore passo-passo.....	113
Lezione 24: Il controllo del motore passo-passo con telecomando.....	115

Lezione 0: Installazione di IDE

Introduzione

In questa lezione imparerai come configurare il computer per utilizzare Arduino e come fare per le lezioni che seguono.

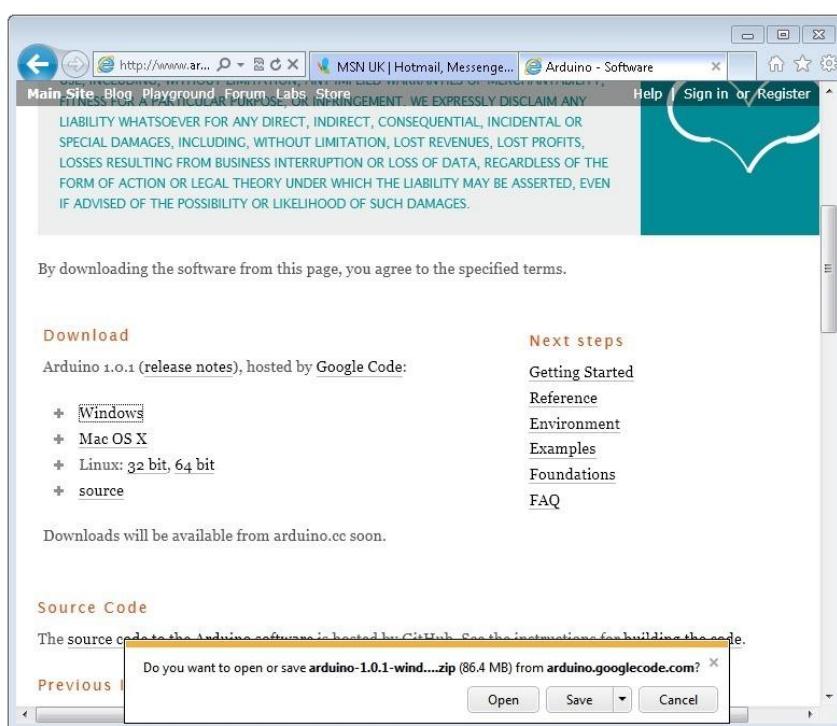
Il software di Arduino che verrà utilizzato per programmare il vostro Arduino è disponibile per Windows, Mac e Linux. Il processo di installazione è diverso per le tre piattaforme e sfortunatamente vi è una certa quantità di lavoro manuale per installare il software. Non vi è alcun programma di installazione; invece, è necessario decomprimere un file che vi dà una cartella contenente il programma Arduino e pochi altri oggetti.

In una fase separata, è necessario installare i driver USB.

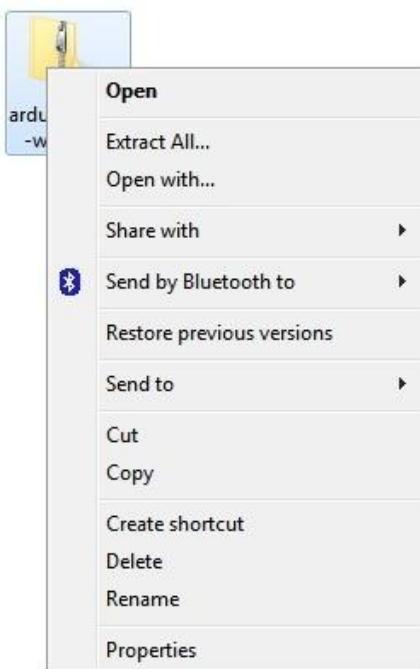
Installazione di Arduino (Windows)

Cominciate visitando il sito web Arduino.cc. A partire da aprile 2014, si consiglia di utilizzare v1.05 come 1.5 è ancora in versione beta. Se 1.5 non è più in versione beta è il tempo di leggere questo, si può provare!

Inizia scaricando il file zip per Windows. C'è solo una versione del software, se si sta utilizzando Windows XP attraverso Windows 7.



Quando il file zip scaricato, estrarre il contenuto sul desktop, facendo clic destro sul file e selezionando 'Estrarre Tutti...' dal menu a comparsa.



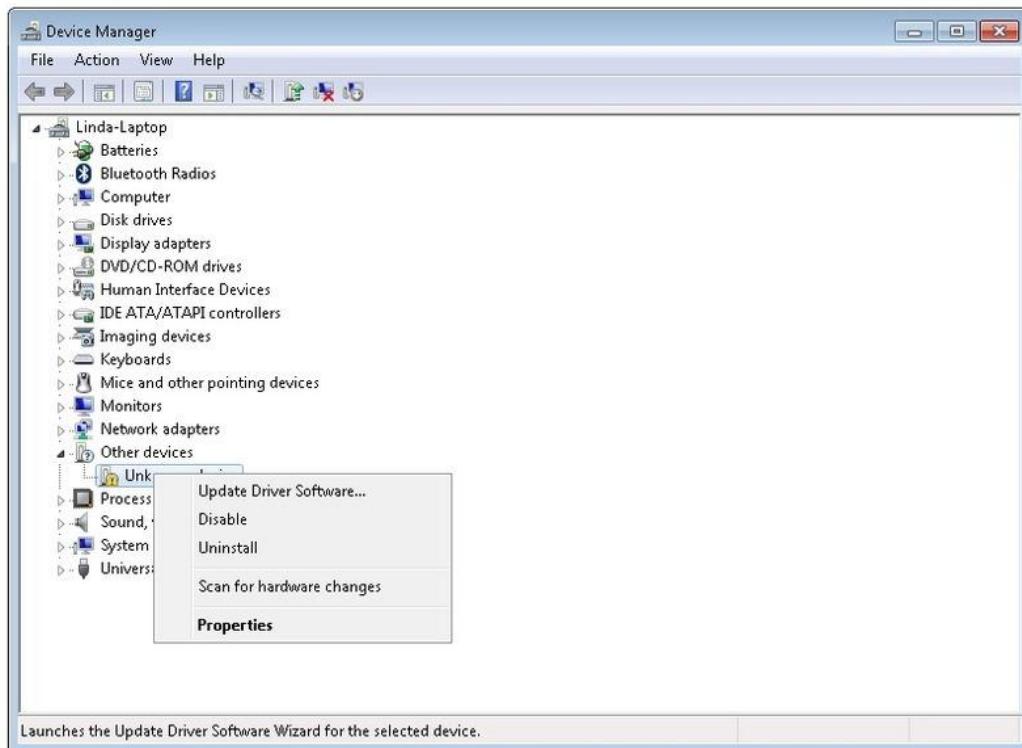
Prossima, selezionare il desktop e fare clic su 'Estrarre'. È possibile spostare da qualche altra parte sul vostro computer in seguito, semplicemente spostando la cartella, ma per ora, basta tenere sul desktop.

La cartella Arduino contiene sia il programma Arduino stesso ed i piloti che permettono al Arduino deve essere collegato al computer tramite un cavo USB. Prima di lanciare il software di Arduino, che si sta per installare i piloti USB.

Collegare un'estremità del cavo USB nella Arduino e l'altra in una presa USB del computer. La luce di alimentazione sul LED si accende e si può ottenere un messaggio 'Trovato nuovo hardware' da Windows. Ignorare il messaggio e annullare tutti i tentativi che Windows fa per cercare di installare automaticamente i driver per voi.

Il metodo più affidabile di installazione dei piloti USB è quello di utilizzare Gestione periferiche. Questo è accessibile in modi diversi a seconda della versione di Windows. In Windows 7, è necessario innanzitutto aprire il Pannello di controllo, quindi selezionare l'opzione per visualizzare le icone, e si dovrebbe trovare la Gestione periferiche nell'elenco.

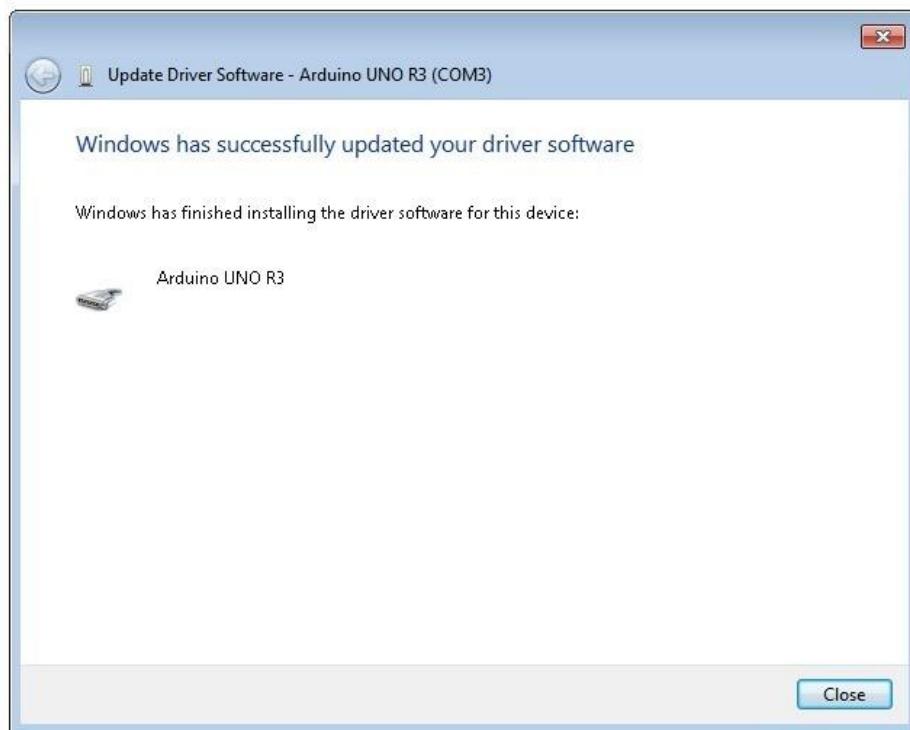
Sotto "altri dispositivi", si dovrebbe vedere una icona per 'dispositivo sconosciuto' con un piccolo triangolo di avvertimento giallo accanto a esso. Questo è il vostro Arduino.



Fai clic destro sul dispositivo e selezionare l'opzione di menu superiore (Aggiornamento software driver ...). Verrà quindi chiesto a uno 'Cerca automaticamente un driver aggiornato' o 'Cerca sul computer il software del driver'. Selezionare l'opzione per navigare e accedere alla Arduino-1.0.2-windows \ arduino1.0.2 \ drivers.



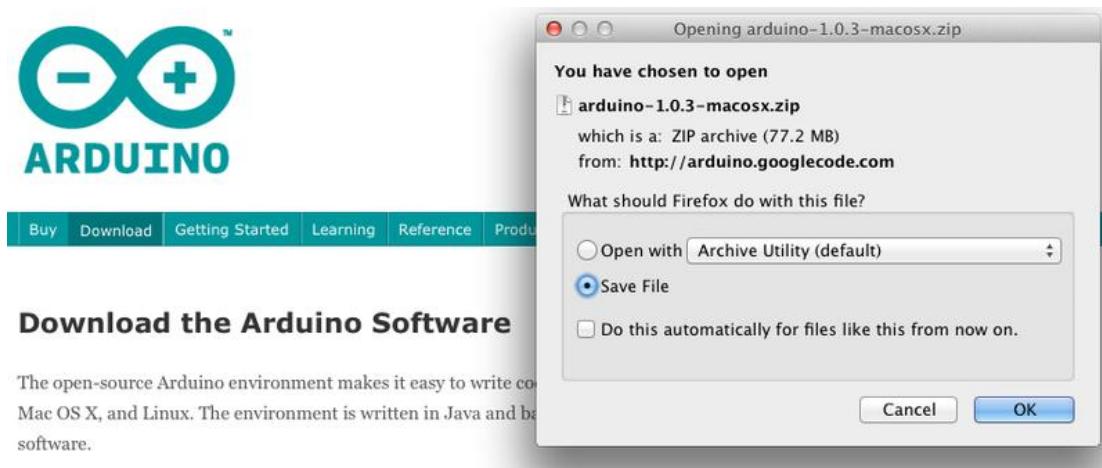
Fare clic su 'Prossima' e si può ottenere un avviso di protezione, in caso affermativo, consentire al software da installare. Una volta che il software è stato installato, si otterrà un messaggio di conferma.



Tutto qui. Ora siete pronti per l'azione, in modo da saltare la prossima sezione per l'installazione su Mac e Linux e si sposta direttamente in 'schede e porte'.

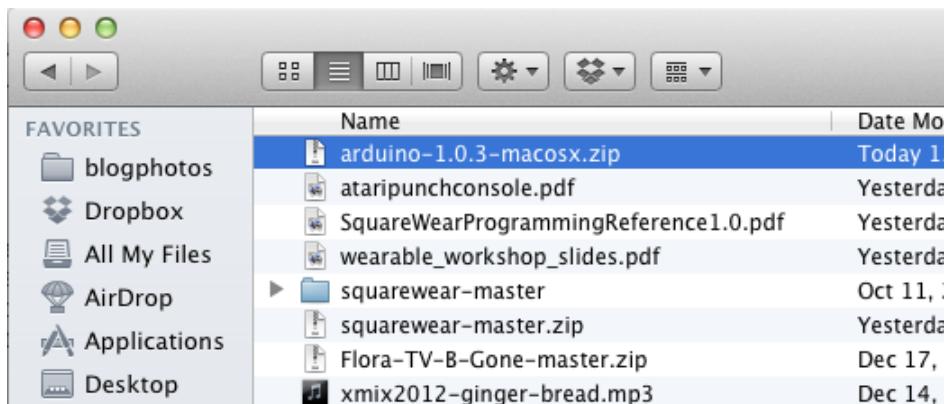
Installazione di Arduino (Mac e Linux)

Cominciate visitando il sito web e il download Arduino.cc l'IDE corrispondente per il sistema operativo. A partire da aprile 2014, si consiglia di utilizzare v1.0.5 come 1.5 è ancora in versione beta. Se 1.5 non è più in versione beta è il tempo di leggere questo, si può provare!



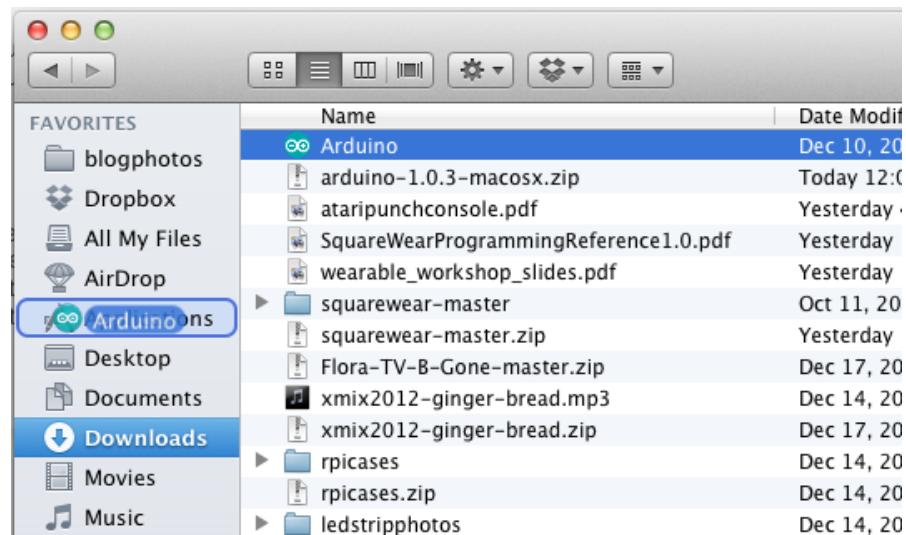
The screenshot shows the Arduino website's download section. Below the download link, there is a note: "The open-source Arduino environment makes it easy to write code for Arduino boards, breadboards, and other microcontroller-based projects. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on the Processing software."

Salvare il software di installazione sul desktop o dovunque

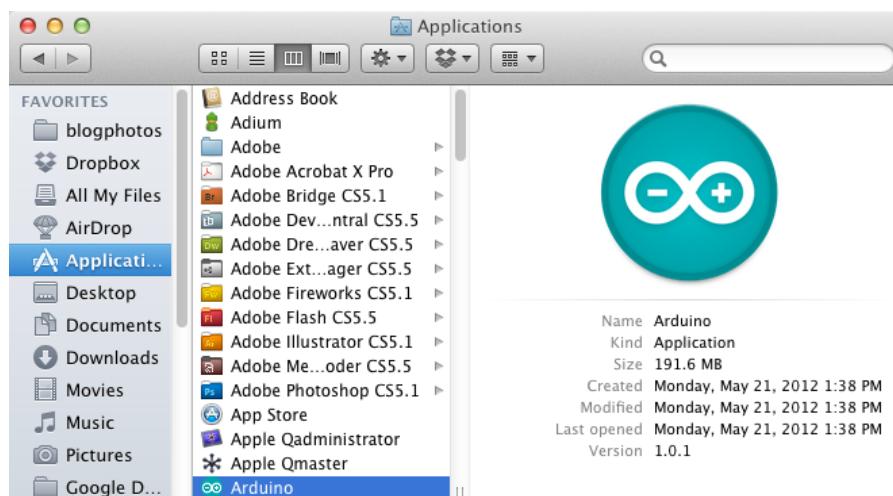


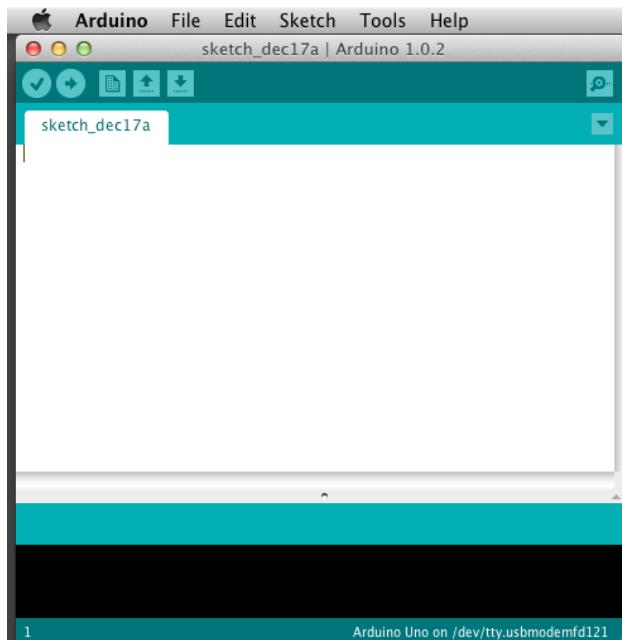
Il processo di installazione del software di Arduino sul Mac è molto più facile che sul PC.

Come prima, il primo passo è quello di scaricare il file. Nel caso di Mac, si tratta di un file zip



Una volta scaricato, fare doppio clic sul file zip, che estrarre un singolo file chiamato 'Arduino.app'. Questa è l'intera applicazione Arduino; basta trascinarlo nella cartella Applicazioni.



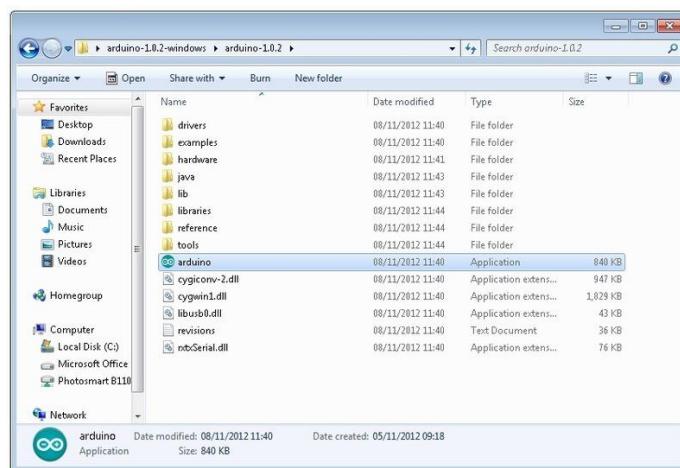


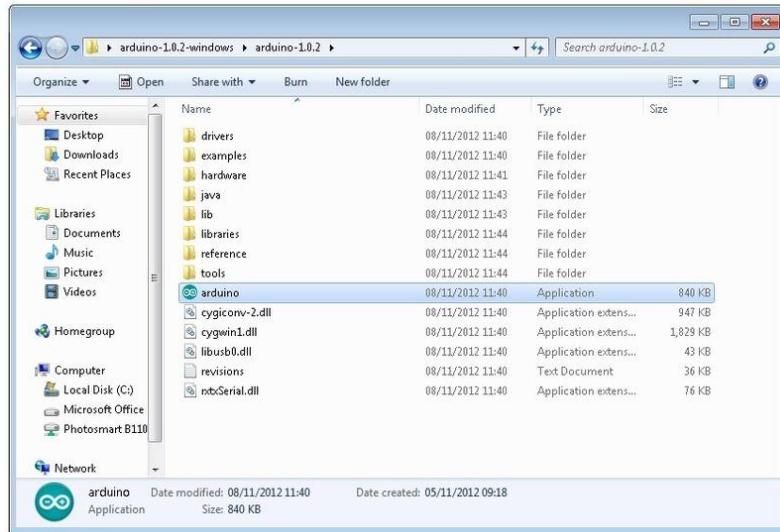
Ora è possibile trovare e avviare il software di Arduino nella cartella Applicazioni. Come avete intenzione di usarlo spesso, si potrebbe desiderare di fare clic destro sulla sua icona nel dock e impostarlo su 'Mantieni nel Dock'.

Ci sono molte distribuzioni di Linux e le istruzioni per ogni distribuzione sono un po 'diverso. La comunità Arduino ha fatto un grande lavoro di mettere insieme gruppi di istruzioni per ogni distribuzione.

Boards e porte

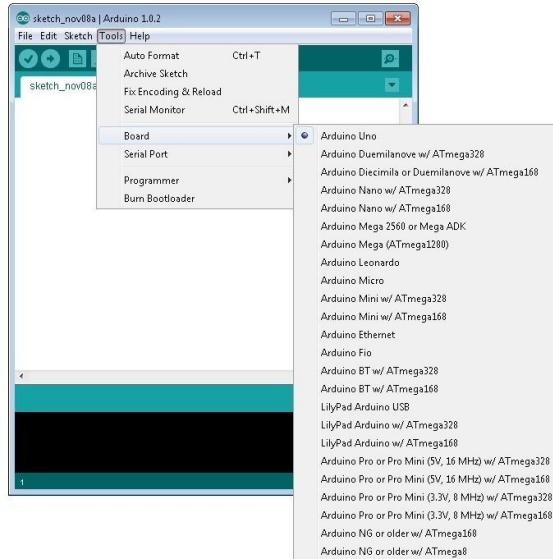
Ora siete pronti per avviare il software di Arduino, in modo da qualunque piattaforma che si sta utilizzando, aprire la cartella Arduino e apre l'applicazione di Arduino in esso contenute.



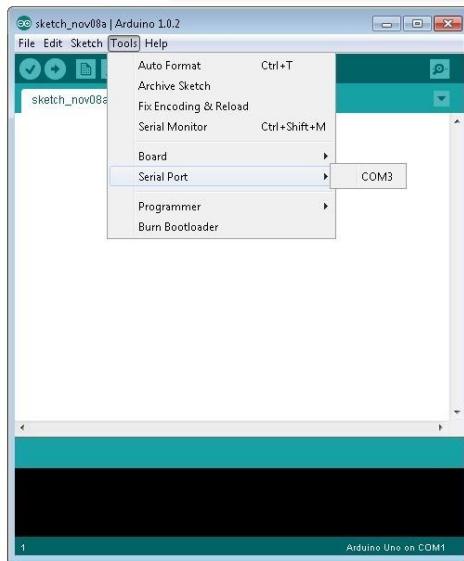


Ciò avvierà il Arduino, ma prima di poter iniziare la programmazione, si deve capire la software di Arduino quale tipo di scheda Arduino che si sta utilizzando e anche selezionare la porta a cui si connette.

A dire il IDE Arduino che tipo di scheda che si sta utilizzando. dal menu 'Strumenti', selezionare Consiglio e poi 'Arduino Uno' o 'Leonardo' a seconda dei casi.

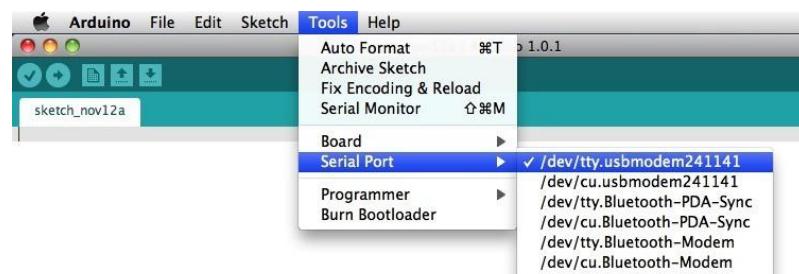


Anche sul menu 'Strumenti', troverete l'opzione 'porte seriali'. Selezionare questa opzione.



Se si utilizza Windows, ci sarà probabilmente una sola opzione qui e sarà o dire COM3 o COM4. Anche se c'è una sola opzione, sarà comunque necessario selezionarla.

Se si utilizza un Mac o Linux, ci saranno più opzioni là, ma di solito è la prima opzione nella lista, in quanto questo sarà il dispositivo più recente collegato. Questo è utile, come il nome del porto può non guardare come ha nulla a che fare con Arduino. Probabilmente sarà chiamato qualcosa come /dev/tty.usbmodemXXXX o /dev/ttyUSBn.



Lezione 1: Add librerie

Una volta che hai dimestichezza con il software di Arduino e utilizzando le funzioni built-in, si consiglia di estendere le sue funzionalità con librerie aggiuntive.

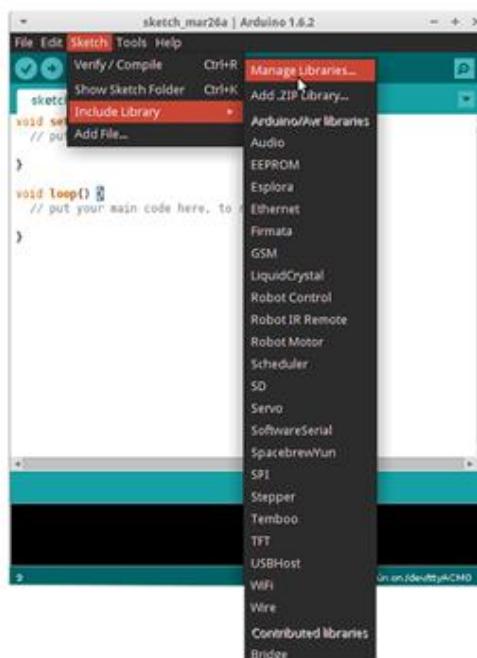
Quali sono le librerie?

Le biblioteche sono una raccolta di codice che rende facile per il collegamento ad un sensore, visualizzazione, modulo, ecc, ad esempio, la libreria incorporata LiquidCrystal rende facile parlare con display LCD carattere. Ci sono centinaia di librerie aggiuntive disponibili su Internet per il download. Le librerie integrate e alcune di queste librerie aggiuntive sono elencate nel riferimento. Per utilizzare queste librerie aggiuntive, è necessario installarli.

Come installare una libreria

Utilizzando il Library Manager

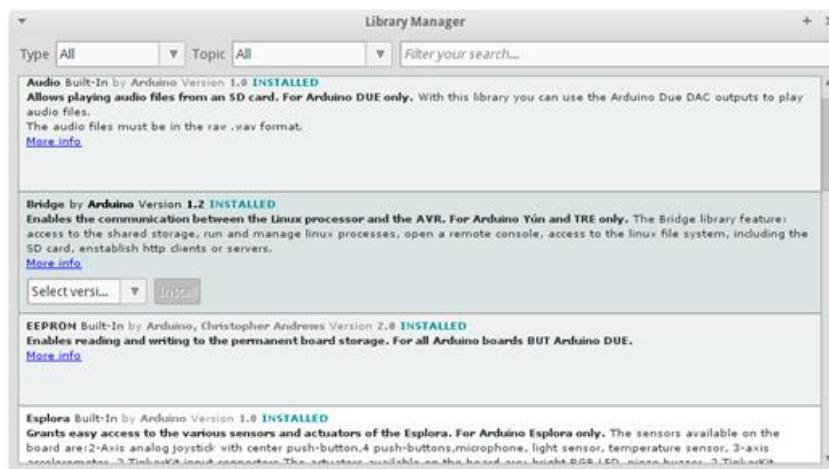
Per installare una nuova libreria nel vostro Arduino IDE, è possibile utilizzare il Gestore Library (disponibile da IDE versione 1.6.2). Aprire l'IDE e fare clic su Schizzo> Includere> Libreria> gestire librerie.



Il gestore di libreria si aprirà e troverete un elenco di librerie che sono già installati o pronti per l'installazione. In questo esempio, si installare la libreria Bridge. Scorrere l'elenco per trovarlo, quindi selezionare la versione della libreria che si desidera installare. A volte, solo una versione della libreria è disponibile. Se non viene visualizzato il menu di selezione della versione, non ti preoccupare; è normale.



Infine, fare clic su Installa e attendere l'IDE per installare la nuova libreria. Il download può richiedere del tempo a seconda della velocità di connessione. Una volta finita, un tag installata dovrebbe comparire accanto alla biblioteca Bridge. È possibile chiudere il gestore di librerie.

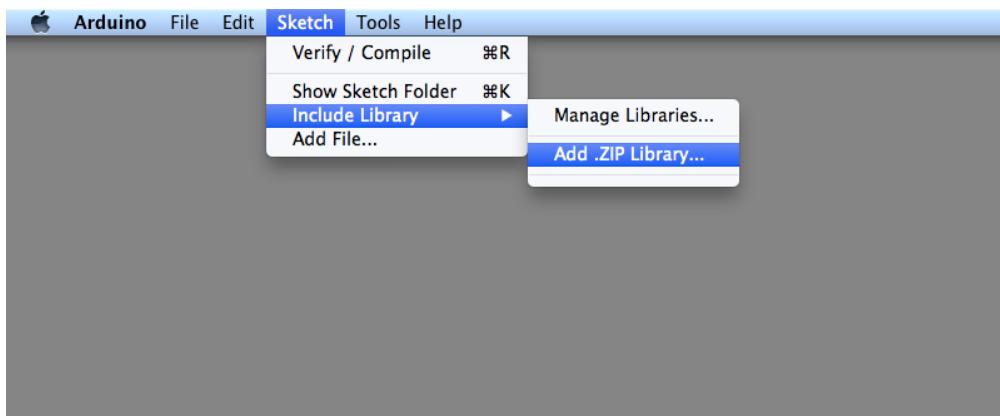


Ora è possibile trovare la nuova libreria disponibile nel menu: Biblioteca. Se si desidera aggiungere la propria libreria, aprire una nuova emissione su GitHub.

Importazione di una libreria .zip

Le biblioteche sono spesso distribuiti come file ZIP o una cartella. Il nome della cartella è il nome della libreria. All'interno della cartella sarà il seguente: file cpp, .h di file, spesso un file keywords.txt, la cartella esempi, e altri file richiesti dalla libreria. A partire dalla versione 1.0.5, è possibile installare librerie di terze parti nell'IDE. Non decomprimere la libreria scaricato; lasciarla così com'è.

Nel Arduino IDE, passare a schizzo> Biblioteca> Aggiungi .ZIP libreria



Verrà richiesto di selezionare la libreria che si desidera aggiungere. Passare alla posizione del file .zip e aprirlo.

Tornare al menu Sketch> Importa libreria. Ora dovreste vedere la biblioteca in fondo al menu a discesa. È pronto per essere utilizzato nel vostro disegno. Il file zip sarà stato ampliato nella cartella librerie nella Arduino abbozza directory.

NB: La libreria sarà disponibile da utilizzare in schizzi, ma gli esempi per la libreria non verrà mostrato in File> Esempi fino a dopo l'IDE è stato riavviato.

Installazione manuale

Per installare la libreria, prima, chiudere l'applicazione Arduino. Poi decomprimere il file ZIP contenente la libreria. Ad esempio, se si sta installando una libreria chiamata "ArduinoParty", decomprimere ArduinoParty.zip. Dovrebbe contenere una cartella denominata ArduinoParty, con i file come ArduinoParty.cpp e ArduinoParty.h all'interno. (Se i file cpp e .h non sono in una cartella, è necessario crearne uno. In questo caso, devi fare una cartella chiamata "ArduinoParty" e spostare in esso tutti i file che erano nel ZIP presentare, come ArduinoParty.cpp e ArduinoParty.h.)

Trascinare la cartella ArduinoParty in questa cartella (cartella librerie). In Windows, sarà probabilmente chiamato "My Documents \ Arduino \ libraries". Per gli utenti Mac, sarà probabilmente chiamato "Documenti / Arduino / librerie". Su Linux, sarà la cartella "biblioteche" nel vostro album da disegno.

La cartella biblioteca Arduino dovrebbe apparire così (su Windows):

My Documents\Arduino\libraries\ArduinoParty\ArduinoParty.cpp

My Documents\Arduino\libraries\ArduinoParty\ArduinoParty.h

My Documents\Arduino\libraries\ArduinoParty\examples

....

o come questo (su Mac e Linux):

Documents/Arduino/libraries/ArduinoParty/ArduinoParty.cpp

Documents/Arduino/libraries/ArduinoParty/ArduinoParty.h

Documents/Arduino/libraries/ArduinoParty/examples

....

Ci possono essere più file che solo il .cpp e .h file in modo da assicurarsi che siano tutti lì. (La libreria non funziona se si mette il .cpp e .h file direttamente nelle librerie cartella o se sono annidati in una cartella in più. Ad esempio: Documents \ Arduino \ Librerie \ ArduinoParty.cpp e Documents \ Arduino \ le librerie \ ArduinoParty \ ArduinoParty \ ArduinoParty.cpp non funzioneranno.)

Riavviare l'applicazione Arduino. Assicurarsi che la nuova libreria appare nel menu Sketch> Importa libreria. Tutto qui. Hai installato una libreria!

Sommario

In questa lezione, installeremo tutte le librerie che verranno usate nel tutorial. Aprire la cartella librerie e installare i file ZIP uno per uno in modo da non avere bisogno di fare questo passo nei seguenti lezioni. Abbiamo appena collegare il componente come lo schema e caricare il codice fornito. Poi il kit lavorerà.

Lezione 2: Lampeggio

Generali

In questa lezione, si apprenderà come programma la vostra scheda di controllo UNO R3 per rendere il Arduino incorporato lampeggiare LED.

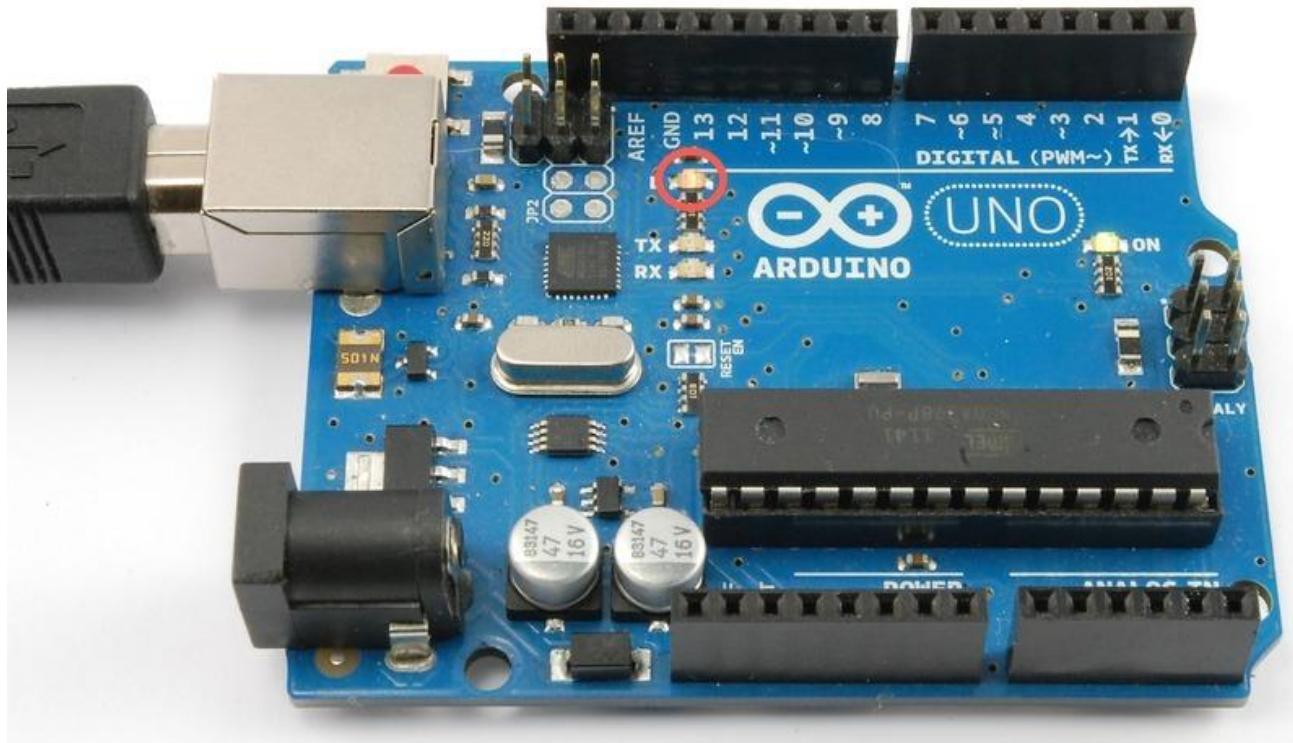
Componente Richiesto

(1) Elegoo UNO R3

Principio

La scheda UNO R3 ha righe di connettori lungo entrambi i lati che vengono utilizzati per la connessione a vari dispositivi elettronici e plug-in "scudi" che estende la sua capacità.

Essa ha anche un singolo LED che è possibile controllare da schizzi. Questo LED è costruita sul bordo UNO R3 ed è spesso indicato come il 'L' LED come questo è come è etichettato sulla scheda.



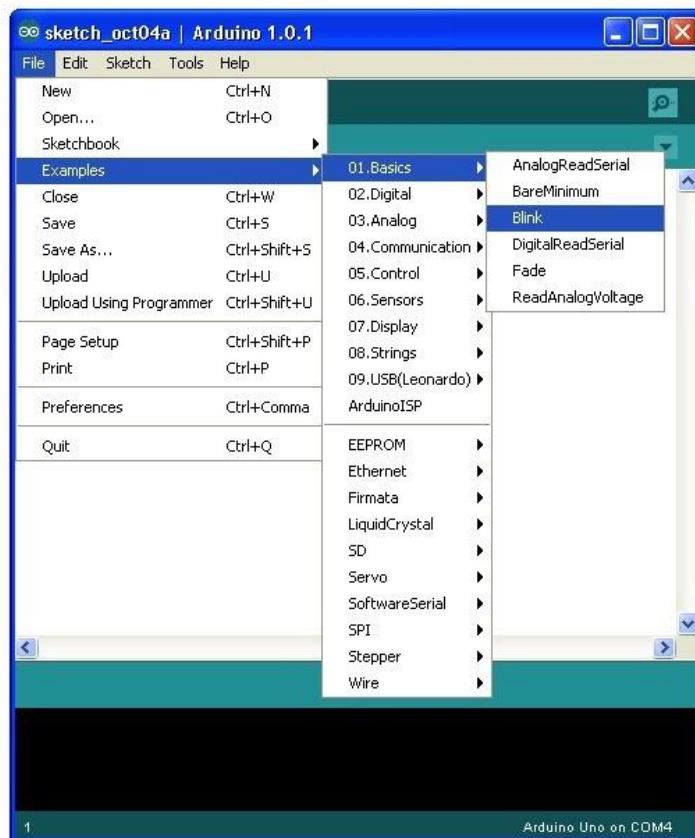
Si possono trovare che la vostra scheda UNO R3 's' L 'ha portato già lampeggiare quando viene collegato ad una presa USB. Questo perché le schede sono in genere forniti con il 'Blink' schizzo pre-installato.

In questa lezione, si riprogramma la scheda UNO R3 con il nostro schizzo di lampeggio e quindi modificare la velocità con cui lampeggia.

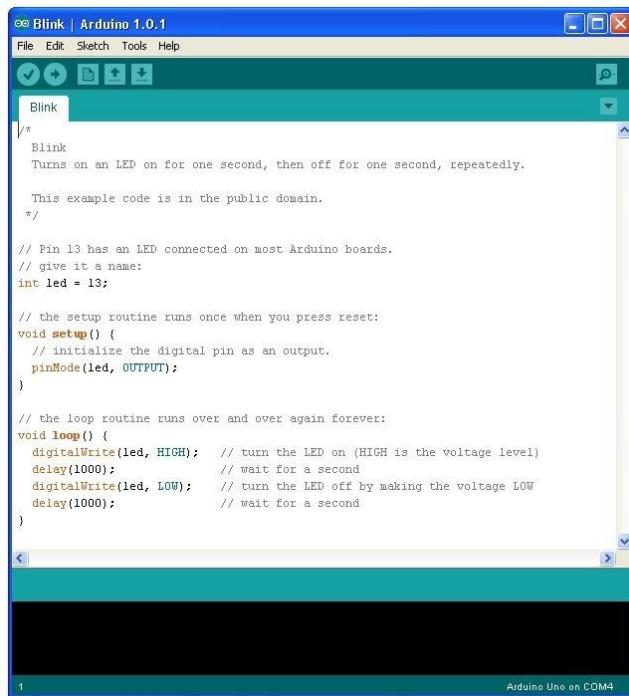
Nella lezione 0, si imposta il Arduino IDE e fatto in modo che si potrebbe trovare la porta seriale giusta per poter collegare al vostro bordo UNO R3. Il momento è venuto di mettere che il collegamento alla prova e programmare il bordoUNO R3.

L'IDE di Arduino include una grande collezione di esempi di schizzi che è possibile caricare e utilizzare. Questo include lampeggio un esempio schizzo per rendere la 'L' LED.

Caricare lo schizzo 'di lampeggio' che troverete nel sistema di menu del IDE in File> Esempi> 01.Base



Quando si apre la finestra di disegno, ingrandirla in modo che si può vedere l'intero disegno nella finestra.



L'esempio schizzi inclusi con il Arduino IDE sono 'sola lettura'. Cioè, è possibile caricare su una scheda UNO R3, ma se li si cambia, non è possibile salvarle come lo stesso file.

Dal momento che ci accingiamo a cambiare questo disegno, la prima cosa che devi fare è salvare la propria copia.

Dal menu File sul Arduino IDE, selezionare 'Salva con nome ..' e quindi salvare il disegno con il nome di 'Nostra Lampeggia'.

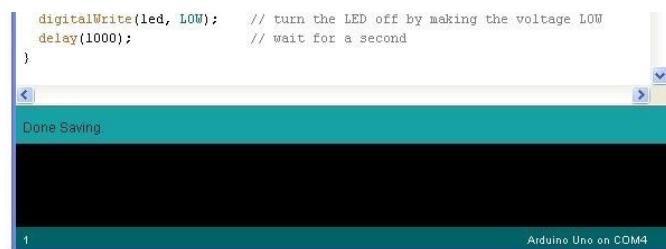


Hai salvato la vostra copia di 'Blink' nel vostro album da disegno. Questo significa che se si volesse trovare di nuovo, si può solo aprirlo usando il File> opzione di menu di Album di schizzi.



Allegare la tua scheda Arduino al computer con il cavo USB e verificare che il 'Consiglio Tipo' e 'Porta seriale' siano impostate correttamente. Potrebbe essere necessario fare riferimento alla Lezione 0.

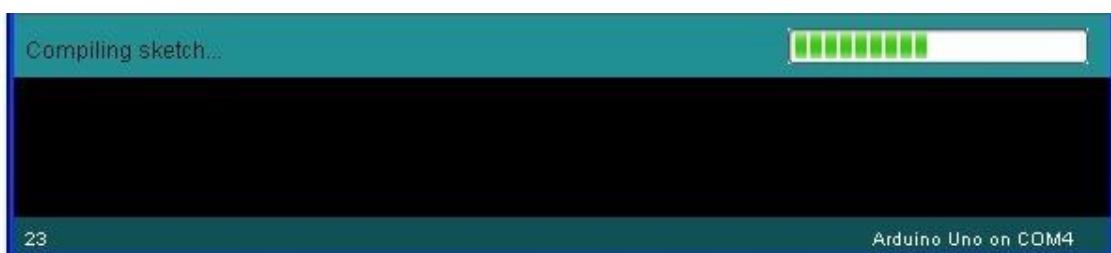
La Arduino IDE vi mostrerà le impostazioni correnti per scheda nella parte inferiore della finestra.



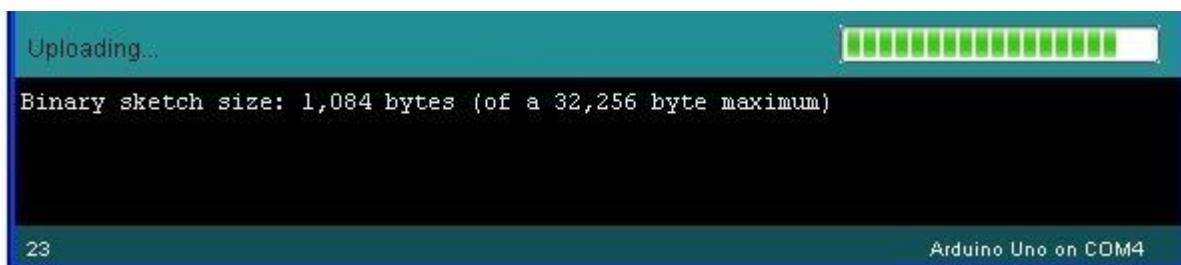
Fare clic sul pulsante 'Carica'. Che è il secondo pulsante da sinistra nella barra degli strumenti.



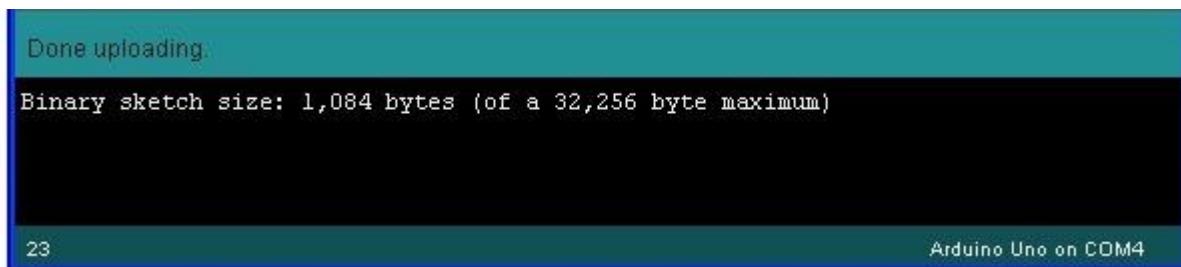
Se si guarda l'area di stato dell'IDE, si vedrà una barra di avanzamento e una serie di messaggi. In un primo tempo, si dirà 'Compilazione Sketch ...'. Questo trasforma il disegno in un formato adatto per il caricamento della scheda.



Successivamente, lo stato passerà a 'Caricamento'. A questo punto, il LED sul Arduino dovrebbe iniziare a lampeggiare come lo schizzo viene trasferito.



Infine, lo stato passerà a 'Fatto il caricamento'. Viene visualizzato un messaggio che indica che il disegno sta usando 1,084 byte dei 32,256 byte disponibili ..



Si prega di notare che durante la 'Compilazione Sketch ...' fase, è possibile ottenere il seguente messaggio di errore:



Essa può significare che la scheda non è collegata a tutti, o non sono stati installati i driver (se necessario) o che sia selezionata la porta seriale sbagliata.

Se si verifica questo, tornare alla Lezione 0 e verificare l'installazione.

Al termine del caricamento è stato completato, il consiglio di amministrazione dovrebbe riavviare e iniziano a lampeggiare.

Aprire il codice

Di notare che una parte enorme di questo disegno è composto da commenti. Queste non sono le istruzioni dei programmi reali; piuttosto, semplicemente spiegare come funziona il programma. Sono lì per il vostro vantaggio.

Tutto tra / * e * / alla parte superiore del disegno è un blocco di commento; spiega quali il disegno è per.

Linea singola I commenti iniziano con // e tutto ciò che fino alla fine di quella linea è considerato un commento.

La prima linea di codice è:

Codice della copia

1. int led = 13;

Come il commento di cui sopra spiega, questo è dare un nome al perno che il LED è collegato. Si tratta di 13 sulla maggior parte dei Arduino, tra cui l'UNO e Leonardo.

Successivamente, abbiamo la funzione di 'setup'. Di nuovo, come il commento dice, questa viene eseguita quando viene premuto il pulsante di ripristino. Viene eseguito anche ogni volta che il ripristino della scheda, per qualsiasi motivo, come l'energia prima di essere applicato ad esso, o dopo uno schizzo è stato caricato.

Codice della copia

1. Imposta invalida() {
2. // inizializzare il perno digitale come uscita.
3. Modalità di perno(led, OUTPUT);
4. }

Ogni Arduino schizzo deve avere una funzione di 'setup', e il luogo dove si potrebbe desiderare di aggiungere istruzioni del proprio è tra il {e}.

In questo caso, vi è un solo comando lì, che, come il commento stati racconta la scheda Arduino che ci accingiamo a utilizzare il pin LED come uscita.

E 'anche obbligatorio per uno schizzo di avere una funzione di' loop '. A differenza della funzione di 'setup' che viene eseguito solo una volta, dopo un reset, la funzione di 'loop' sarà, dopo aver terminato l'esecuzione dei suoi comandi, ripartire immediatamente.

Codice della copia

1. Loop valido() {
2. digitalWrite (LED, ad alta); // Accendere il LED (AD ALTA è il livello di tensione)
3. ritardo (1000); // Attendere un secondo
4. digitalWrite (led, LOW); // Spegnere il LED, rendendo la bassa tensione

5. ritardo (1000); // Attendere un secondo

6. }

All'interno della funzione di loop, i comandi prima di tutto ruotare il perno LED (AD ALTA), poi 'ritardo' di 1000 millisecondi (1 secondo), poi girare il LED pin fuori e mettere in pausa per un altro secondo.

Si è ora intenzione di rendere il vostro LED lampeggi più velocemente. Come si può immaginare, la chiave di questo sta nel cambiare il parametro () per il comando 'ritardo'.

```
// the loop routine runs over and over again forever:  
void loop() {  
    digitalWrite(led, HIGH);    // turn the LED on (HIGH is the voltage level)  
    delay(1000);              // wait for a second  
    digitalWrite(led, LOW);    // turn the LED off by making the voltage LOW  
    delay(1000);              // wait for a second  
}
```

Questo periodo di ritardo è espresso in millisecondi, quindi se si desidera che il led lampeggi due volte più veloce, modificare il valore da 1000 a 500. Questo sarebbe poi una pausa di mezzo secondo ogni ritardo, piuttosto che un intero secondo.

Carica di nuovo disegno e si dovrebbe vedere l'inizio LED a lampeggiare più rapidamente.

Lezione 3: LED

Generali

In questa lezione imparerai come cambiare la luminosità di un LED usando diversi valori di resistenza.

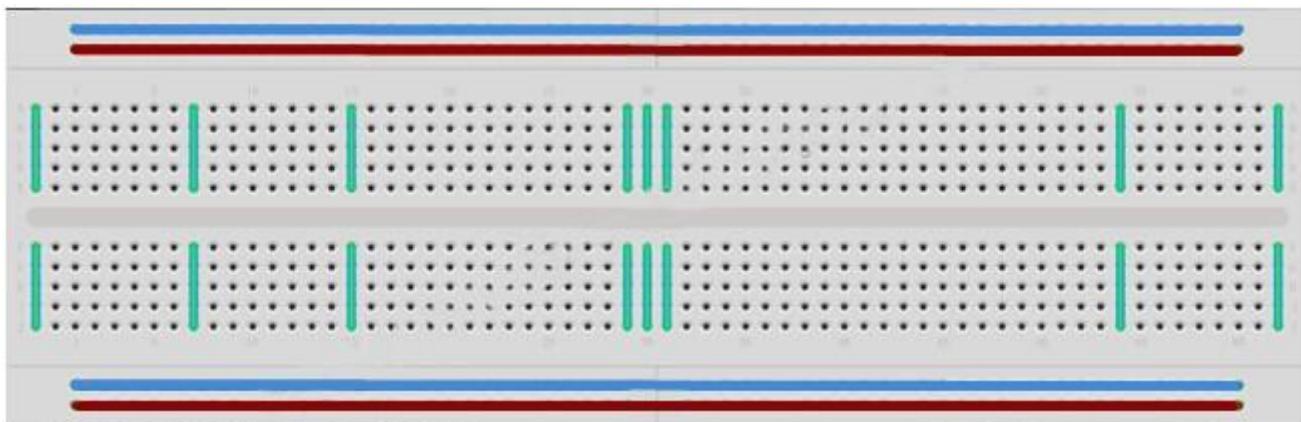
Componenti necessari

- (1) Elegoo UNO R3
- (1) LED rosso 5 millimetri
- (1) 220 ohm resistore
- (1) 1k ohm resistore
- (1) 10k ohm resistore
- (2) M-M fili

Componente di Introduzione

BREADBOARD MB-102:

Una basetta consente ai circuiti prototipi rapidamente, senza dover saldare le connessioni. Di seguito è riportato un esempio.



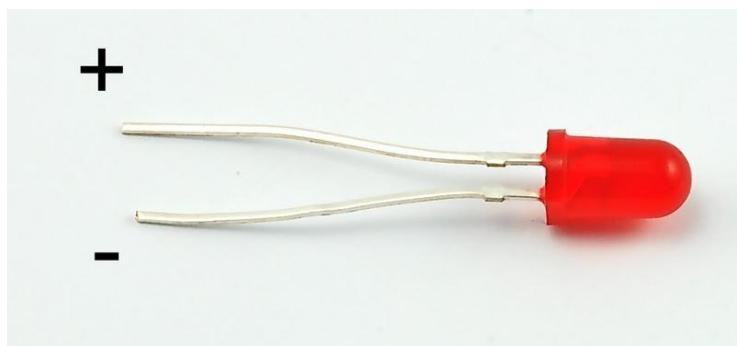
Breadboards disponibili in varie dimensioni e configurazioni. Il tipo più semplice è una griglia di fori in un blocco di plastica. All'interno sono strisce di metallo che forniscono il collegamento elettrico tra i fori nelle righe più brevi. Spingendo le gambe di due componenti differenti nella stessa riga li unisce elettricamente. Un canale profondo esecuzione nel mezzo indica che c'è un'interruzione nei collegamenti lì, cioè, è possibile spingere un chip con le gambe ai lati del canale senza collegare insieme. Alcuni basette hanno due strisce di fori che corrono lungo i bordi lunghi della scheda che sono separate dalla rete principale. Questi hanno strisce esegue la lunghezza della tavola all'interno e fornire un modo per collegare una tensione comune. Essi sono di solito in coppia per 5 volti e terreno. Queste strisce sono indicati come le rotaie e che consentono di collegare l'alimentazione di molti componenti o punti nella scheda.

Mentre basette sono grandi per la prototipazione, hanno alcune limitazioni. Poiché i collegamenti sono innesto e temporaneo, non sono affidabili come connessioni saldate. Se si hanno problemi intermittenti con un circuito, potrebbe essere dovuto ad una cattiva connessione su un tagliere.

LED:

LED fare grandi spie. Usano pochissima elettricità e hanno praticamente durano per sempre. In questa lezione, si utilizzerà forse il più comune di tutti i LED: un LED rosso 5 mm. 5 mm si riferisce al diametro del LED. Altri formati comuni sono 3 mm e 10 mm.

Non è possibile collegare direttamente un LED ad una fonte di batteria o di tensione perché 1) il LED ha una positiva e una cavo negativo e non si accende se posta nel modo sbagliato e 2) un LED deve essere utilizzato con una resistenza per limitare o 'choke' la quantità di corrente che fluisce attraverso di essa; in caso contrario, si brucerà!



Se non si utilizza un resistore con un LED, allora può essere distrutto quasi immediatamente, come troppo corrente passerà, riscaldandolo e distruggendo la 'giunzione' dove si produce la luce.

Ci sono due modi per dire che è il cavo positivo del LED e che il negativo.

In primo luogo, il conduttore positivo è più lungo.

In secondo luogo, se il cavo negativo entra nel corpo del LED, vi è un bordo piatto per il caso del LED.

Se vi capita di avere un LED che ha un lato piatto accanto al conduttore più a lungo, si deve presumere che il piombo è più positivo.

RESISTENZE:

Come suggerisce il nome, resistenze resistono il flusso di elettricità. Più alto è il valore della resistenza, più resistente e la corrente elettrica meno fluirà attraverso di essa. Abbiamo intenzione di usare questo per controllare la quantità di energia elettrica fluisce attraverso il LED e, quindi, come brillantemente che brilla.

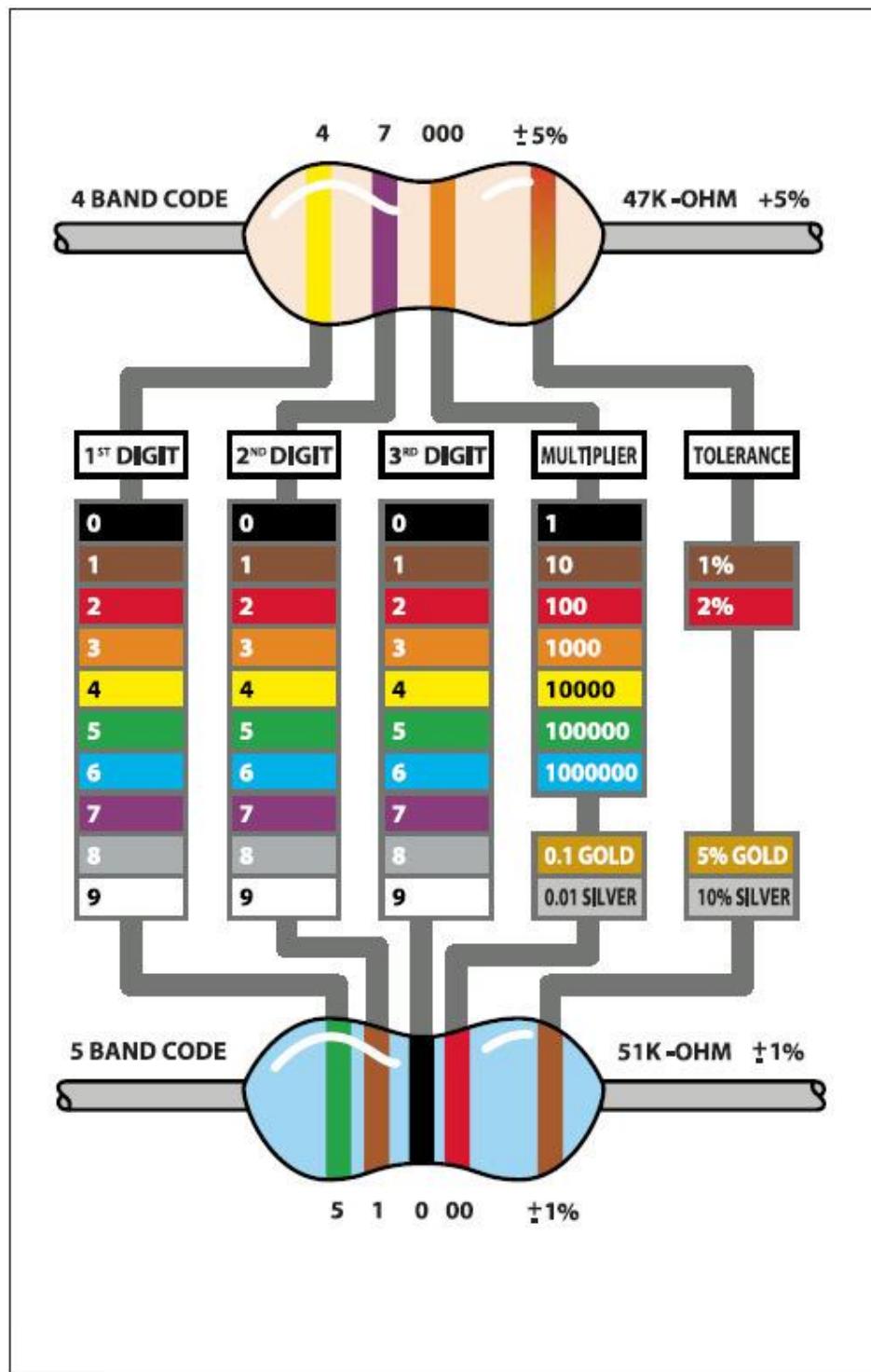


Ma prima, più resistenze ...

L'unità della resistenza è chiamata Ohm, che di solito è abbreviato in Ohm lettera greca Omega. Perché un Ohm è un basso valore di resistenza (che non resiste molto a tutti), anche noi indichiamo i valori dei resistori a $k\Omega$ ($1,000\ \Omega$) e MW ($1,000,000\ \Omega$). Questi sono chiamati kilo-ohm e mega-ohm.

In questa lezione, abbiamo intenzione di utilizzare tre differenti valori di resistenza: $220\ \Omega$, $1k$ e $10k$. Queste resistenze hanno lo stesso aspetto, salvo che essi hanno diverse strisce colorate su di loro. Queste strisce si dice che il valore della resistenza.

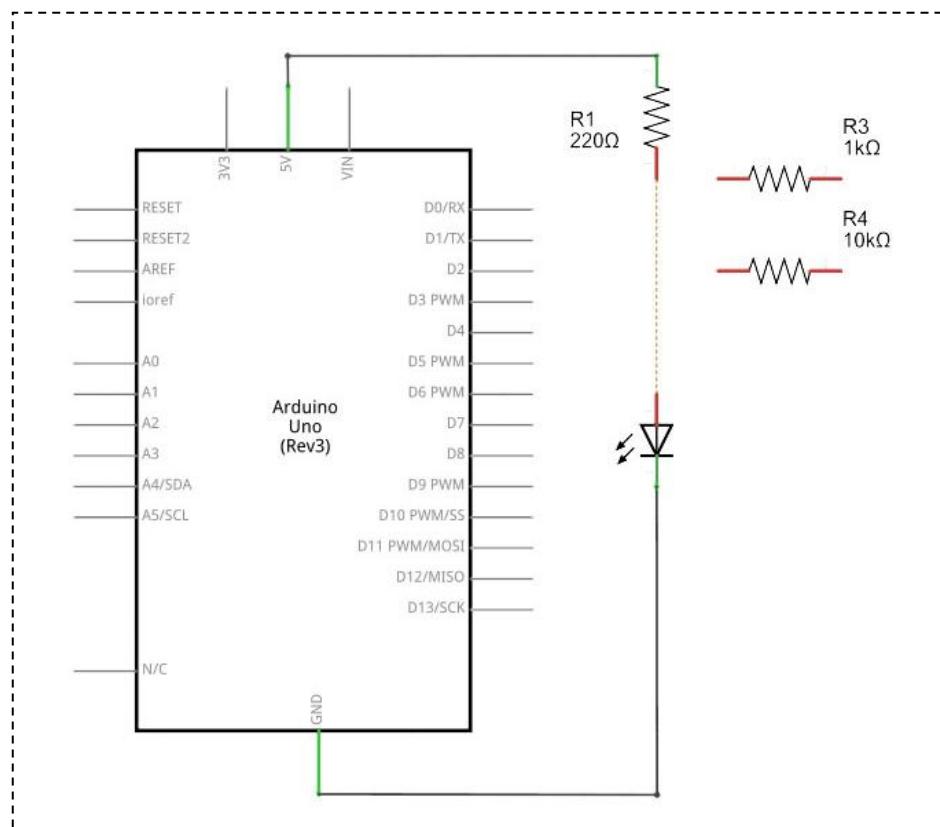
Il codice colore resistore ha tre strisce colorate e poi una striscia oro ad una estremità.



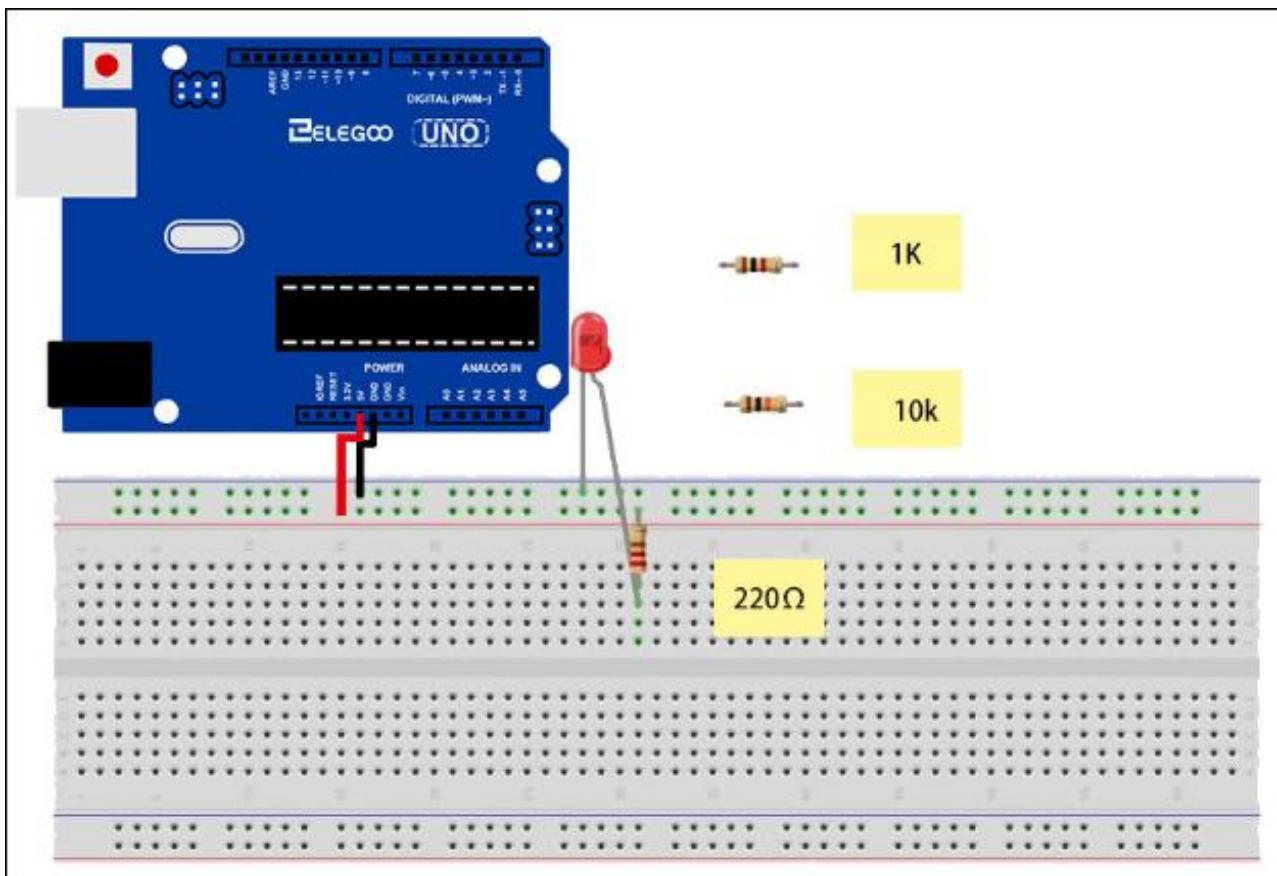
Diversamente LED, resistori non hanno un conduttore positivo e negativo. Possono essere collegati in entrambi i casi intorno.

Connessione

schematico



Schema di collegamento

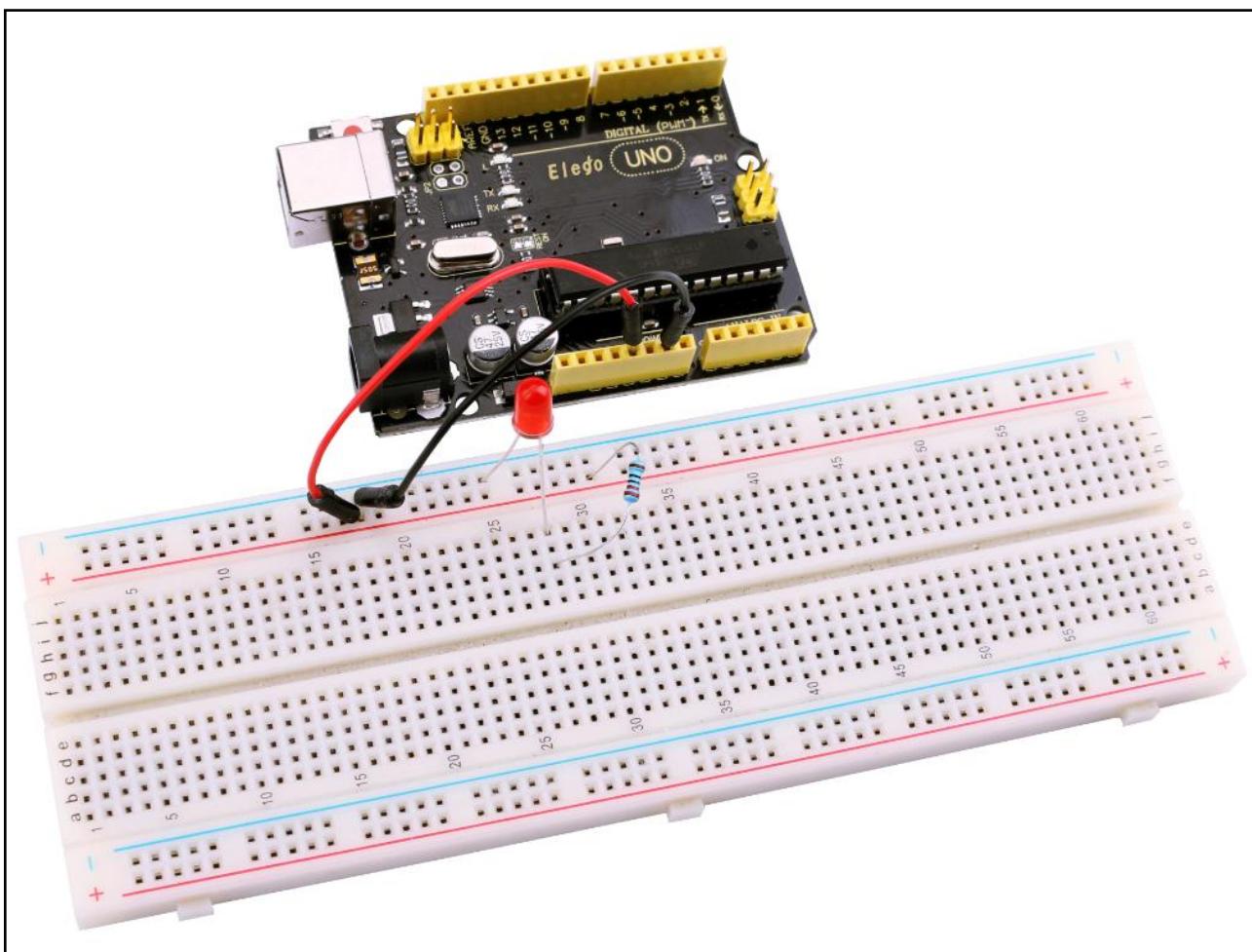


L'UNO è una comoda fonte di 5 volte, che useremo per fornire energia a LED e la resistenza. Non c'è bisogno di fare qualsiasi cosa con il vostro UNO, se non per collegarlo a un cavo USB. Con la resistenza da $220\ \Omega$ a posto, il LED dovrebbe essere abbastanza luminoso. Se si scambia la resistenza da $220\ \Omega$ per la resistenza 1k, il LED apparirà un po 'dimmer. Infine, con la resistenza 10 kW a posto, il LED sarà quasi invisibile. Tirare il cavo ponticello rosso fuori dalla basetta e toccare nel foro e rimuoverlo, in modo che si comporta come un interruttore. Si dovrebbe solo essere in grado di notare la differenza.

Al momento, è 5V visitando una gamba della resistenza, l'altra gamba del resistore andare al lato positivo del LED e l'altro lato del LED andare a GND. Tuttavia, se abbiamo spostato la resistenza in modo che è venuto dopo che il LED, come mostrato di seguito, il LED si illumina comunque.

Probabilmente si vuole mettere la resistenza di $220\ \Omega$ al suo posto.

Non importa quale lato del LED mettiamo la resistenza, finché c'è qualche parte.



Lezione 4: RGB LED

Generali

RGB LEDs sono un modo divertente e facile per aggiungere un pò di colore ai vostri progetti. Dal momento che sono come 3 LED regolari in uno, come utilizzare e collegarli non è molto diverso.

Essi sono per lo più in 2 versioni: Anodo comune o catodo comune.

Anodo comune utilizza 5V sul pin comune, mentre catodo comune collega a terra.

Come per ogni LED, abbiamo bisogno di collegare alcune resistenze in linea (3 in totale) in modo da poter limitare la corrente in fase di elaborazione.

Nel nostro schizzo, inizieremo con il LED di stato di colore rosso, poi sfumare a Green, poi dissolvenza Blue ed infine di nuovo al colore rosso. In questo modo ci sarà il ciclo attraverso la maggior parte del colore che può essere raggiunto.

Componenti necessari

- (1) Elegoo UNO R3
- (1) Bread board
- (4) M-M fili
- (1) RGB LED
- (3) 220 ohm resistori

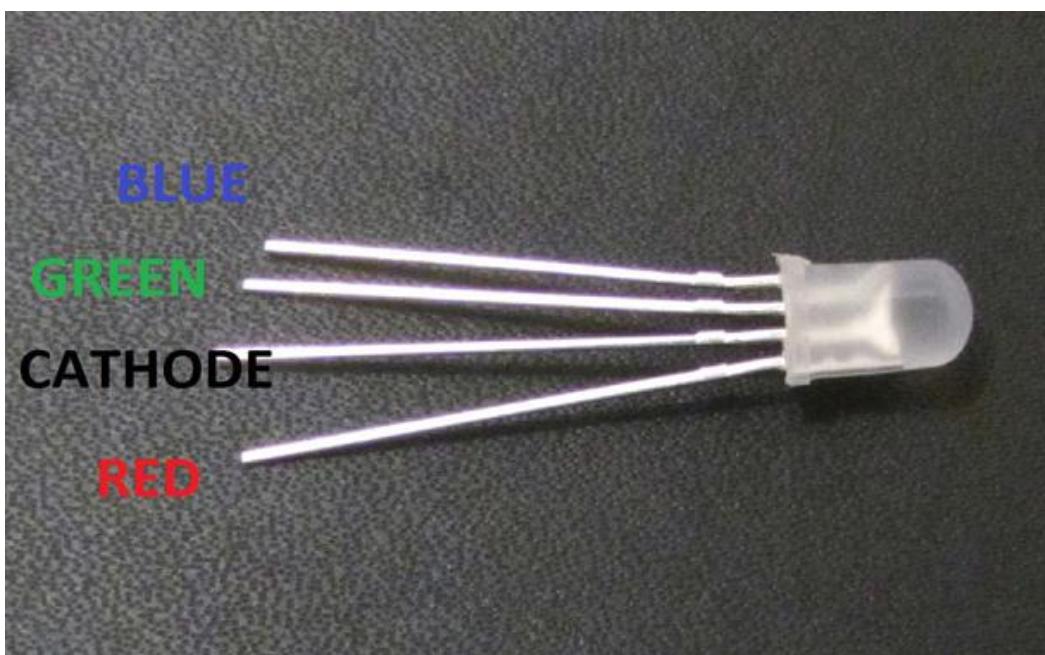
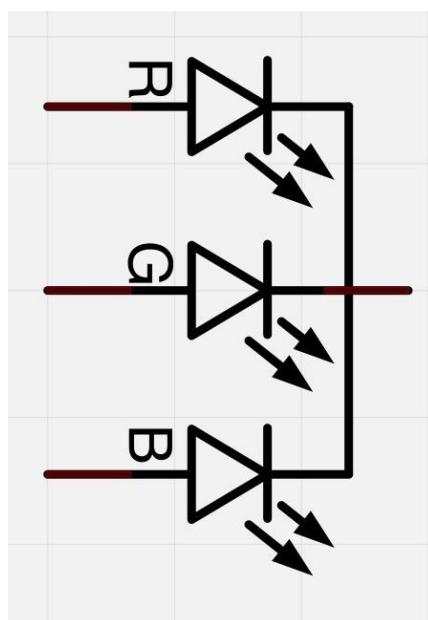
Componente di Introduzione

RGB:

Ad un primo sguardo, RGB (rosso, verde, blu) i LED sono proprio come i LED normali. Invece, all'interno del consueto pacchetto di LED, ci sono in realtà tre LED, uno rosso, uno verde e sì, uno blu. Controllando la luminosità di ciascuno dei singoli LED potete mescolare praticamente qualsiasi colore che si desidera.

Mescoliamo i colori allo stesso modo in cui si mescolare vernice su una tavolozza - regolando la luminosità di ciascuno dei LED a tre. Il modo più difficile fare questo sarebbe quello di utilizzare diverse resistori di valore (o resistenze variabili) come abbiamo fatto con la Lezione 2, ma questo è un sacco di lavoro! Fortunatamente per noi, il bordo ONU R3 dispone di una funzione analogWrite che è possibile utilizzare con i perni contrassegnate con un ~ per produrre una quantità variabile di potere al LED appropriata.

Il LED RGB ha quattro conduttori. C'è un conduttore andare al collegamento positivo di ciascuno dei LED all'interno del pacchetto e un unico cavo che è collegato a tutti i tre lati negativi del LED singolo.



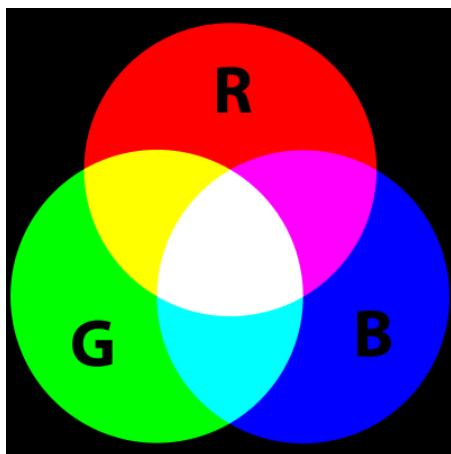
Il collegamento comune negativo del pacchetto LED è il secondo perno dal lato piatto. È inoltre il più lungo dei quattro conduttori e verrà collegato a terra.

LED all'interno del pacchetto Ciascuno richiede la propria resistenza 270Ω evitare troppa corrente che fluisce attraverso molto. I tre conduttori positivi dei LED (uno rosso, uno verde e uno blu) sono collegati ai perni di uscita UNO utilizzano questi resistori.

COLORI:

La ragione per cui si può mescolare tutto il colore che gradite variando le quantità di luce rossa, verde e blu è che l'occhio ha tre tipi di recettore della luce in esso (rosso, verde e blu). L'occhio e il cervello processa la quantità di rosso, verde e blu e convertirlo in un colore dello spettro.

In un certo senso, utilizzando i tre LED, stiamo giocando un trucco sugli occhi. Questa stessa idea è usata nei televisori, dove il LCD ha rosso, verde e blu di punti di colore accanto all'altro che compongono ciascun pixel.



Se si imposta la luminosità di tutti i tre LED essere la stessa, allora il colore generale della luce sarà bianco. Se spegniamo il LED blu, in modo che solo i LED rosso e verde sono la stessa luminosità, poi la luce apparirà di colore giallo.

Possiamo controllare la luminosità di ciascuna delle parti rosso, verde e blu del LED separatamente, rendendo possibile mescolare qualsiasi colore vogliamo.

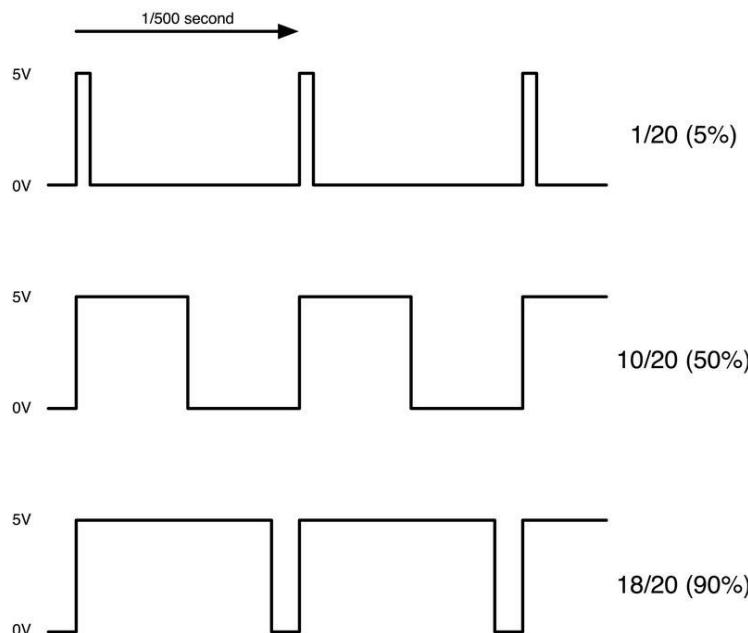
Nero non è tanto un colore come assenza di luce. Pertanto, il più vicino si può venire a nero con il nostro LED è di spegnere tutti i tre colori.

PWM Teoria

Ad impulsi larghezza di della modulazione (PWM) è una tecnica di controllo di potenza.

Usiamo anche qui per controllare la luminosità di ciascun LED.

Lo schema seguente mostra il segnale da uno dei perni PWM sul UNO.



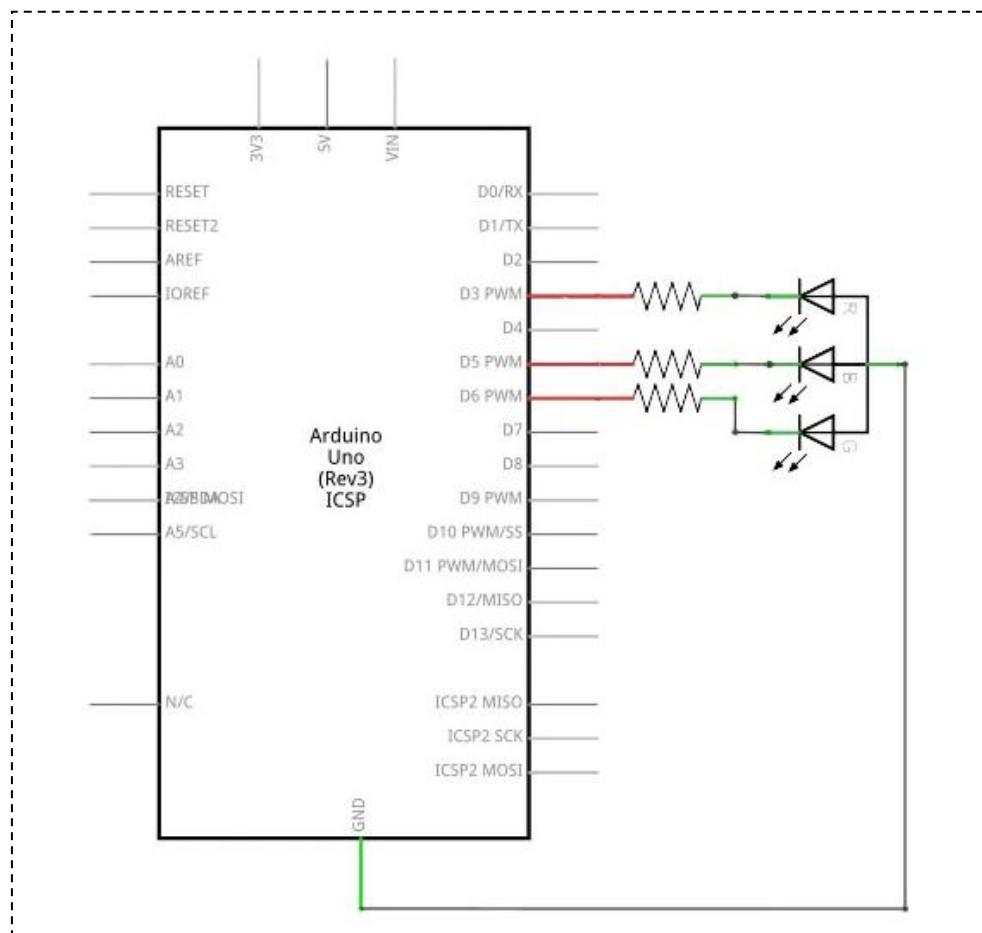
Circa ogni 1/500 di secondo, l'uscita PWM produrrà un impulso. La durata di questo impulso è controllata dalla funzione 'analogWrite'. Così 'analogWrite (0)' non produrrà alcun impulso a tutti e 'analogWrite (255)' produrrà un impulso che dura completamente fino al successivo impulso è dovuto, in modo che l'uscita è in realtà su tutto il tempo.

Se si specifica un valore nel analogWrite che è da qualche parte tra 0 e 255, quindi produrremo un impulso. Se l'impulso di uscita è alta solo il 5% del tempo, allora qualsiasi stiamo guidando ricevere solo il 5% della piena potenza.

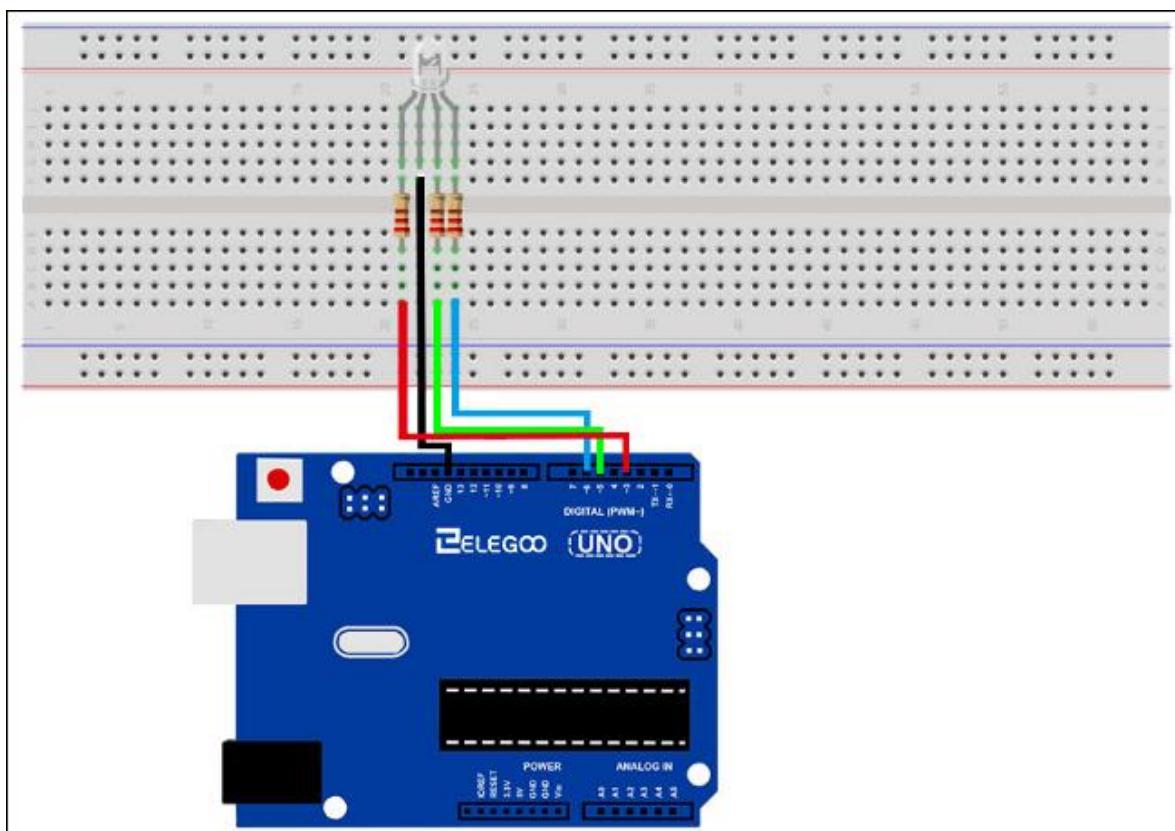
Se, tuttavia, l'uscita è a 5V per il 90% del tempo, allora il carico otterrà il 90% della potenza erogata ad esso. Non possiamo vedere i LED di accensione e spegnimento a quella velocità, così a noi, sembra proprio come la luminosità sta cambiando.

Connessione

Schematico



Schema di collegamento



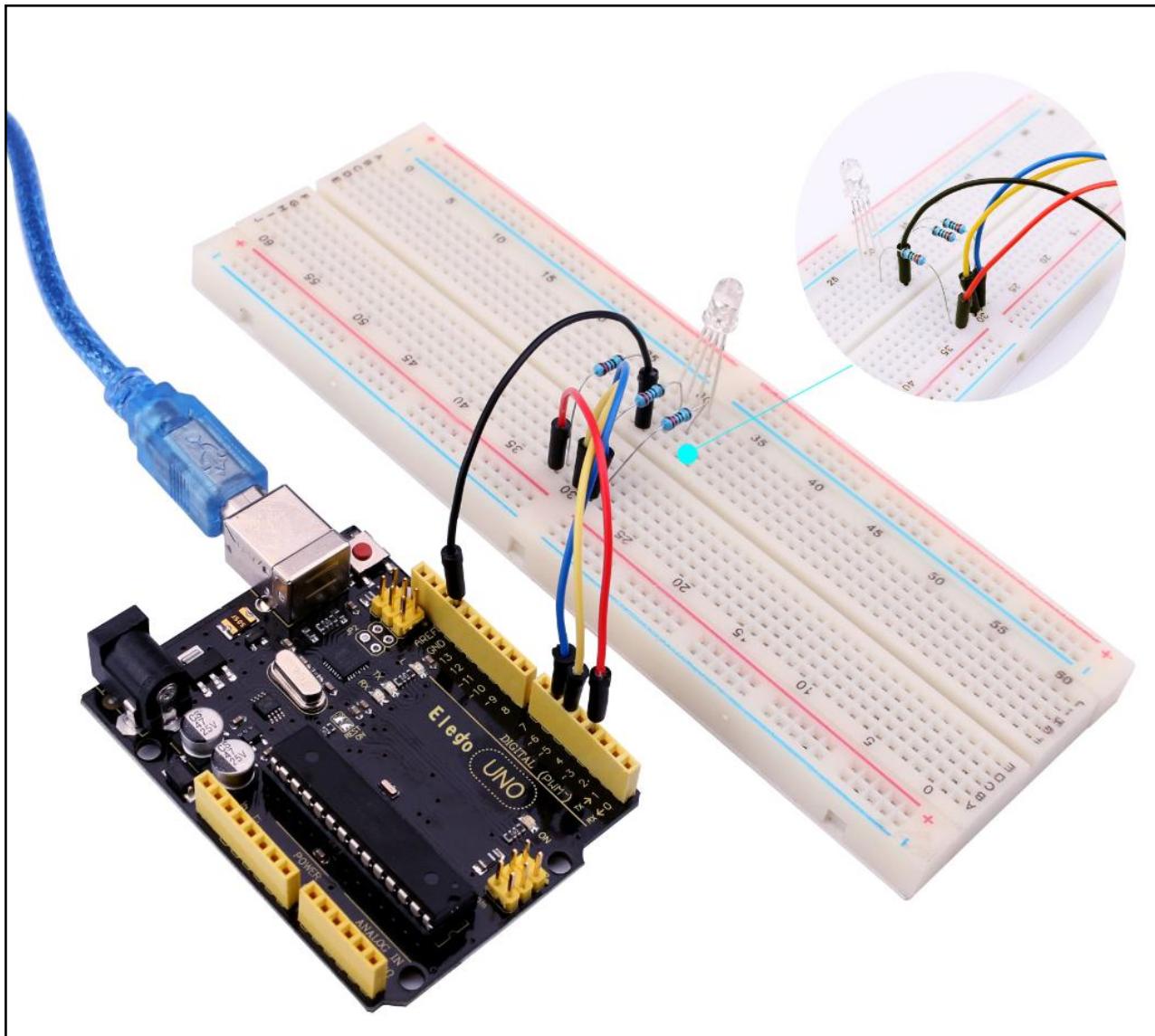
Codice

Il nostro codice potrà essere utilizzato per i cicli per scorrere i colori.

Il primo ciclo FOR saranno destinati dal rosso al verde.

Il secondo ciclo FOR andrà dal verde al blu.

L'ultimo ciclo FOR saranno destinati dal blu al rosso.



Lezione 5: Ingressi Digitali

Generali

In questa lezione, si impara a usare pulsanti con gli ingressi digitali per girare un LED acceso e spento.

Premendo il pulsante più vicino alla parte superiore della basetta si accende l'acceso a LED; premendo l'altro pulsante si accende il led spento.

Componenti necessari

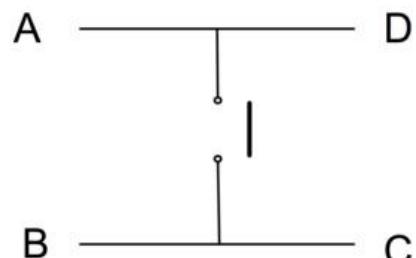
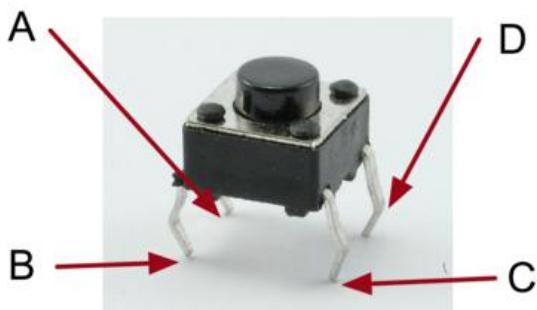
- (1) Elegoo UNO R3
- (1) Breadboard
- (1) 5mm rosso LED
- (1) 220 ohm resistore
- (2) Interruttori a pulsante
- (6) M-M fili

Introduzione di componente

INTERRUTTORI A PULSANTE:

Gli interruttori siano componenti molto semplici. Quando si preme un pulsante o lanciate una leva, si collegano due contatti insieme in modo che l'elettricità possa scorrere attraverso di loro.

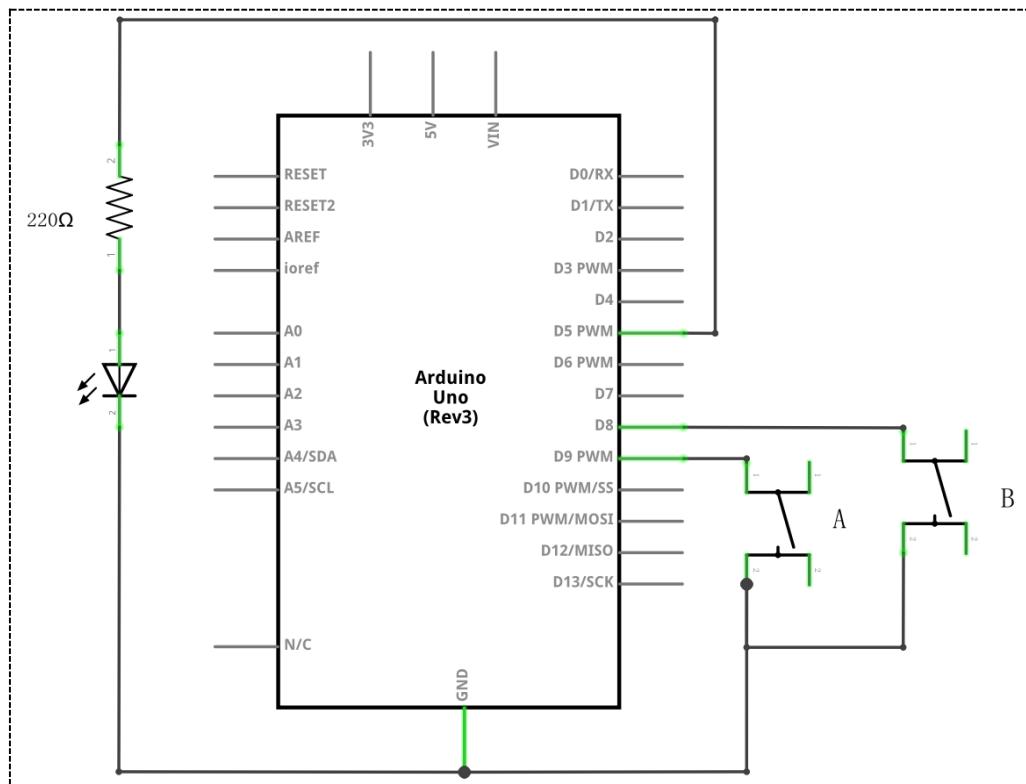
I piccoli interruttori tattili che vengono utilizzati in questa lezione hanno quattro connessioni, che può essere un po 'di confusione.



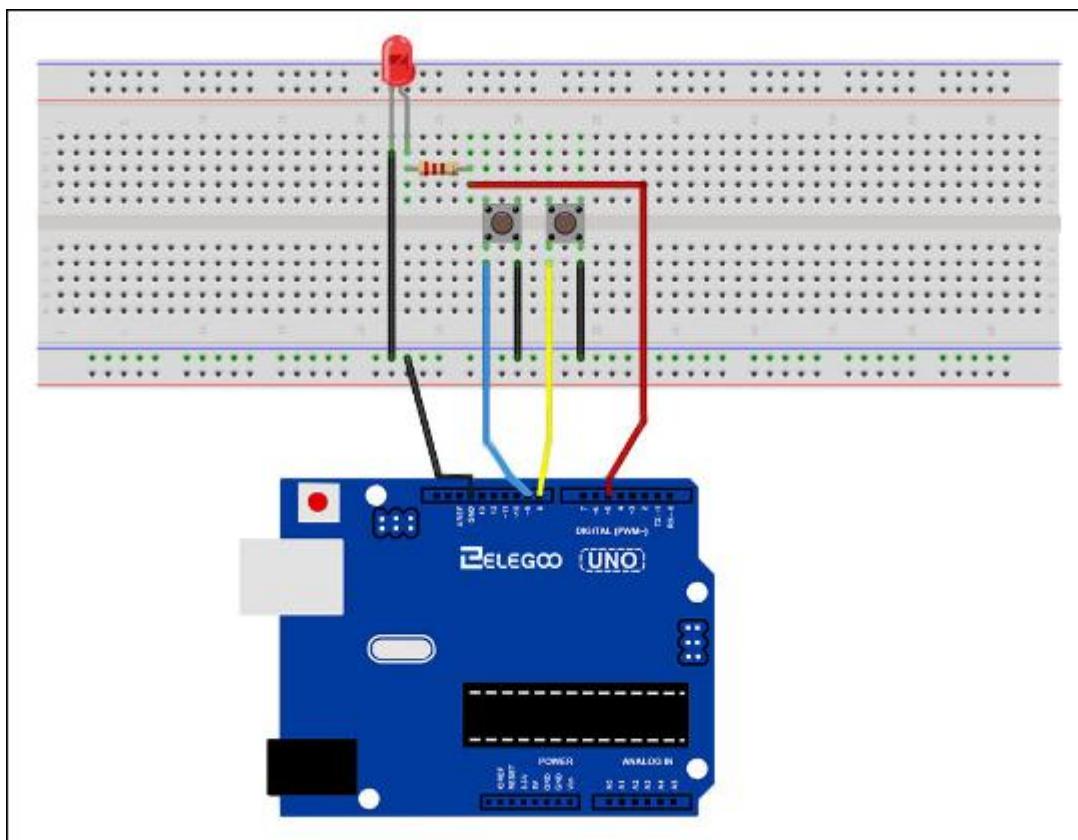
In realtà, ci sono solo veramente due connessioni elettriche. All'interno della confezione interruttore, perni B e C sono collegati tra loro, come sono A e D.

CONNESSIONE

schematico



schema di collegamento



Anche se i corpi degli interruttori sono quadrate, i perni sporgono da lati opposti del commutatore. Ciò significa che i perni saranno solo abbastanza distanti quando sono posizionati correttamente sulla basetta.

Ricordate che il LED deve avere il cavo negativo più breve verso destra.

Codice

Caricare il disegno sulla vostra tavola UNO. Premendo il pulsante in alto si accende il LED mentre si tiene premuto il pulsante in basso si spegnerlo.

La prima parte del disegno definisce tre variabili per i tre perni che devono essere utilizzati. Il 'ledPin' è il pin di uscita e 'buttonApin' faranno riferimento allo switch più vicino alla parte superiore della basetta e 'buttonBpin' per l'altro interruttore.

La funzione 'setup' definisce il ledPin come un'uscita come normale, ma ora abbiamo i due ingressi da affrontare. In questo caso, si usa il set del pinMode di essere 'INPUT_PULLUP' in questo modo:

```
pinMode(buttonApin, INPUT_PULLUP);  
pinMode(buttonBpin, INPUT_PULLUP);
```

La modalità di perno INPUT_PULLUP significa che il perno deve essere utilizzato come ingresso, ma se non altro è collegato all'ingresso, dovrebbe essere 'tirato' HIGH. In altre parole, il valore di default per l'ingresso è HIGH, se non viene tirato LOW dall'azione della pressione del tasto.

Questo è il motivo per cui gli interruttori sono collegati a GND. Quando si preme un interruttore, collega il pin di ingresso a GND, cosicché non è più ALTO.

Poiché l'ingresso è normalmente alto e diventa BASSO solo quando si preme il pulsante, la logica è un po capovolto. Tratteremo questo nella funzione 'loop'.

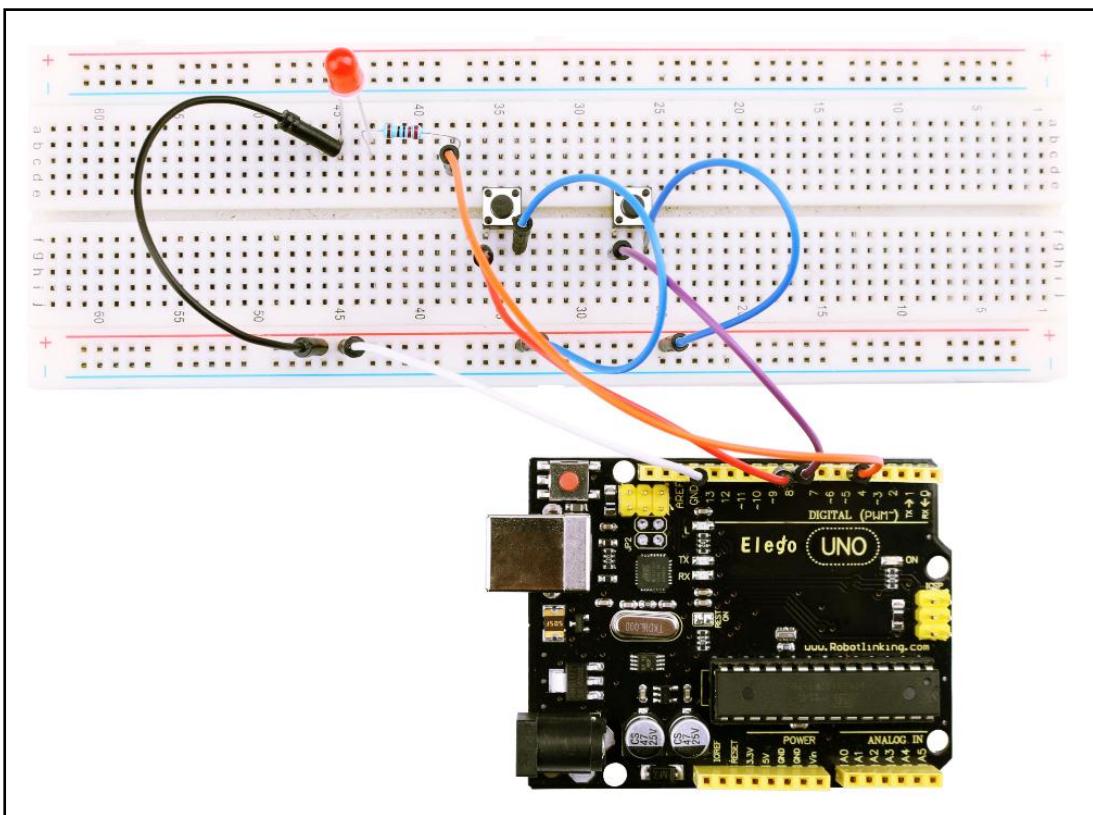
```
void loop()  
{  
    if (digitalRead(buttonApin) == LOW)  
    {  
        digitalWrite(ledPin, HIGH);  
    }  
    if (digitalRead(buttonBpin) == LOW)
```

```
{  
    digitalWrite(ledPin, LOW);  
}  
}
```

Nella funzione 'loop', ci sono due 'se' affermazioni, uno per ogni tasto. Ogni fa un 'digitalRead' sull'ingresso appropriata.

Ricordate che se si preme il pulsante, l'ingresso corrispondente sarà basso. Se il pulsante A è bassa, poi un 'digitalWrite' sul ledPin lo accende.

Allo stesso modo, se si preme il pulsante B, un basso viene scritto nel ledPin.



Lezione 6: Otto LED con 74HC595

Generali

In questa lezione, si apprenderà come utilizzare otto grandi LED rossi con un UNO senza dover rinunciare a 8 perno di uscita!

Anche se si potrebbe cablare otto LED ciascuno con una resistenza a un pin UNO si sarebbe rapidamente iniziare a corto di pin sul vostro UNO. Se non si dispone di un sacco di cose collegato al UNO. Va bene a farlo - ma spesso noi vogliamo pulsanti, sensori, servi, ecc e prima di sapere che non hai perni a sinistra. Così, invece di fare questo, si intende utilizzare un chip chiamato 74HC595 convertitore seriale in parallelo. Questo chip è dotato di otto uscite (perfetti) e tre ingressi che si utilizza per alimentare i dati in esso un po 'alla volta.

Questo chip lo rende un po 'più lento per guidare i LED (è possibile modificare solo i LED circa 500.000 volte al secondo invece di 8.000.000 di un secondo), ma è ancora molto veloce, modo più veloce di esseri umani in grado di rilevare, quindi ne vale la pena!

Componenti necessari

(1) Elegoo UNO R3

(1) Breadboard

(8) LEDs

(8) 220 ohm resistors

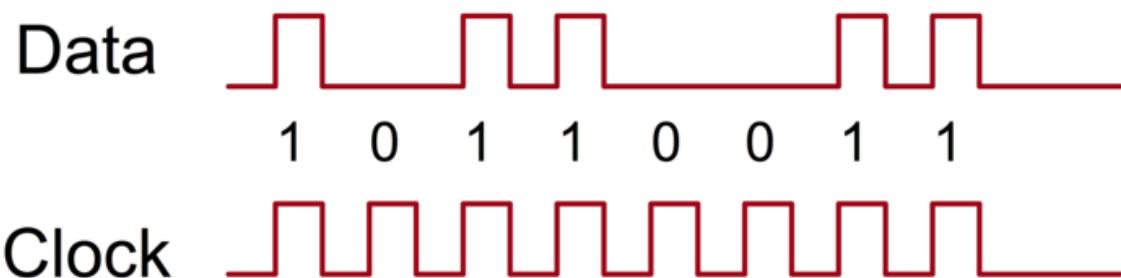
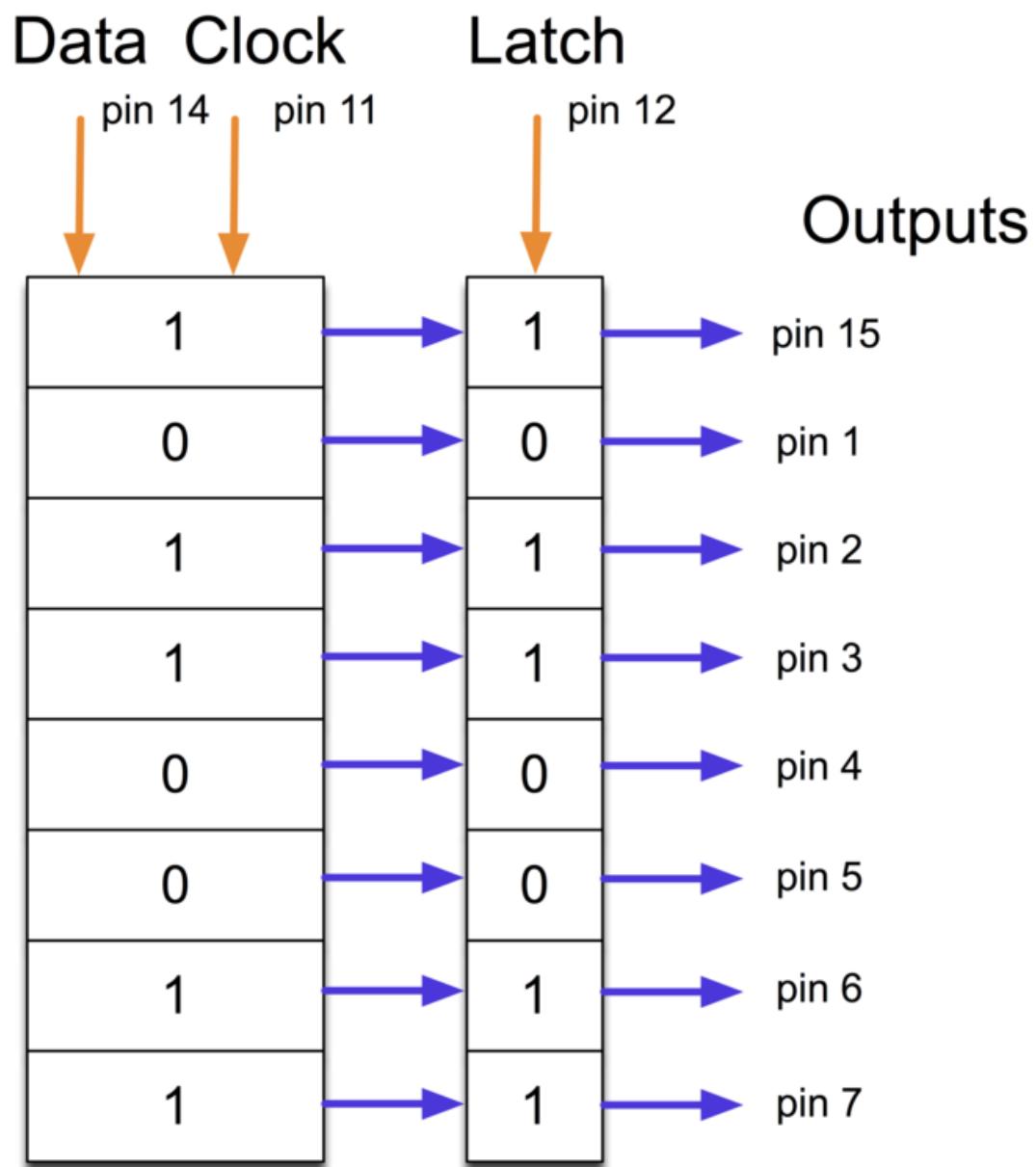
(1) 74HC595 IC

(14) M-M wires

Introduzione di Componente

74HC595 Registro a Scorrimento:

Il registro a scorrimento è un tipo di chip che contiene ciò che può essere pensato come otto locazioni di memoria, ognuno dei quali può essere un 1 o uno 0. Per impostare ciascuno di questi valori acceso o spento, si nutrono nei dati utilizzando il 'dat' e 'perno del chip orologio'.

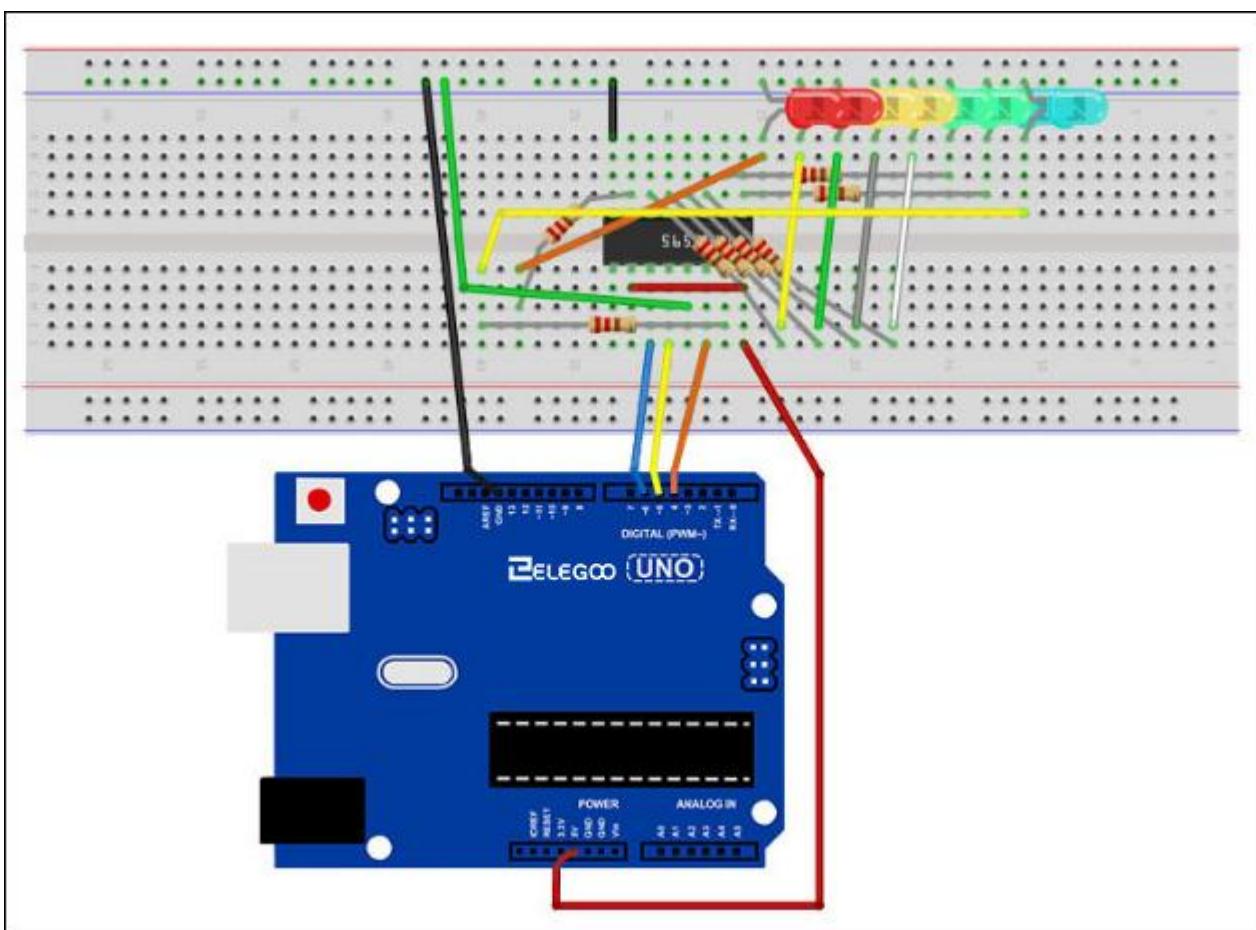


Il perno orologio ha bisogno di ricevere otto impulsi. Ad ogni impulso, se il perno di dati è alto, allora 1 viene spinto nel registro a scorrimento; in caso contrario, al 0. Quando sono stati ricevuti tutti gli otto impulsi, consentendo il 'latch' copie perno quegli otto valori al registro il fermo. Ciò è necessario; In caso contrario, i LED sbagliate sarebbero sfarfallio come i dati vengono caricati nel registro a scorrimento.

Il chip ha anche un pin di uscita di abilitazione (OE), che viene utilizzato per attivare o disattivare le uscite tutto in una volta. Si potrebbe collegare ad un perno UNO PWM-grado e utilizzare 'analogWrite' per controllare la luminosità dei LED. Questo pin è attivo basso, così abbiamo legarlo a GND.

Connessione

Schema di Collegamento



Come abbiamo otto LED e otto resistenze per collegare, ci sono in realtà un bel paio di collegamenti da effettuare.

È probabilmente più facile mettere il chip 74HC595 in primo luogo, come quasi tutto il resto si collega ad esso. Mettere in modo che la piccola tacca a forma di U è verso la parte superiore della basetta. Pin 1 del chip è di sinistra della tacca.

- Digitale 4 dal UNO va al perno #14 del registro a scorrimento
- Digitale 5 dal UNO va al perno #12 del registro a scorrimento
- Digitale 6 dal UNO va al perno #11 del registro a scorrimento

Tutti meno una delle uscite dalla IC è sul lato sinistro del chip. Quindi, per facilità di collegamento, che è dove i LED sono, anche.

Dopo il chip, mettere i resistori in atto. È necessario essere attenti a che nessuno dei conduttori delle resistenze si toccano. Si dovrebbe controllare di nuovo prima di collegare l'alimentazione del UNO. Se si hanno difficoltà a organizzare le resistenze, senza toccare la loro porta, quindi aiuta a ridurre i cavi in modo che essi sono distesi vicino alla superficie della basetta.

Quindi, posizionare i LED sul tagliere. I conduttori del LED più positive devono essere tutti verso il chip, a seconda di quale lato della basetta sono su.

Fissare il ponticello porta come indicato sopra. Non dimenticate quello che va dal pin 8 del circuito integrato alla colonna GND della basetta.

Caricare il disegno elencato al pò più tardi e provarlo. Ogni LED dovrebbe luce, a sua volta fino a quando tutti i LED sono accesi, e poi tutti si spegne e il ciclo si ripete.

Codice

La prima cosa da fare è definire i tre perni che stiamo per usare. Queste sono le uscite digitali UNO che saranno collegati al chiavistello, clock e dati pin del 74HC595.

```
int latchPin = 5;  
int clockPin = 6;  
int dataPin = 4;
```

Successivamente, si definisce una variabile chiamata "LED". Questo sarà utilizzato per contenere il modello di cui LED attualmente accesi o spenti. I dati di tipo 'byte' rappresenta i numeri con otto bit. Ogni bit può essere accesa o spenta, quindi questo è perfetto per tenere traccia di quale dei nostri otto LED siano accese o spente.

```
byte leds = 0;
```

La funzione 'setup' solo imposta i tre perni che stiamo usando per essere uscite digitali.

```
void setup()
```

```
{
```

```
  pinMode(latchPin, OUTPUT);
```

```
  pinMode(dataPin, OUTPUT);
```

```
  pinMode(clockPin, OUTPUT);
```

```
}
```

La funzione 'loop' trasforma inizialmente tutti i LED off, dando il 'LEDs' variabile il valore 0. E poi chiama 'updateShiftRegister' che invierà il modello 'LEDs' per il registro a scorrimento in modo che tutti i LED si spengono. Ci occuperemo di come 'updateShiftRegister' opere successive.

La funzione loop si ferma per mezzo secondo e poi comincia a contare da 0 a 7 utilizzando la 'per' anello e la variabile 'i'. Ogni volta che si utilizza la funzione Arduino 'bitset' per impostare il bit che controlla che il LED nel 'LEDs' variabili. E poi chiama anche 'updateShiftRegister' in modo che l'aggiornamento LEDs per riflettere ciò che è nel 'LEDs' variabili.

Vi è poi un secondo di ritardo mezzo prima di 'i' viene incrementato e il successivo LED è acceso

```
void loop()
```

```
{
```

```
  leds = 0;
```

```
  updateShiftRegister();
```

```
  delay(500);
```

```
  for (int i = 0; i < 8; i++)
```

```
  {
```

```
    bitSet(leds, i);
```

```
    updateShiftRegister();
```

```
    delay(500);
```

```
  }
```

```
}
```

La funzione 'aggiornamento di registro a scorrimento', prima di tutto, imposta il latchPin su BASSA, poi chiama la funzione UNO 'SHIFTOUT' prima di mettere il 'latchPin' su di nuovo alta. Questo richiede quattro parametri. I primi due sono i perni da utilizzare per i dati e Orologio, rispettivamente.

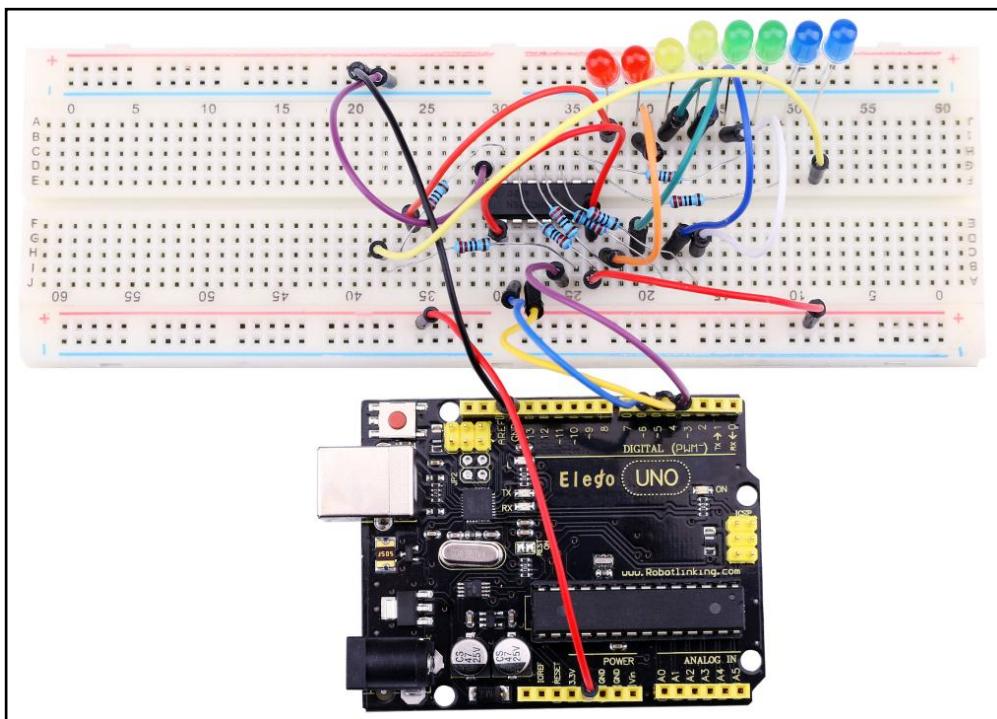
Il terzo parametro specifica a quale estremità dei dati che si desidera avviare. Stiamo per iniziare con il bit più a destra, che viene indicato come il 'Least Significant Bit' (LSB).

L'ultimo parametro è i dati effettivi essere spostato nel registro a scorrimento che, in questo caso, è 'LEDs.

void updateShiftRegister()

```
{  
    digitalWrite(latchPin, LOW);  
    shiftOut(dataPin, clockPin, LSBFIRST, leds);  
    digitalWrite(latchPin, HIGH);  
}
```

Se desiderate svolgere uno dei LED off piuttosto che su, si potrebbe chiamare una funzione Arduino simile (`bitClear`) con la variabile '`LEDs`'. Questo imposterà quel po 'di "LEDs" per essere 0 e si sarebbe quindi solo bisogno di seguire con una chiamata a' aggiornamento di registro a scorrimento 'per aggiornare i LED attuali.



Lezione 7: Il Monitore Seriale

Generali

In questa lezione, si costruirà su Lezione 6, aggiungendo la possibilità di controllare i LED dal computer utilizzando il monitor seriale Arduino. Il monitor seriale è il 'tether' tra il computer e il vostro UNO. Esso consente di inviare e ricevere messaggi di testo, a portata di mano per il debug e anche che controllano l'ONU da una tastiera! Ad esempio, si sarà in grado di inviare comandi dal computer per accendere i LED.

In questa lezione, si utilizzerà esattamente le stesse parti e un layout simile a tagliere Lezione 6. Quindi, se non l'hai già fatto, seguire Lezione 6 ora.

Passi

Dopo aver caricato questo schizzo sul vostro UNO, cliccare sul pulsante più a destra della barra degli strumenti nella Arduino. Il pulsante è cerchiato in basso.



Si aprirà la finestra seguente.

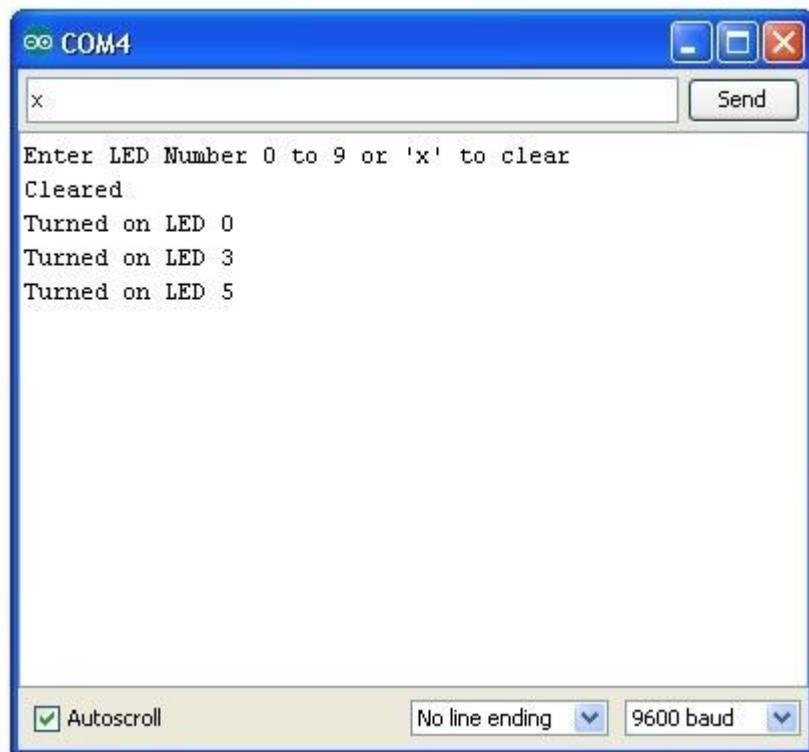


Questa finestra è chiamata Monitore Seriale, ed è parte del software di Arduino IDE. Il suo compito è quello di permettere a entrambi di inviare messaggi dal computer a una scheda UNO (via USB), e anche di ricevere messaggi dal UNO.

Il messaggio "Enter LED Numero 0 a 9 o 'x' per cancellare" è stato inviato dal Arduino. E ci sta dicendo quello che comanda possiamo inviare al Arduino: inviare il 'x' (per accendere tutti i LED off) o il numero del LED che si desidera attivare (dove 0 è il LED in basso, 1 è il prossimo uno, fino a 7 per il LED superiore).

Provate a digitare i seguenti comandi nella zona superiore del monitor di serie che è a livello con il tasto 'Invia'. 'Invia', dopo aver digitato ciascuno di questi personaggi: x 0 3 5

Digitando x non avrà alcun effetto se i LED sono già tutti fuori, ma come si entra ogni numero, il LED corrispondente si illumina e si otterrà al messaggio di conferma dalla scheda UNO. Apparirà la Serial Monitor come illustrato di seguito.



Digitare X di nuovo e premere 'Invia' per spegnere tutti i LED.

Codice

Come ci si potrebbe aspettare, lo schizzo si basa sul disegno usato nella lezione 6. Quindi, ci limiteremo a coprire i nuovi bit qui. Potrai trovare utile fare riferimento alla piena schizzo nel vostro IDE Arduino.

Nella funzione 'setup', ci sono tre nuove linee alla fine:

```
void setup()
{
    pinMode(latchPin, OUTPUT);
    pinMode(dataPin, OUTPUT);
    pinMode(clockPin, OUTPUT);
    updateShiftRegister();
    Serial.begin(9600);
    while (! Serial); // Wait until Serial is ready - Leonardo
    Serial.println("Enter LED Number 0 to 7 or 'x' to clear");
}
```

Al primo, abbiamo il comando 'Serial.begin (9600)'. Questo inizia comunicazione seriale, in modo che l'ONU può inviare comandi attraverso il collegamento USB. Il valore 9600 è chiamato il 'velocità di trasmissione' della connessione. Questo è quanto velocemente i dati devono essere inviati. È possibile modificare questo a un valore più alto, ma si dovrà anche cambiare il monitor seriale Arduino allo stesso valore. Ne discuteremo più avanti; per ora, lasciare a 9600.

La linea che inizia con 'mentre' assicura che ci sia qualcosa all'altra estremità della connessione USB per Arduino per parlare prima che inizi l'invio di messaggi. In caso contrario, il messaggio potrebbe essere inviato, ma non visualizzato. Questa linea è in realtà necessaria solo se si utilizza un Arduino Leonardo perché l'ONU di Arduino ripristina automaticamente la scheda Arduino quando si apre il monitor seriale, mentre questo non accade con il Leonardo. L'ultima delle nuove linee in 'setup' invia il messaggio che si vede nella parte superiore del monitor seriale.

La funzione 'loop' è dove avviene tutta l'azione:

```
void loop()
{
    if (Serial.available())
    {
        char ch = Serial.read();
        if (ch >= '0' && ch <= '7')
        {
            int led = ch - '0';
            bitSet(leds, led);
            updateShiftRegister();
            Serial.print("Turned on LED ");
            Serial.println(led);
        }
        if (ch == 'x')
        {

```

```
    leds = 0;  
  
    updateShiftRegister();  
  
    Serial.println("Cleared");  
  
}  
  
}  
  
}
```

Tutto ciò che accade all'interno del ciclo è contenuta all'interno di un 'se' dichiarazione. Quindi, a meno la chiamata alla funzione incorporata Arduino 'Serial.available ()' è 'vero', poi niente altro accadrà.

Serial.available () restituirà 'true' se i dati sono stati inviati al ONU e se ci sono dati pronti per essere elaborati. I messaggi in arrivo sono tenuti in quello che viene chiamato un buffer e Serial.available () restituisce vero se quel buffer non è vuoto.

Se è stato ricevuto un messaggio, allora è alla prossima riga di codice:

```
char ch = Serial.read();
```

Si possono leggere i carattere successivo dal buffer e rimuove dal buffer. Si assegna anche al 'ch' variabile. Il 'ch' variabile è di tipo 'char' che sta per 'carattere'. Come suggerisce il nome, che detiene un singolo carattere.

Se avete seguito le istruzioni del messaggio nella parte superiore del monitor seriale, allora questo personaggio o sarà al numero a cifra singola compreso tra 0 e 7 o la lettera 'x'.

Il 'se' dichiarazione sulla riga successiva verifica per vedere se è una sola cifra vedendo se 'ch' è maggiore o uguale al carattere '0' e minore o uguale al carattere '7'. Sembra un po 'strani personaggi si confrontano in questo modo, ma è perfettamente accettabile.

Ogni carattere è rappresentato da un numero unico, chiamato il suo valore ASCII. Ciò significa che quando mettiamo a confronto caratteri utilizzando `<=` o `=`, è in realtà i valori ASCII che vengono confrontati.

Se superare il test, poi si arriva alla riga successiva:

```
int led = ch - '0';
```

Ora stiamo eseguendo aritmetica sui personaggi! Stiamo sottraendo la cifra '0' da qualunque

stato inserito cifre. Quindi, se si è digitato '0' e poi '0' - '0' sarà uguale a 0. Se è stato digitato '7' poi '7' - '0' sarà uguale al numero 7, perché è in realtà i valori ASCII che vengono utilizzati nella sottrazione.

Dal momento che conosciamo il numero dei LED che vogliamo attivare, abbiamo solo bisogno di fissare quel pò nel LEDs 'variabile e aggiornare il registro a scorrimento.

```
bitSet(leds, led);
```

```
updateShiftRegister();
```

Le prossime due linee di scrivere di nuovo un messaggio di conferma per il Monitore Seriale.

```
Serial.print("Turned on LED ");
```

```
Serial.println(led);
```

La prima linea utilizza seriale. stampare piuttosto che seriale. println. La differenza tra i due è che Serial.print non inizia una nuova riga dopo la stampa ciò che è in suo parametro. Usiamo questo in prima linea perché siamo la stampa del messaggio in due parti: il bit generale 'Accesso LED' e quindi il numero dei LED.

Il numero dei LED si svolge in un 'int' variabile piuttosto che essere una stringa di testo. Seriale. stampa può prendere o una stringa di testo racchiusa tra virgolette, un 'int' o praticamente qualsiasi tipo di variabile.

Dopo la dichiarazione di 'se' che gestisce il caso, quando una sola cifra è stata gestita, c'è una seconda dichiarazione 'se' che controlla per vedere se 'ch' è la lettera 'x'.

```
if (ch == 'x')  
{  
    leds = 0;  
    updateShiftRegister();  
    Serial.println("Cleared");  
}
```

Se si tratta, allora cancella tutti i LED e invia un messaggio di conferma.

Lezione 8: Fotocellula

Generali

In questa lezione imparerai come misurare l'intensità della luce utilizzando un ingresso analogico. Si costruirà sul circuito e utilizzare il livello di luce per controllare il numero di LED da illuminare.

La fotocellula è al fondo della basetta, dove il piatto era superiore.

Componenti necessari

(1) Elegoo UNO R3

(1) Breadboard

(8) LEDs

(8) 220 ohm resistori

(1) 1k ohm resistori

(1) 74HC595 IC

(1) Fotocellula

(14) M-M fili

Introduzioni di Componente

FOTOCELLULA:

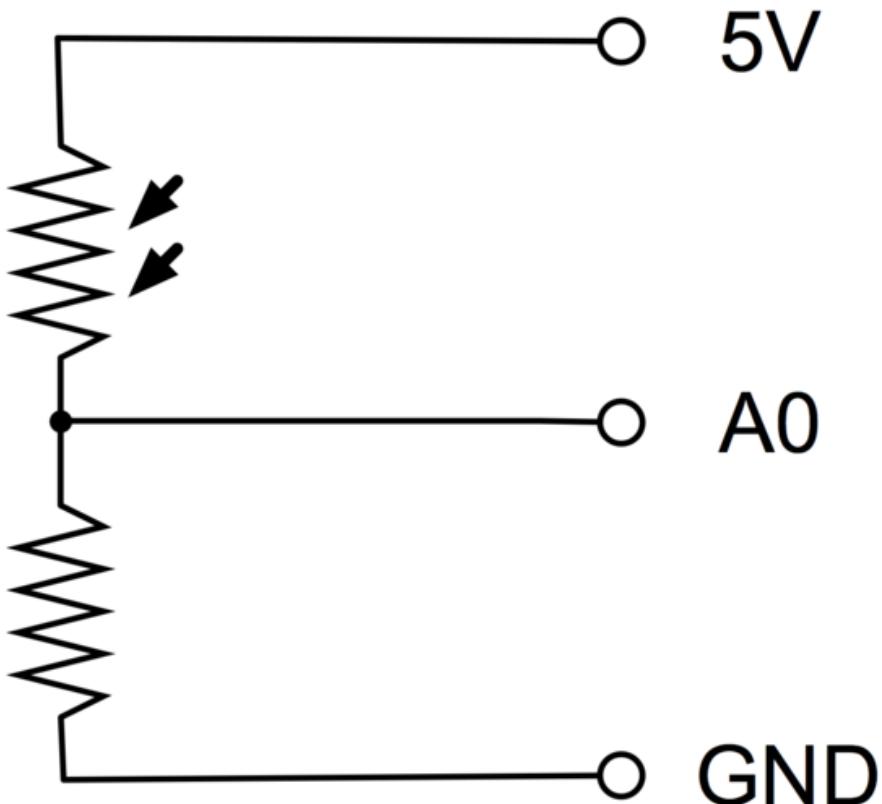
La fotocellula utilizzata è del tipo chiamato un resistore dipendente luce (LDR). Come suggerisce il nome, questa componente comporta come un resistore, tranne che la resistenza cambia in risposta alla quantità di luce sta cadendo su di essi.

Questo ha una resistenza di circa 50 k in prossimità buio e 500 Ω in piena luce. Per convertire questo valore variabile della resistenza in qualcosa che possiamo misurare in ingresso analogico di una scheda UNO R3, ha bisogno di essere convertito in una tensione.

Il modo più semplice per farlo è quello di combinare con un resistore fisso.

Photocell

**Fixed
Resistor
1 kΩ**



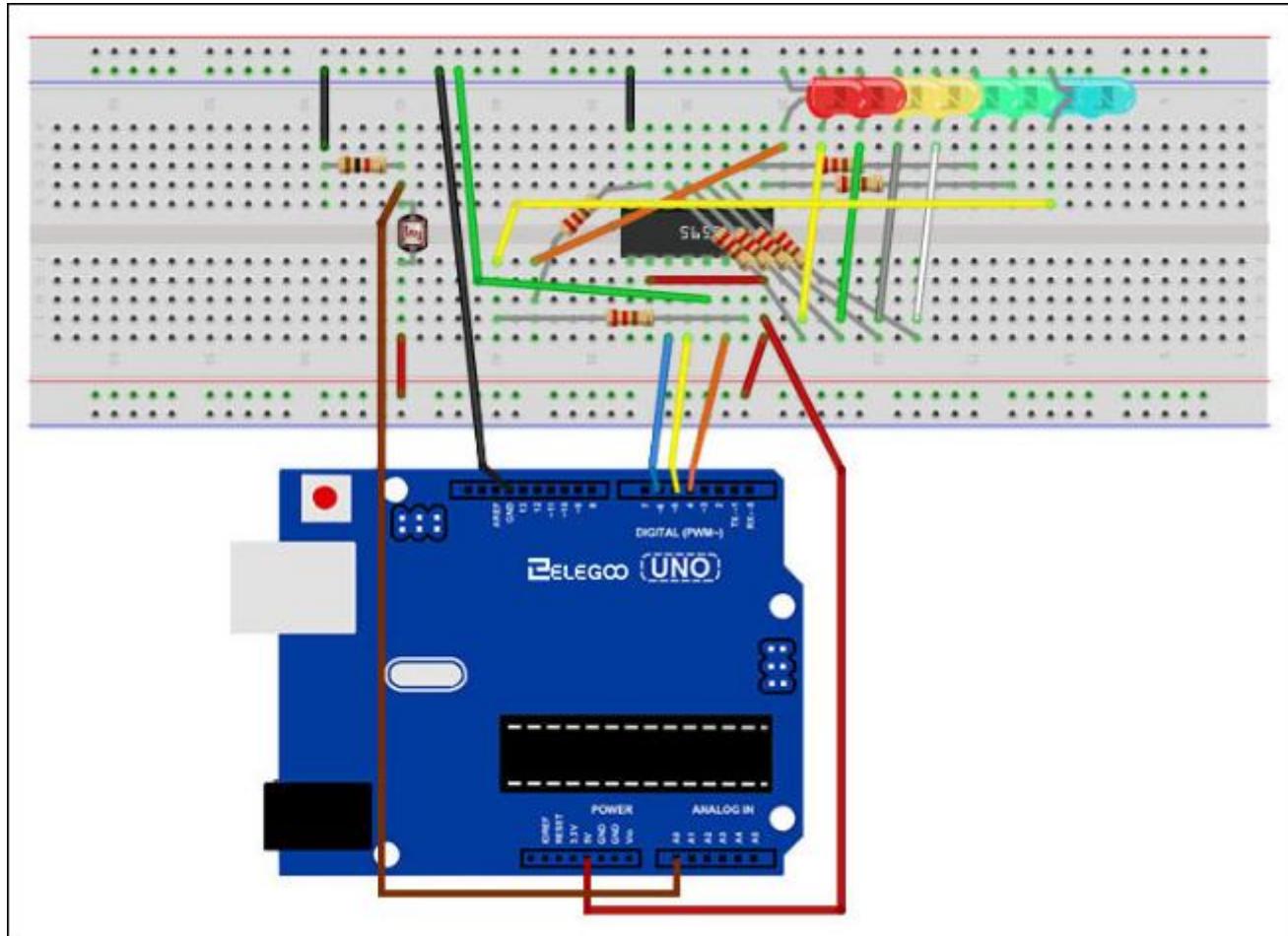
Il resistore e la fotocellula insieme si comportano come una pentola. Quando la luce è molto luminoso, allora la resistenza della fotocellula è molto basso rispetto al resistore valore fisso, e così è come se il piatto erano rivolti al massimo.

Quando la fotocellula in luce opaca, la resistenza diventa maggiore del 1 resistore fisso $k\Omega$ ed è come se il piatto venisse girato verso GND.

Caricare il disegno riportato nella sezione successiva e provare che copre la fotocellula con il dito, e poi tenerlo vicino ad una fonte di luce.

Connessione

Schema di collegamento



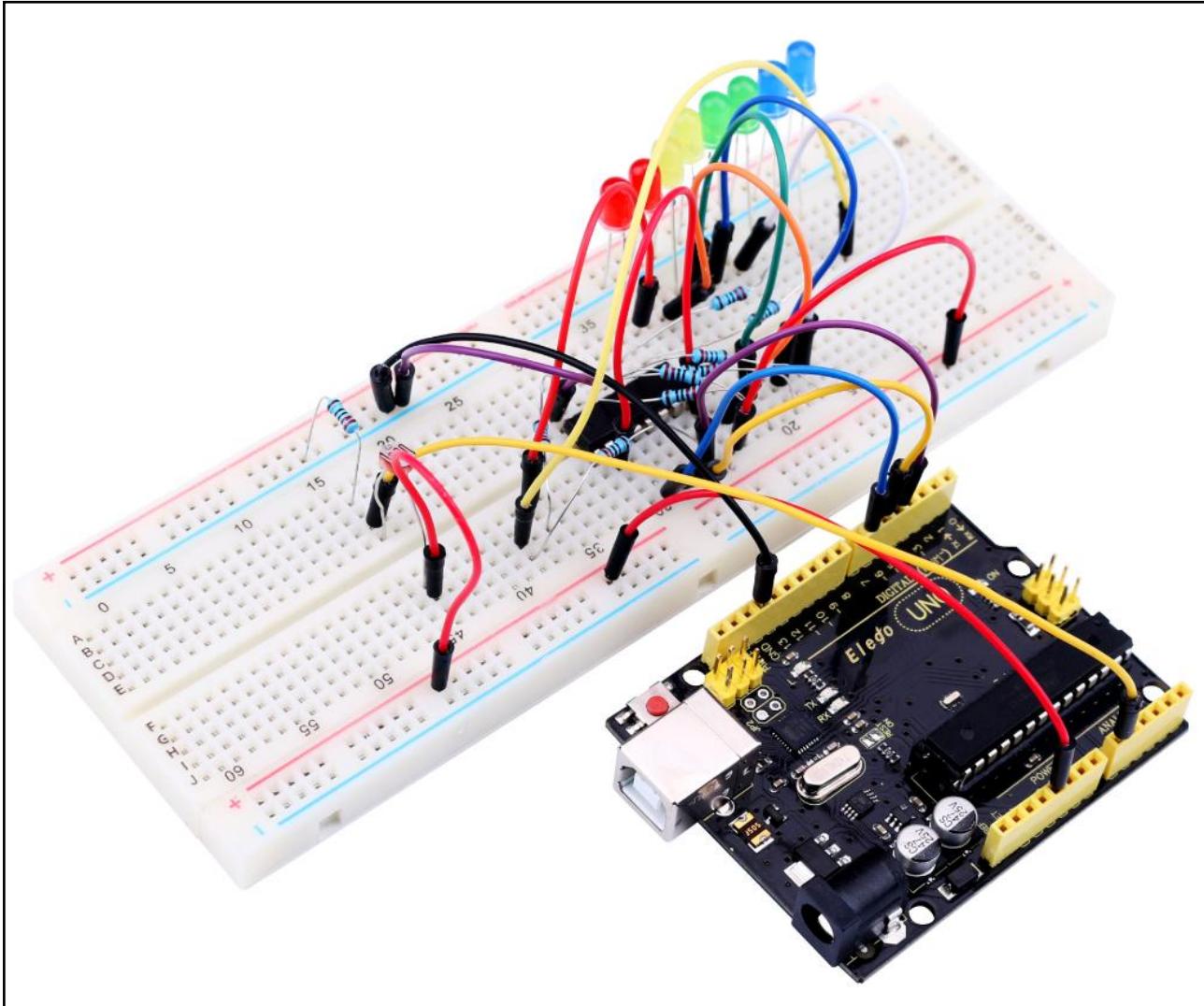
Codice

La prima cosa da notare è che abbiamo cambiato il nome del perno analogico di essere 'lightpin' piuttosto che 'potPin' da quando non abbiamo più una pentola collegata.

L'unica altra modifica sostanziale allo schizzo è la linea che calcola quante dei LED alla luce:

```
int numLEDSLit = reading / 57; // all LEDs lit at 1k
```

Questa volta, si divide la lettura grezza da 57 invece che 114. In altre parole, ci dividiamo per la metà di quanto abbiamo fatto con il piatto di dividerlo in nove zone, da nessun LED acceso a tutti gli otto acceso. Questo fattore supplementare è di spiegare la 1 resistore fisso $1\text{k}\Omega$. Questo significa che quando la fotocellula ha una resistenza di 1 k (uguale al resistore fisso), la lettura grezzo sarà $1023/2 = 511$. Ciò equivale a tutti i LED essere acceso e quindi al bit (numLEDSLit) sarà 9.



Lezione 9: Facendo I Suoni

Generali

In questa lezione, imparerete come generare un suono con un ronzatore attivo.

Componenti necessari

(1) Elegoo UNO R3

(1) Ronzatore attivo

(2) F-M fili

Introduzione di Componente

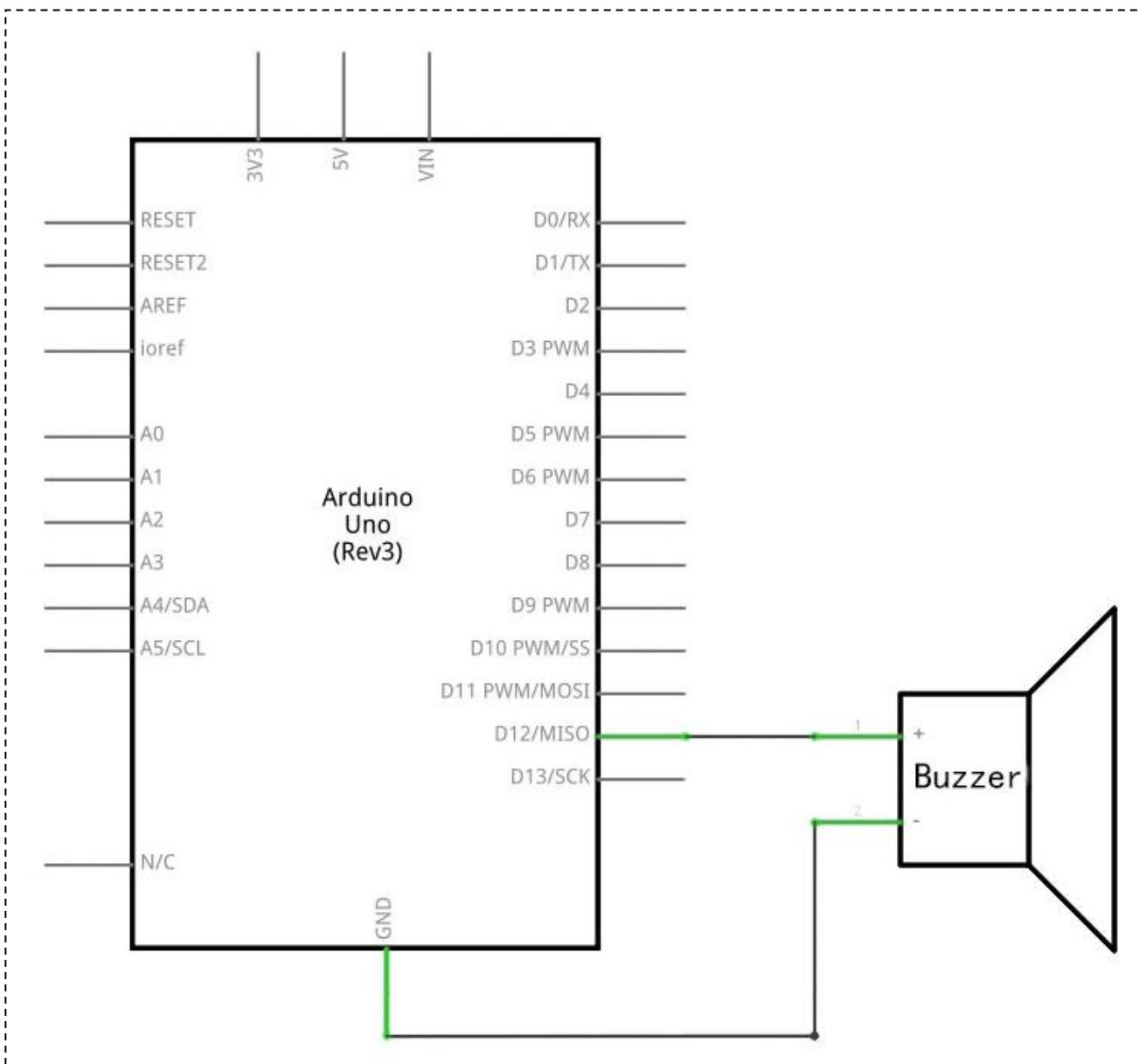
RONZATORE:

Ronzatore elettronici sono DC-alimentato e dotato di un circuito integrato. Essi sono ampiamente utilizzati nei computer, stampanti, fotocopiatrici, allarmi, giocattoli elettronici, dispositivi elettronici per autoveicoli, telefoni, timer e altri prodotti elettronici per dispositivi vocali. Ronzatori possono essere classificati come quelli attivi e passivi. Ruotare i perni di due segnali acustici rivolti verso l'alto. Quello con al circuito verde è un ronzatore passivo, mentre l'altra chiusa con un nastro nero è uno attivo.

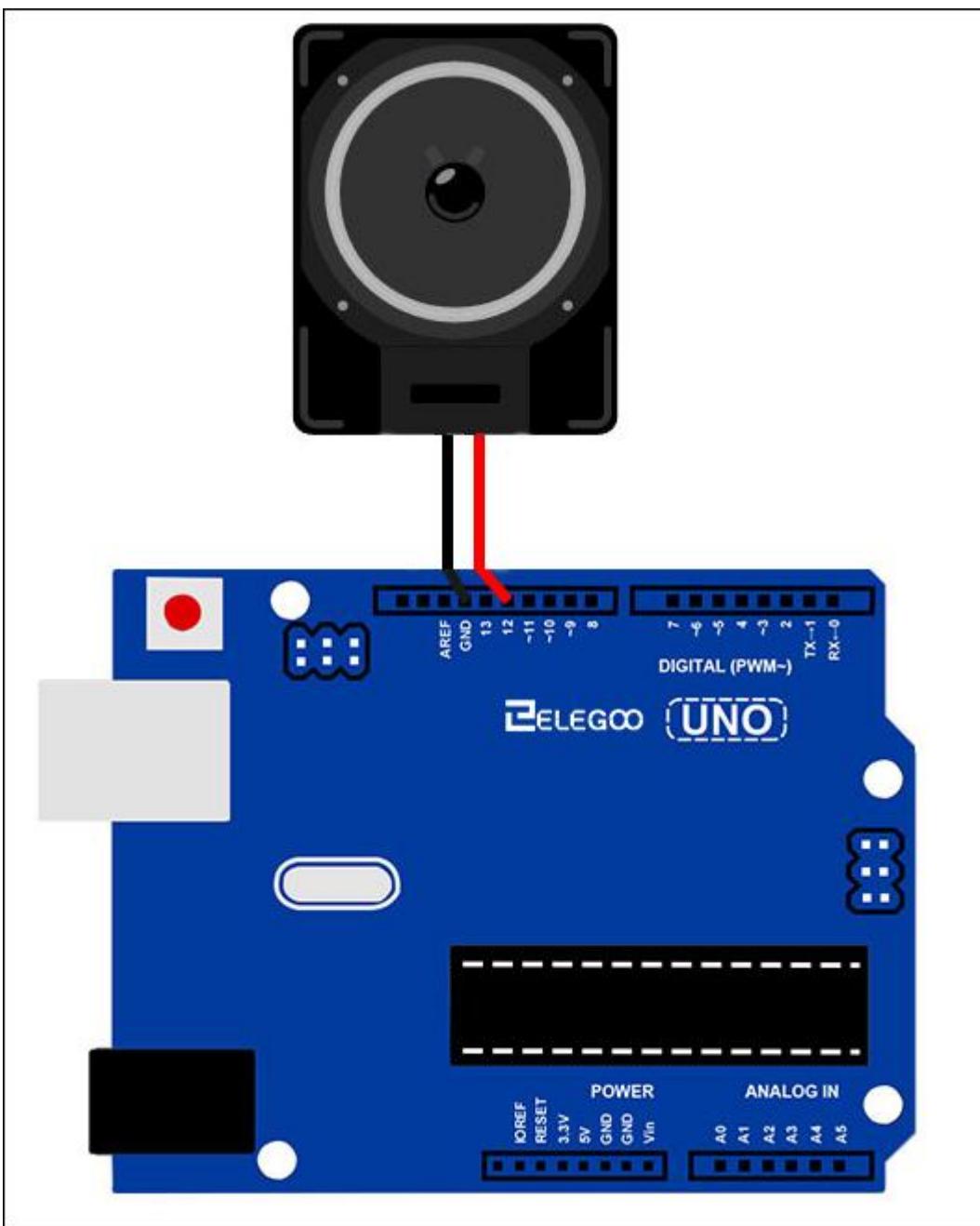
La differenza tra i due è che un buzzer attiva ha un oscillante incorporato sorgente, quindi genererà un suono quando elettrificato. Un ronzatore passivo non ha una tale sorgente in modo che non twittare se si utilizzano segnali DC; invece, è necessario utilizzare onde quadre la cui frequenza è compresa tra 2K e 5K a guidarla. Il ronzatore attivo è spesso più costosa di quella passiva causa di molteplici oscillante incorporato circuiti.

Connessione

Schematico

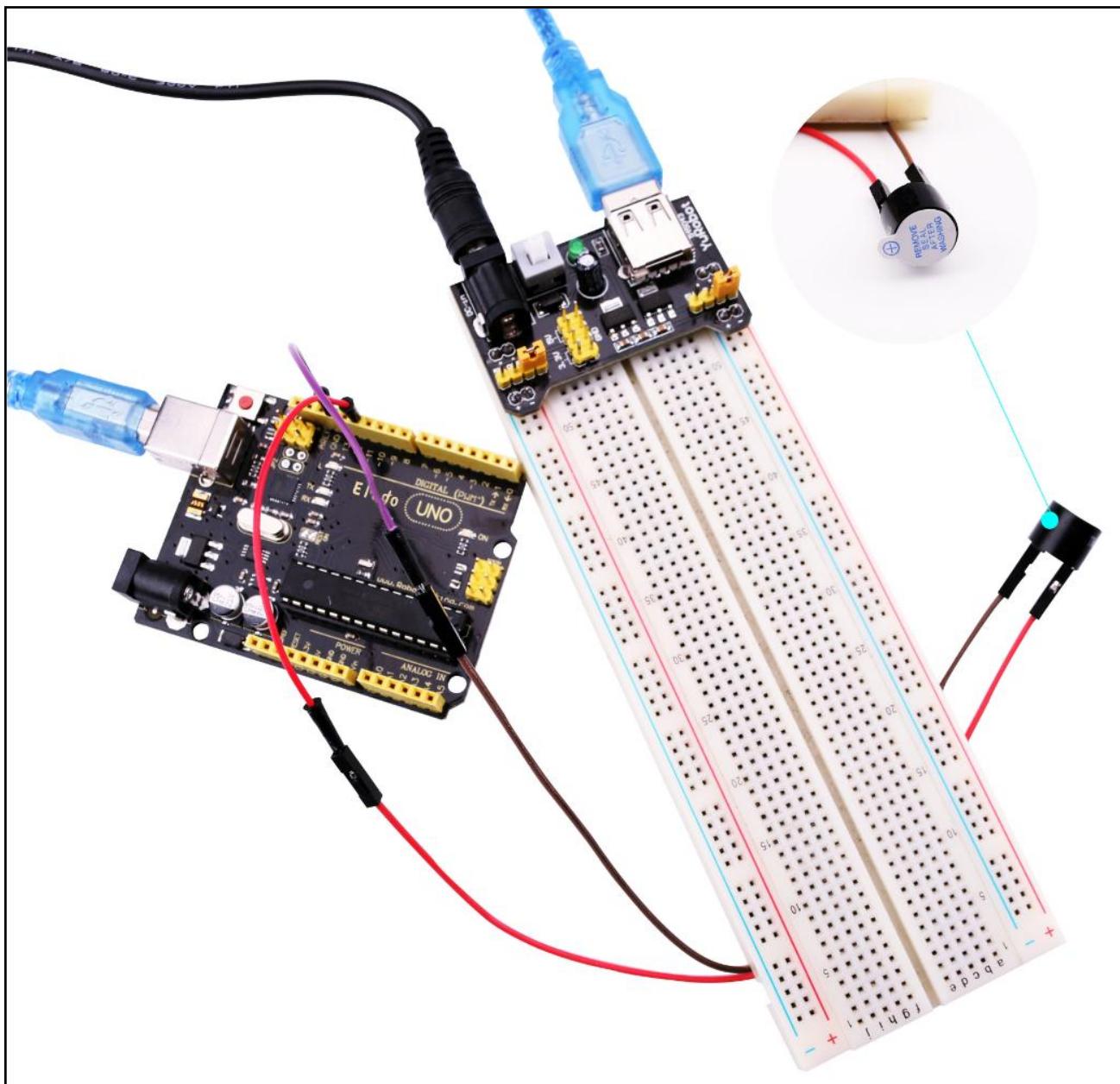


Schema di collegamento



Codice

Vedere il file di codice.



Lezione 10: Passivo Ronzatore

Generali

In questa lezione, si apprenderà come utilizzare un cicalino passivo.

Lo scopo dell'esperimento è quello di generare otto diversi suoni, ogni suono della durata di 0,5 secondi: da Alto Do (523Hz), Re (587Hz), Mi (659Hz), Fa (698Hz), So (784Hz), La (880Hz), Si (988Hz) per Treble Do (1047Hz).

Componenti necessari

(1) Elegoo UNO R3

(1) Passivo Ronzatore

(2) F-M fili

Introduzione di Componente

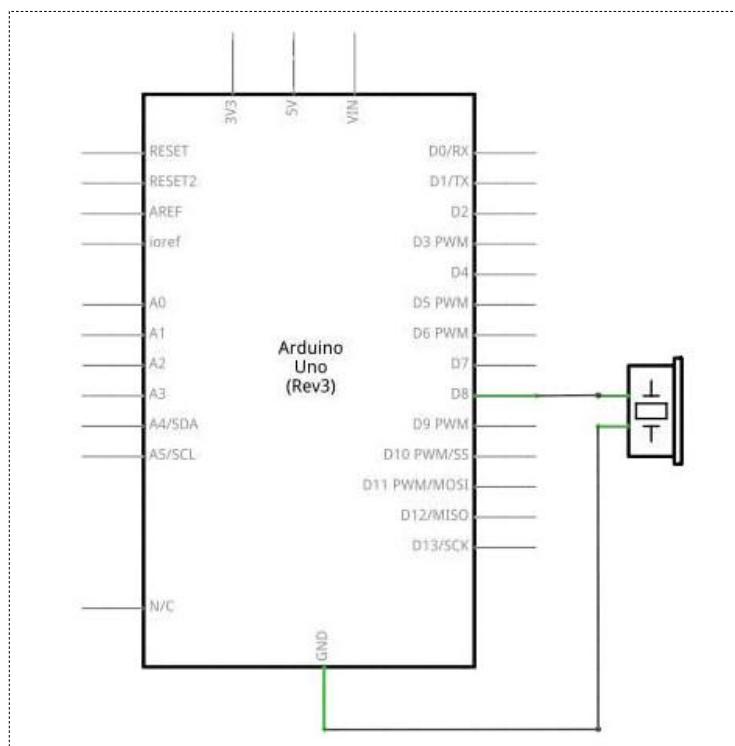
Passivo Ronzatore:

Il principio di funzionamento passivo ronzatore utilizzare audio generatore PWM per rendere l'aria a vibrare. Opportunamente modificato fino a quando la frequenza di vibrazione, può generare suoni diversi. Per esempio, inviando un impulso di 523Hz, che può generare Alto Do, impulso di 587Hz, che può generare di fascia media Re, impulso di 659Hz, può produrre di fascia media Mi. Con il segnale acustico, è possibile riprodurre una canzone.

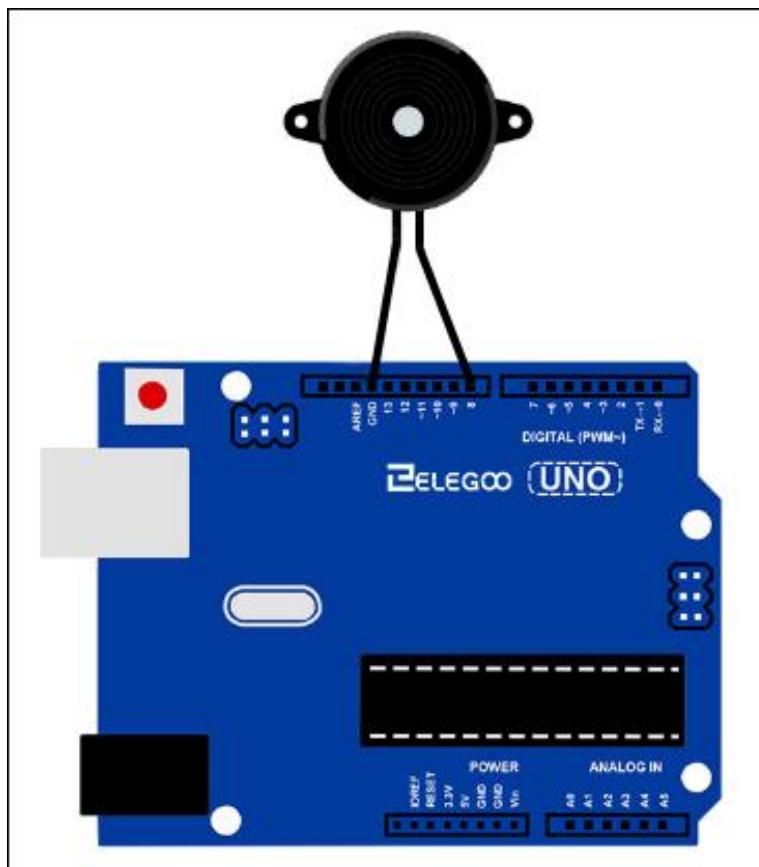
Dobbiamo fare attenzione a non utilizzare la funzione di UNO R3 scheda analogica Scrittura () per generare un impulso per il cicalino, perché l'uscita di impulso di analogWrite () è fissa (500Hz).

Connessione

Schematico



Schema di collegamento



Cablaggio di un segnale acustico collegato alla scheda UNO R3, il rosso (positivo) al perno di 8, filo nero (negativo) al GND

Codice

Prima di poter eseguire questo, assicurarsi di aver installato il <piazzole> libreria e reinstallarlo, se necessario. In caso contrario, il codice non funzionerà.

Lezione 11: Sfera Interruttore

Generali

In questa lezione, si apprenderà come utilizzare un interruttore palla al fine di rilevare piccolo angolo di inclinazione.

Componenti necessari

(1) Elegoo UNO R3

(1) Interruttore palla

(2) F-M fili

Introduzione di Componente

Sensore di inclinazione:

I sensori di inclinazione consentono di rilevare l'orientamento o l'inclinazione. Sono piccole, poco costoso, a bassa potenza e facile da usare. Se usato correttamente, non si usura. La loro semplicità li rende popolari per i giocattoli, dispositivi e gli elettrodomestici. A volte, essi sono indicati come "interruttori a mercurio", "Inclinazione interruttori" o "rotolamento Sensori a sfera" per ovvie ragioni.

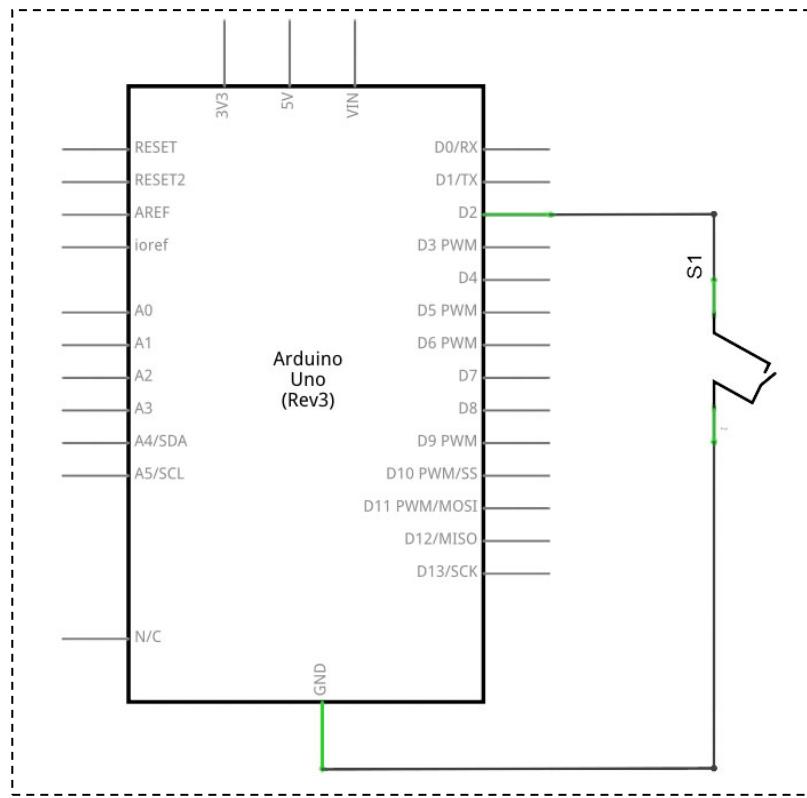
Essi sono generalmente costituiti da una cavità di qualche tipo (cilindrico è popolare, anche se non sempre) con una massa libera conduttiva all'interno, come una goccia di mercurio o rotolamento della sfera. Una estremità della cavità ha due elementi conduttori (poli). Quando il sensore è orientato in modo che tale fine è basso, i rulli di massa sulla poli e li bicchierini, agiscono come un tiro interruttore.

Anche se non è preciso e flessibile come un accelerometro completo, interruttori d'inclinazione in grado di rilevare il movimento o l'orientamento. Un altro vantaggio è che i grandi possono passare il potere da soli.

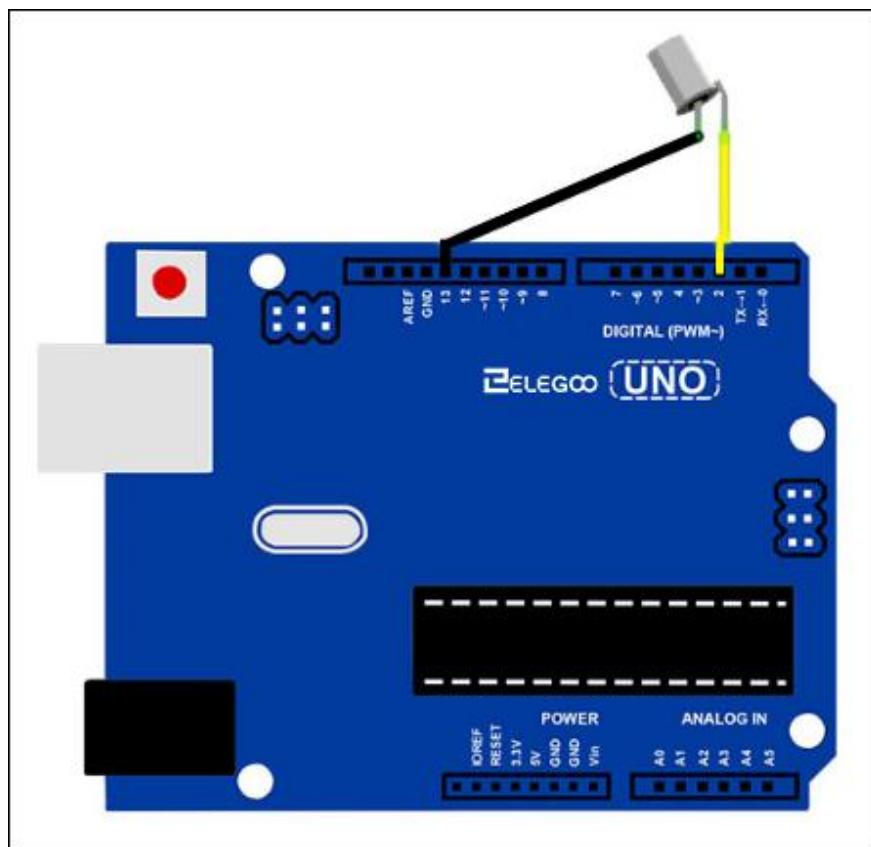
Accelerometri, d'altra parte, digitale o uscita tensione analogica che devono poi essere analizzati utilizzando circuiteria supplementare.

Connessione

Schematico

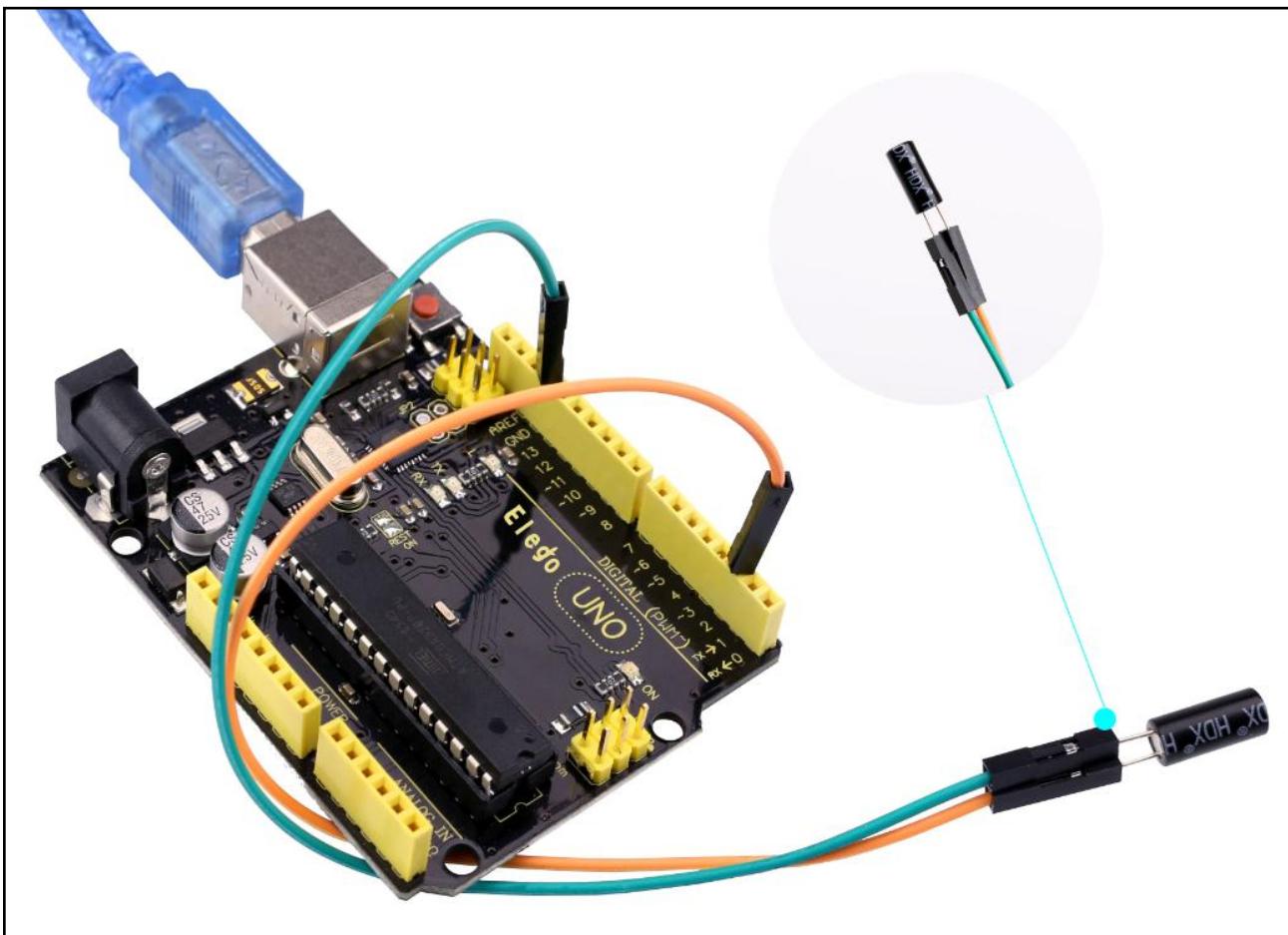


Schema di collegamento



Codice

Vedere il fine di codice.



Lezione 12: Relay

Generali

In questa lezione, si apprenderà come utilizzare un relay.

Componenti necessari

- (1) Elegoo UNO R3
- (1) Breadboard
- (1) PN2222
- (1) 1N4007
- (1) 220 ohm resistori
- (1) 6v DC motore
- (1) Relay
- (1) modulo di alimentazione di breadboard
- (1) 9v adattatore di alimentazione
- (8) M-M fili

Introduzione di Componente

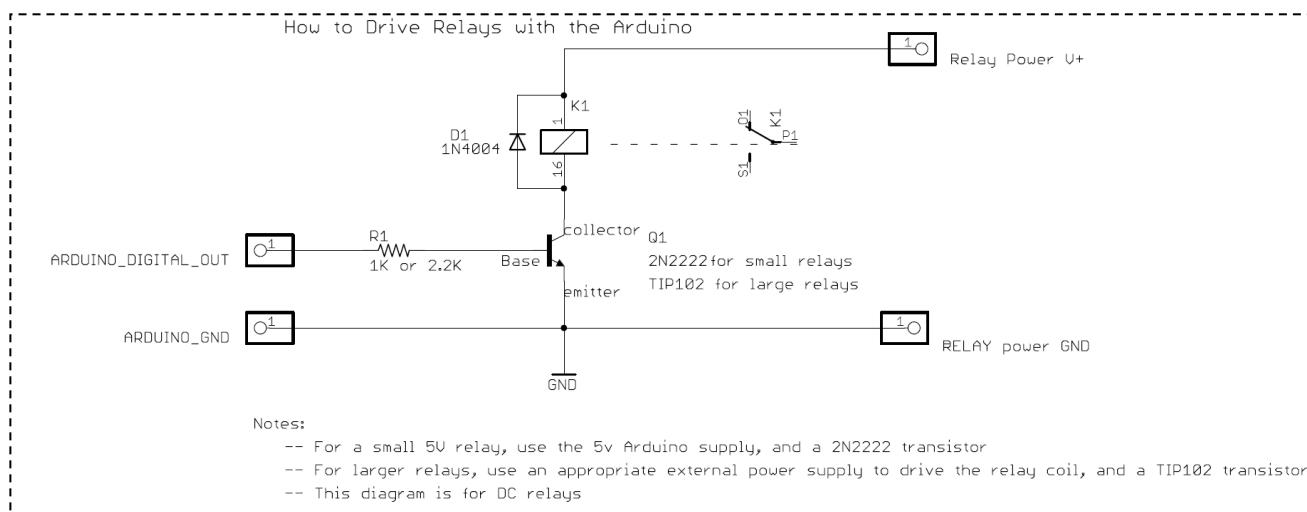
Relay:

Un relay è un interruttore ad azionamento elettrico. Molti Relays utilizzano un elettromagnete per operare meccanicamente un interruttore, ma altri principi operativi sono utilizzati anche come nei relays a stato solido. I relay sono utilizzati dove è necessario controllare un circuito da un segnale a bassa potenza (con isolamento elettrico completo tra controllo e circuiti controllati), o quando più circuiti devono essere controllati da un segnale. I primi Relays sono stati utilizzati nei circuiti telegrafici a lunga distanza come amplificatori. Hanno ripetuto il segnale proveniente da un circuito e ri-trasmessi su un altro circuito. Relays sono stati ampiamente utilizzati nelle centrali telefoniche e primi computer per eseguire operazioni logiche.

Un tipo di relay che può gestire l'elevata potenza necessaria per comandare direttamente un motore elettrico o altri carichi è chiamato un contattore. Relay a stato solido controllo circuiti

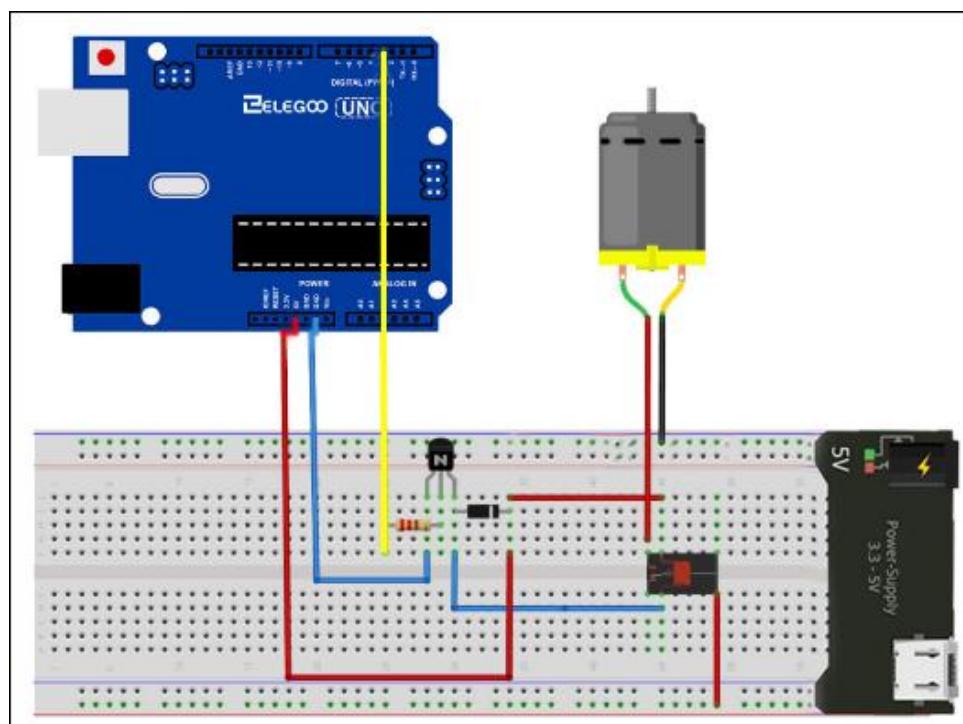
di potenza senza parti in movimento, invece di usare un dispositivo a semiconduttore per eseguire la commutazione. Relay con caratteristiche di funzionamento calibrate e talvolta più bobine operativi sono usati per proteggere i circuiti elettrici da sovraccarico o difetti. Nei sistemi di alimentazione elettrica moderni, queste funzioni sono svolte da strumenti digitali chiamati "relay di protezione".

Di seguito è riportato lo schema di come guidare relay con Arduino (caricato dal arduino.cc)



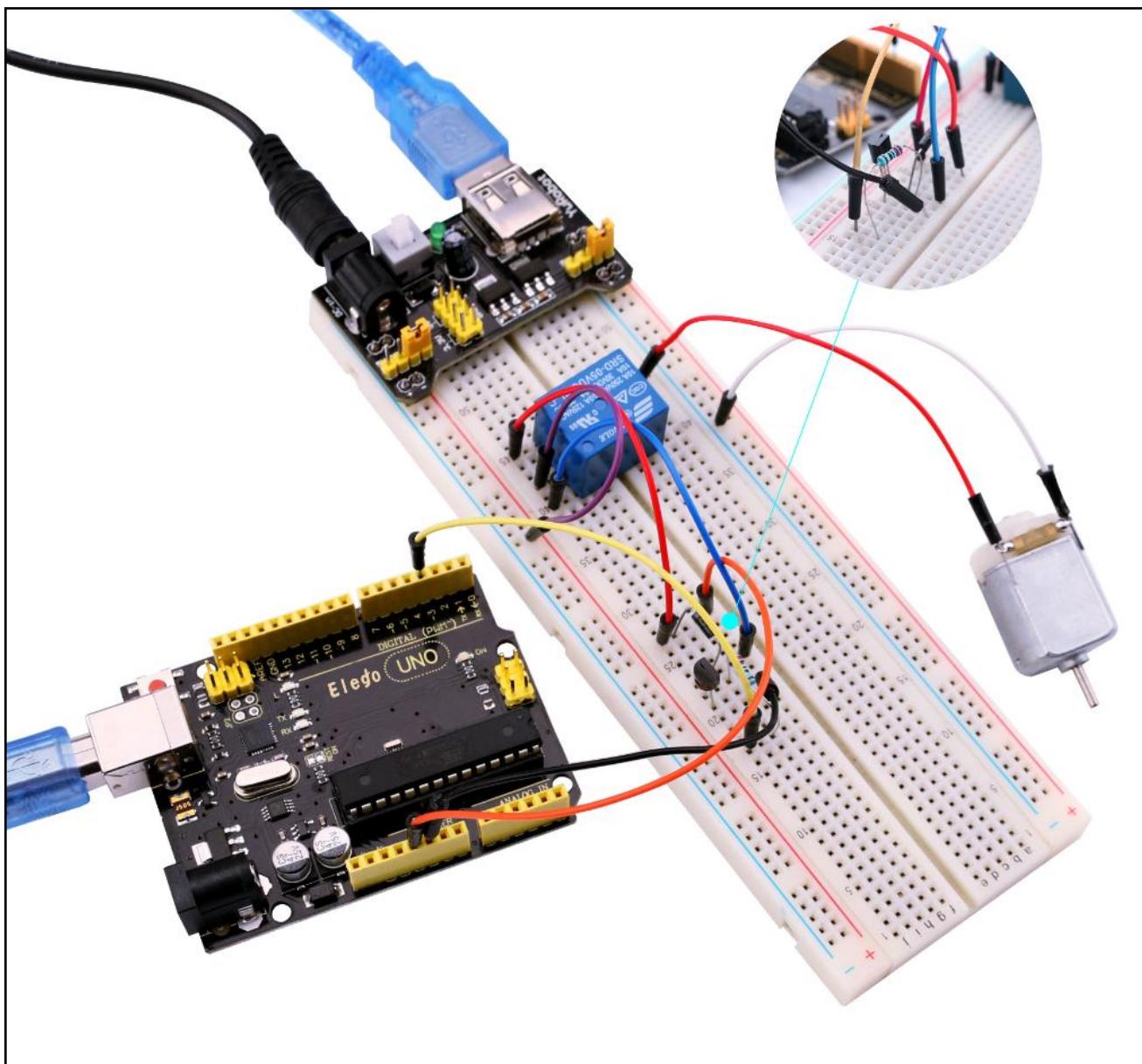
Connessione

Schema di collegamento



Codice

Vedere il file di codice.



Lezione 13: 74HC595 e Display Segmento

Generali

Dopo aver appreso Lezione 6 e Lezione 7, che useremo il registro 74HC595 spostamento per controllare la visualizzazione del segmento. Il display a segmenti mostrerà il numero 9-0.

Componenti necessari

(1) Elegoo UNO R3

(2) Breadboard

(1) 74HC595

(1) Esposizione di segmento

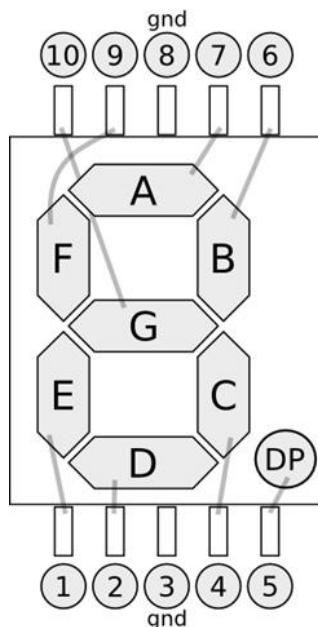
(8) 220 ohm resistore

(20) M-M fili

Introduzione di componente

Display a sette segmenti:

Di seguito è riportato il diagramma del perno a sette segmenti.

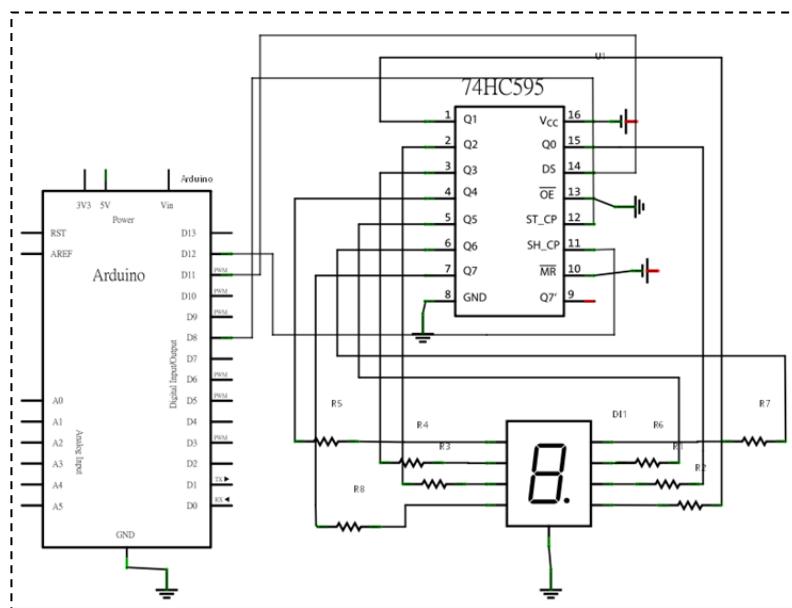


0-9 dieci cifre corrispondono a ciascun segmento sono i seguenti (la seguente tabella si applica catodo comune dispositivo di visualizzazione a sette segmenti, se si utilizza al anodo comune, il tavolo deve essere sostituito ogni 1 0 0 dovrebbero essere tutti sostituiti da 1):

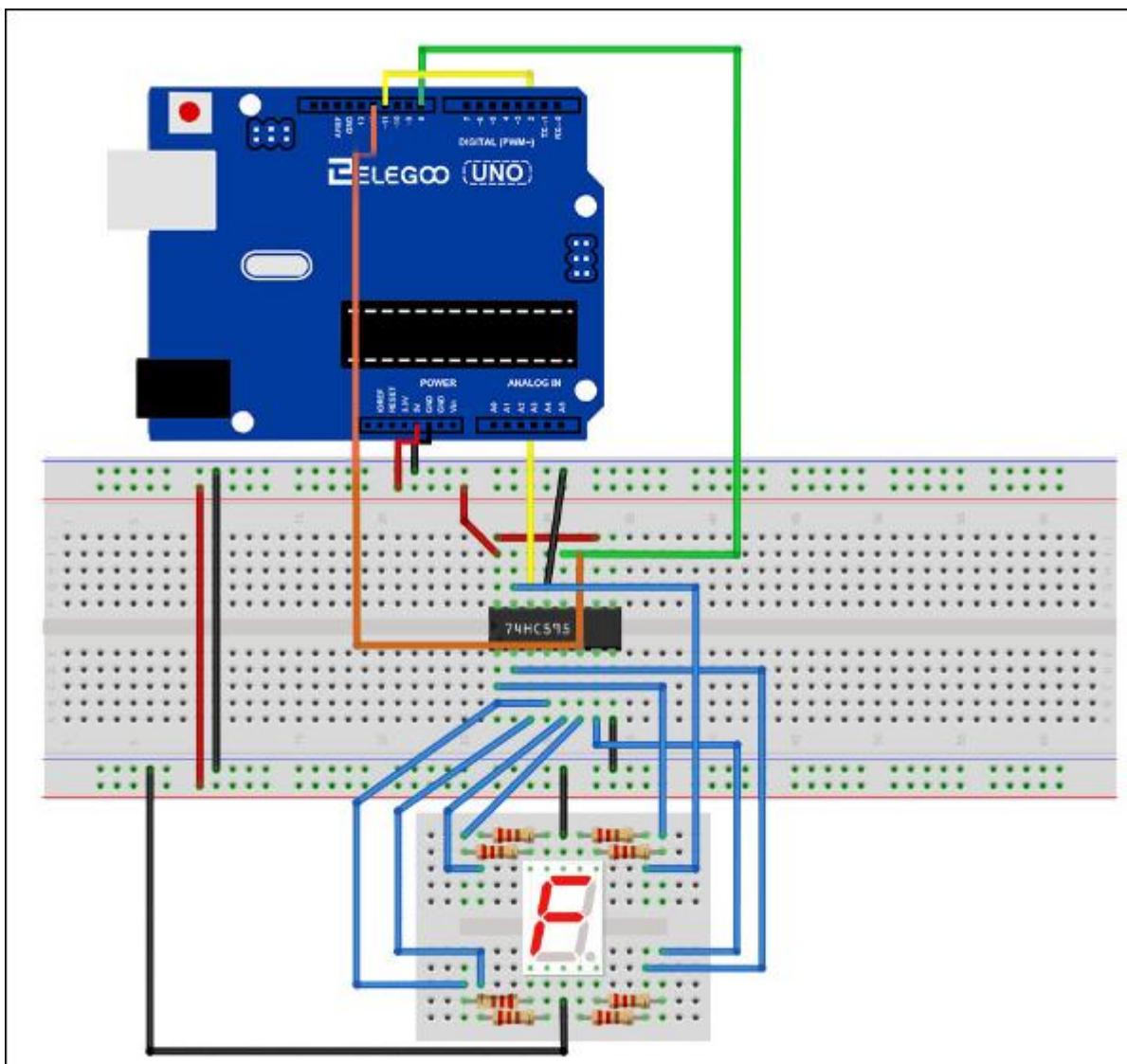
Display digitale	dp	a	b	c	d	e	f	g
0	0	1	1	1	1	1	1	0
1	0	0	1	1	0	0	0	0
2	0	1	1	0	1	1	0	1
3	0	1	1	1	1	0	0	1
4	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1
6	0	1	0	1	1	1	1	1
7	0	1	1	1	0	0	0	0
8	0	1	1	1	1	1	1	1
9	0	1	1	1	1	0	1	1

Connessione

Schematico



Schema di collegamento



La tabella seguente mostra il display 74HC595 del perno tabella di corrispondenza a sette segmenti:

74HC595 perno	Sette spettacoli notevole del perno di controllo (Corsa)
Q0	7 (A)
Q1	6 (B)
Q2	4 (C)
Q3	2 (D)

Q4	1 (E)
Q5	9 (F)
Q6	10 (G)
Q7	5 (DP)

Il primo passo: collegare 74HC595

Per prima, collegare il cablaggio per l'alimentazione e la terra:

- Vcc (perno 16) e MR (perno 10) collegato a 5V
- GND (perno 8) e OE (perno 13) per la terra

Collegare DS, ST_CP e SH_CP perni:

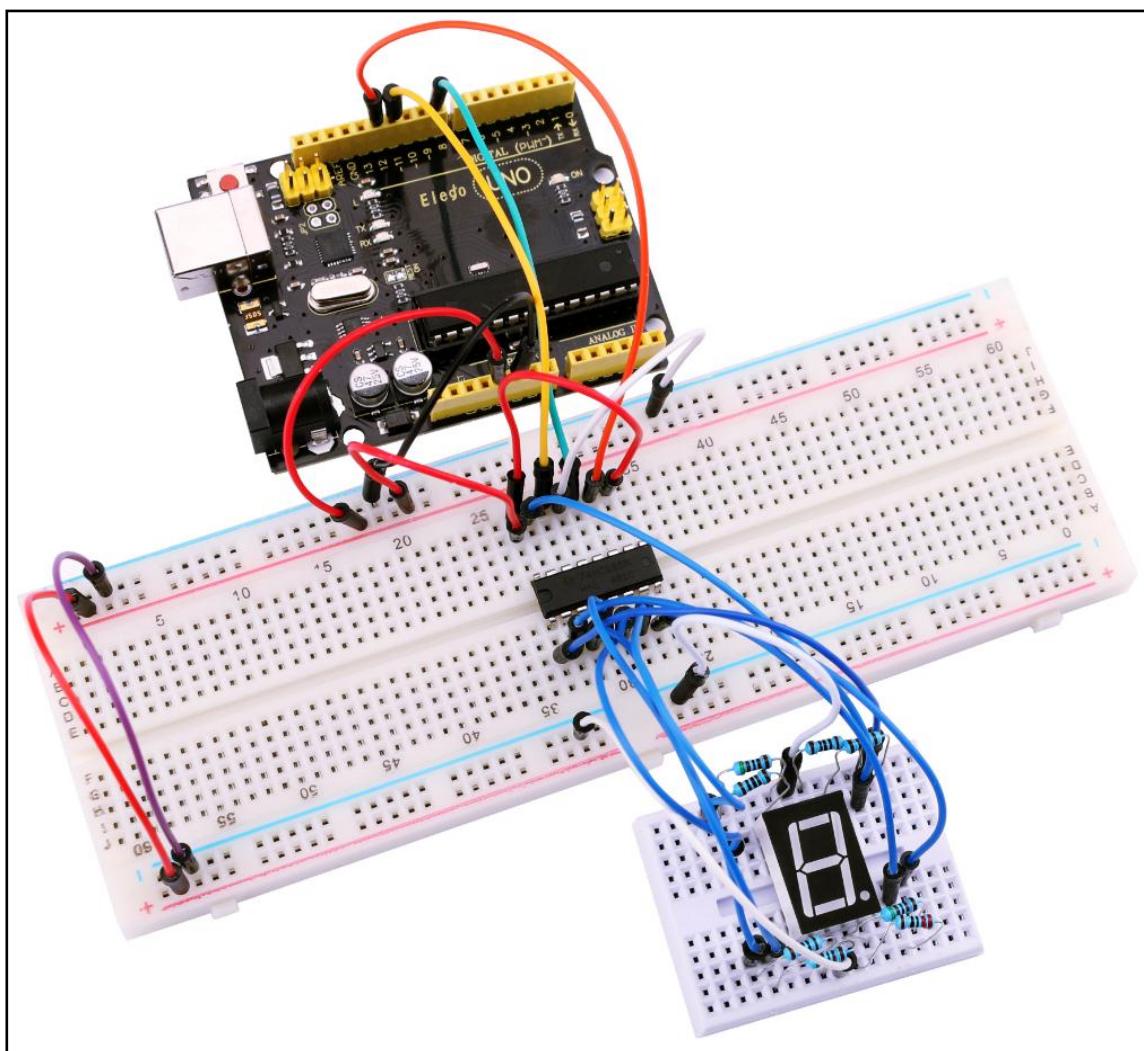
- DS (perno 14) collegato a UNO R3 board perni 11 (linea blu)
- ST_CP (perno 12, perno chiusura) collegato a UNO R3 board perni 8 (verde fili)
- SH_CP (perno 11, orologio perno) collegato a UNO R3 board perni 12 (giallo)

Passo Due: Collegare il display a sette segmenti

- Il display a sette segmenti 3, 8 pin di UNO R3 bordo GND (Questo esempio utilizza il catodo comune, se si utilizza il anodo comune, si prega di collegare il 3, 8 perni per UNO bordo R3 + 5V)
- Secondo la tabella sopra, collegare il Q0 74HC595 ~ Q7 a a sette segmenti Display del perno corrispondente (A ~ G e DP), e quindi ogni piede in una resistenza da 220 ohm in serie.

Codice

Vedere il file di codice.



Lezione 14: 4 -cifre -7-Display Segmento

Generali

In questa lezione, si apprenderà come utilizzare a 4 -cifre- 7-Segmento Display.

Quando si utilizza 1 -cifre -7 segmenti display ed è anodo comune, il perno anodo comune collega alla sorgente di alimentazione; se è catodo comune, il perno catodo comune collega al GND.

Quando si utilizza 4 -cifre -7-Segmento Display, l'anodo comune o pin catodo comune è utilizzato per controllare quali cifre visualizzato. Anche se vi è un solo lavoro cifre, il principio di Persistenza della visione vi permette di vedere tutti i numeri visualizzati perché ogni velocità di scansione è così veloce che quasi non si nota gli intervalli.

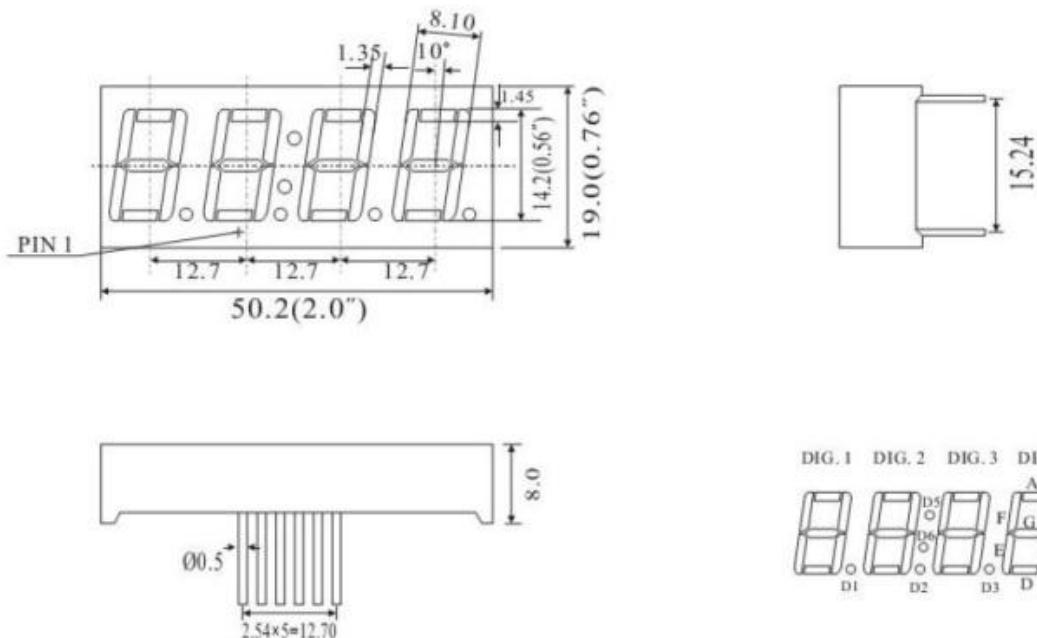
Componenti necessari

- (1) Elegoo UNO R3
- (1) Breadboard
- (1) 74HC595
- (1) 4 -cifre -7-Segmento Display
- (4) 220 ohm resistore
- (20) M-M fili

Introduzione di Componente

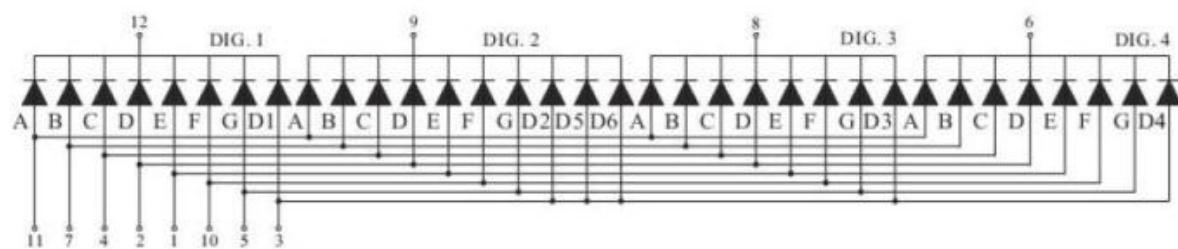
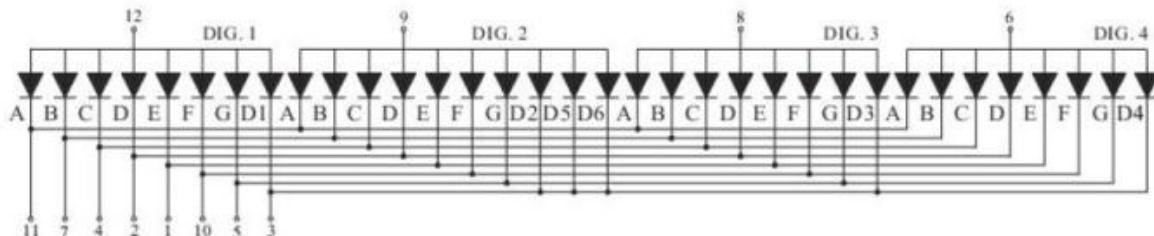
4 -cifre -7-Segmento Display:

Package Dimensions

CPS05643AB

UNIT: MM(INCH) TOLERANCE: $\pm 0.25(0.01")$

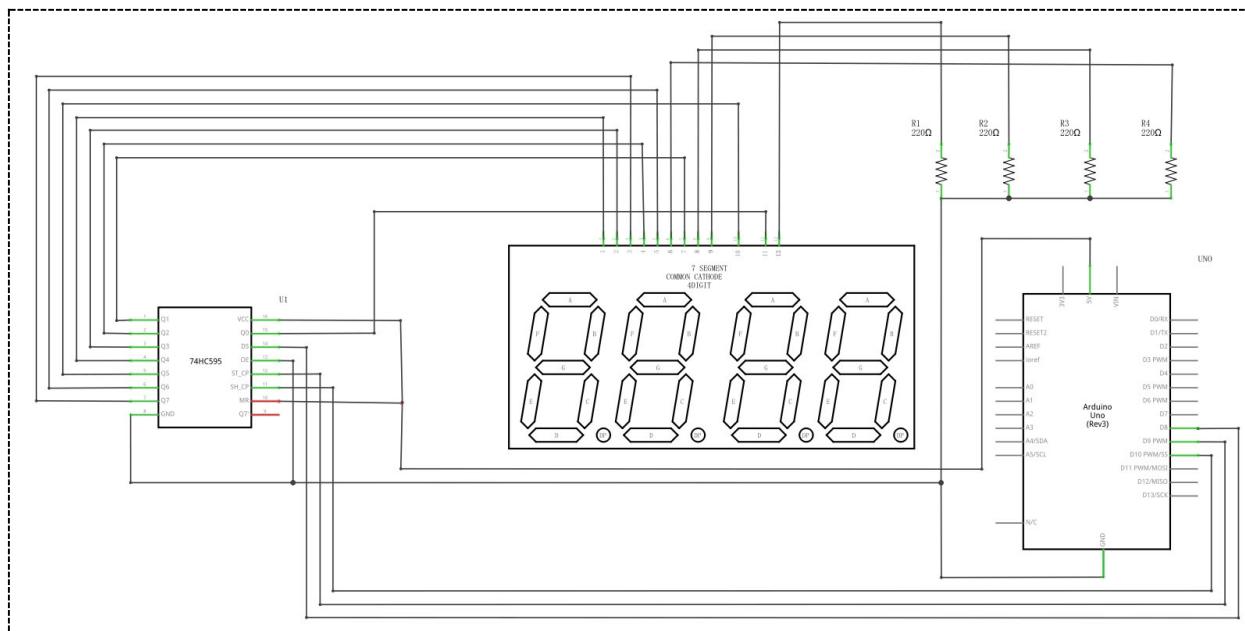
Internal Circuit Diagram

**5643A****5643B**

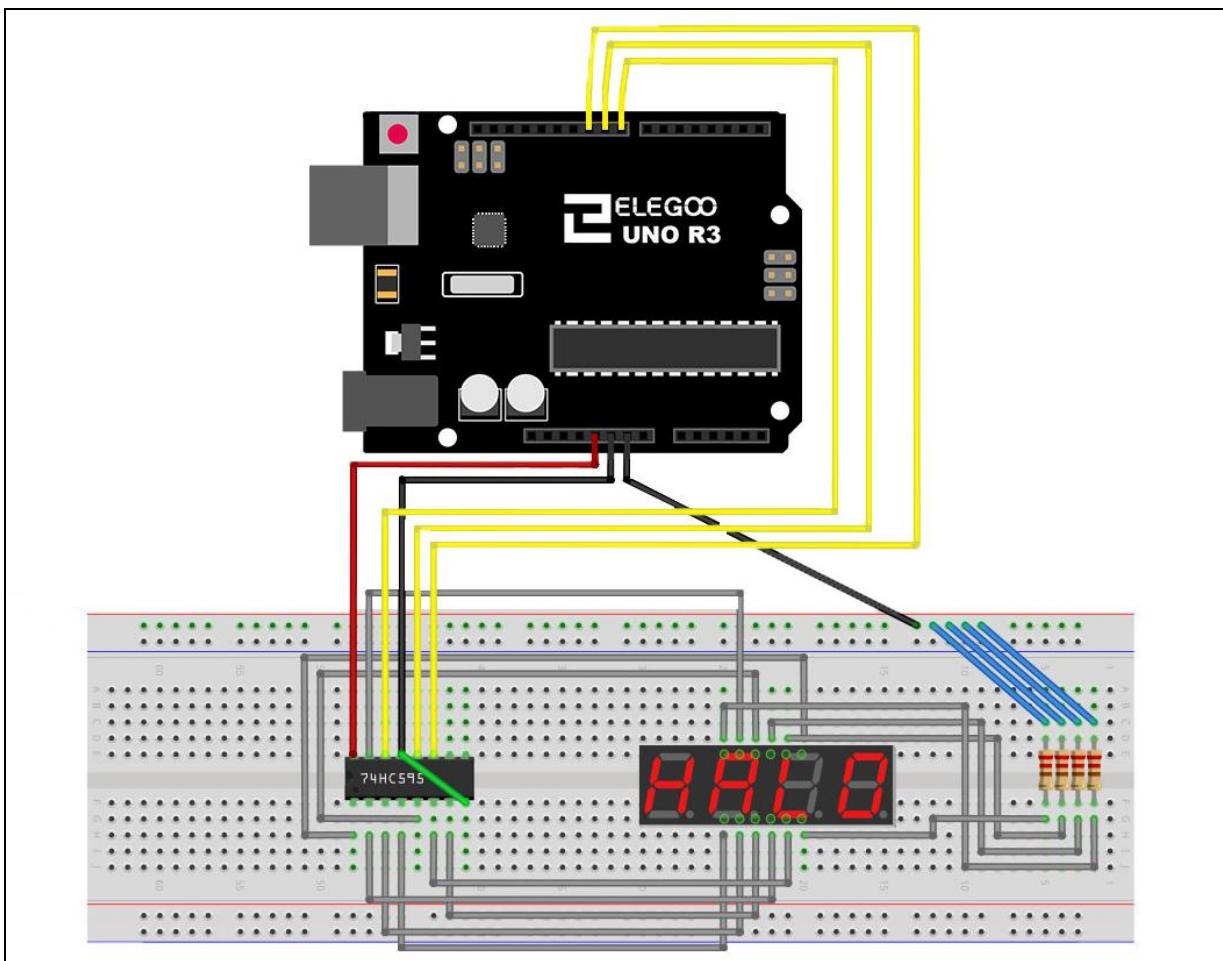
Four Digits Displays Series

Connessione

Schematico

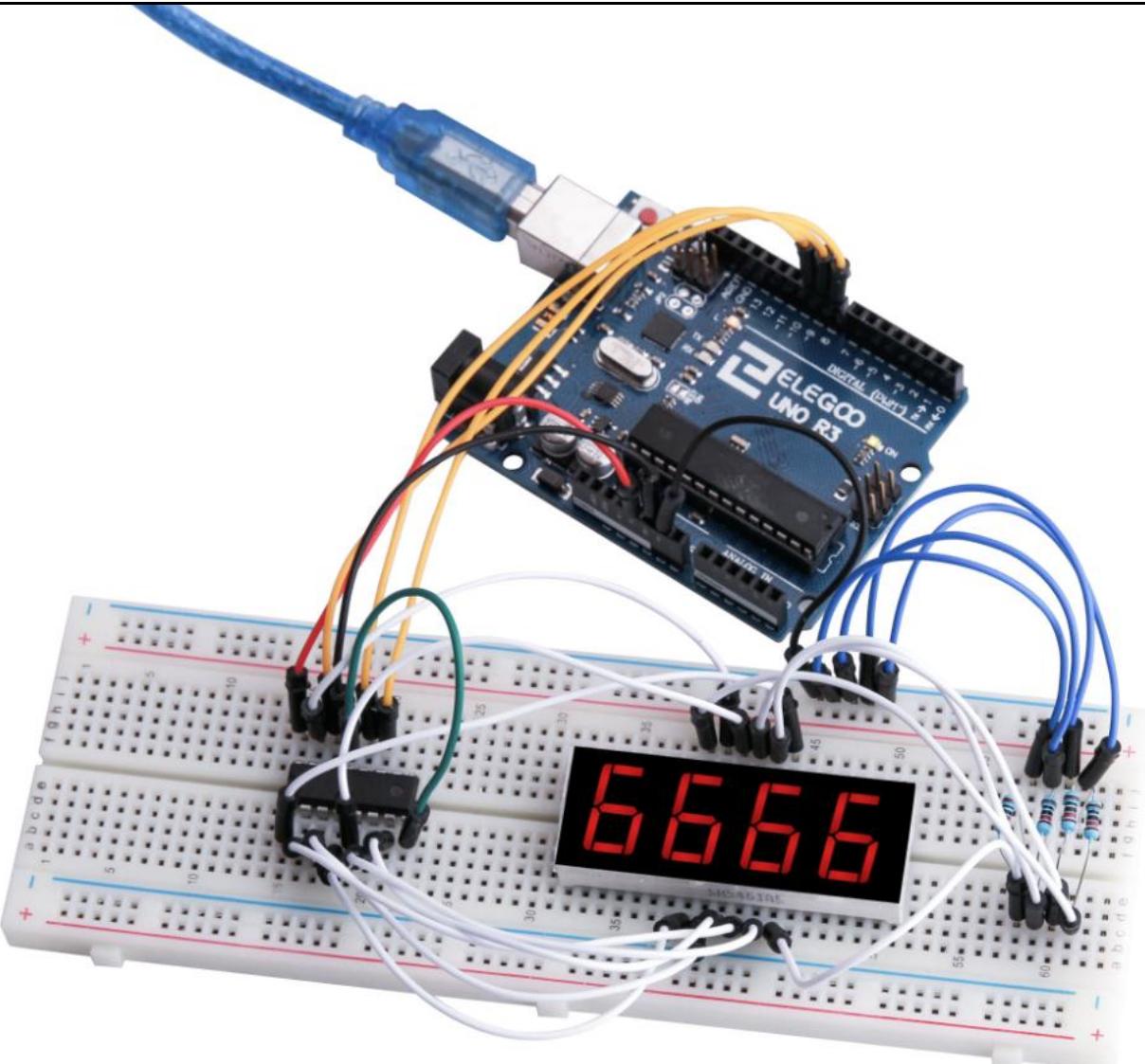


Schema di collegamento



Codice

Vedere il file di codice.



Lezione 15: Servo

Generali

Servo è un tipo di motoriduttore che può solo ruotare di 180 gradi. È controllato con l'invio di impulsi elettrici dal vostro bordo UNO R3. Questi impulsi indicano al servo quale posizione si dovrebbe passare a. Un servo ha tre fili, il filo marrone è GND, quello rosso è VCC, e quello arancione è linea di segnale.

Componenti necessari

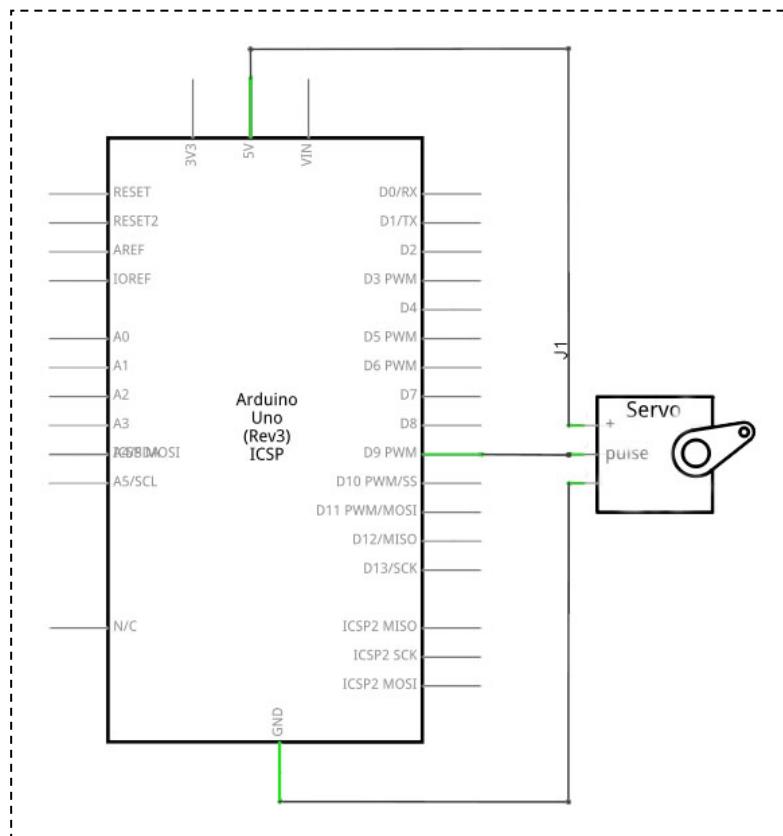
(1) Elegoo UNO R3

(1) Servo

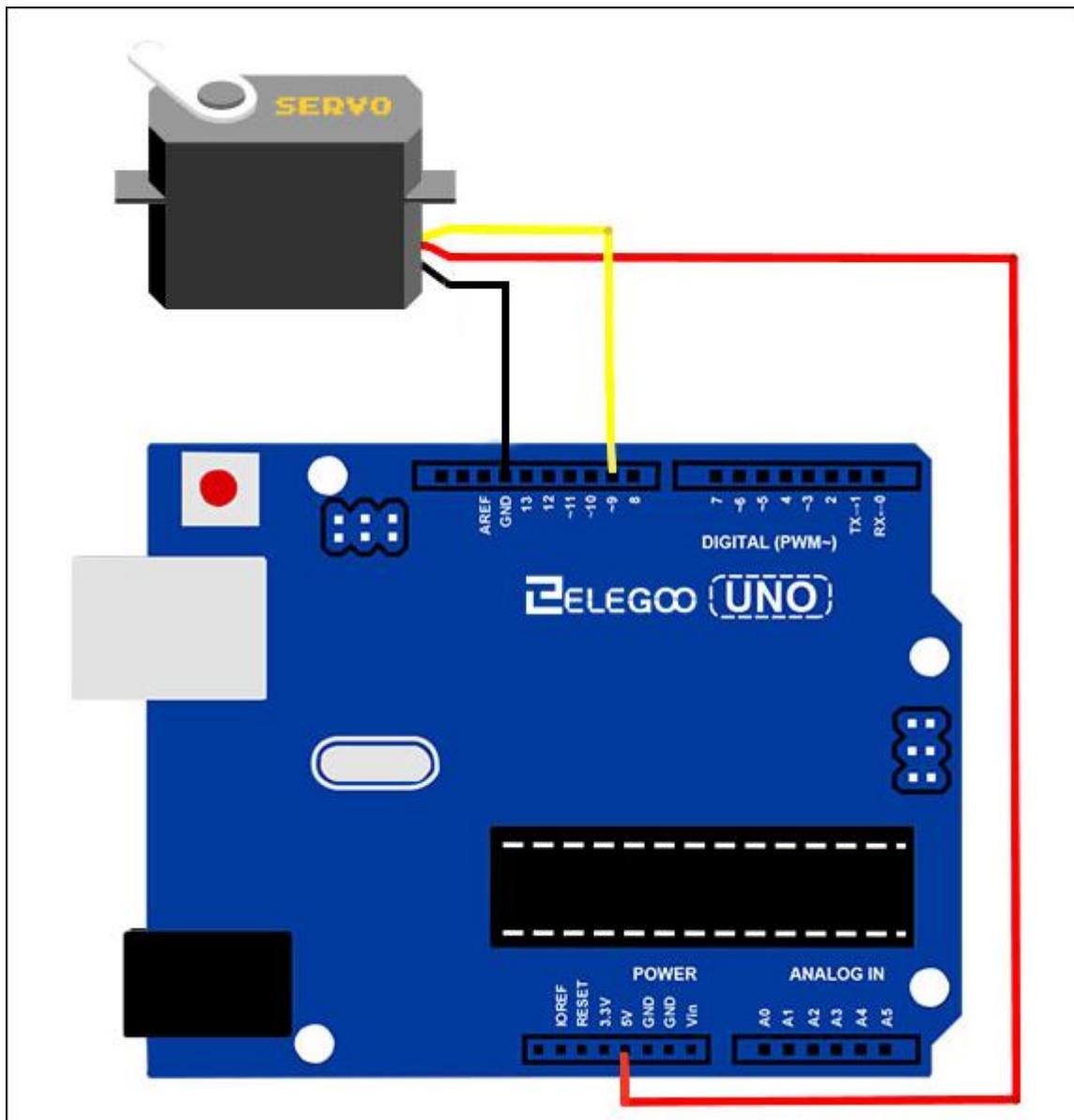
(2) F-M fili

Connessione

Schematico

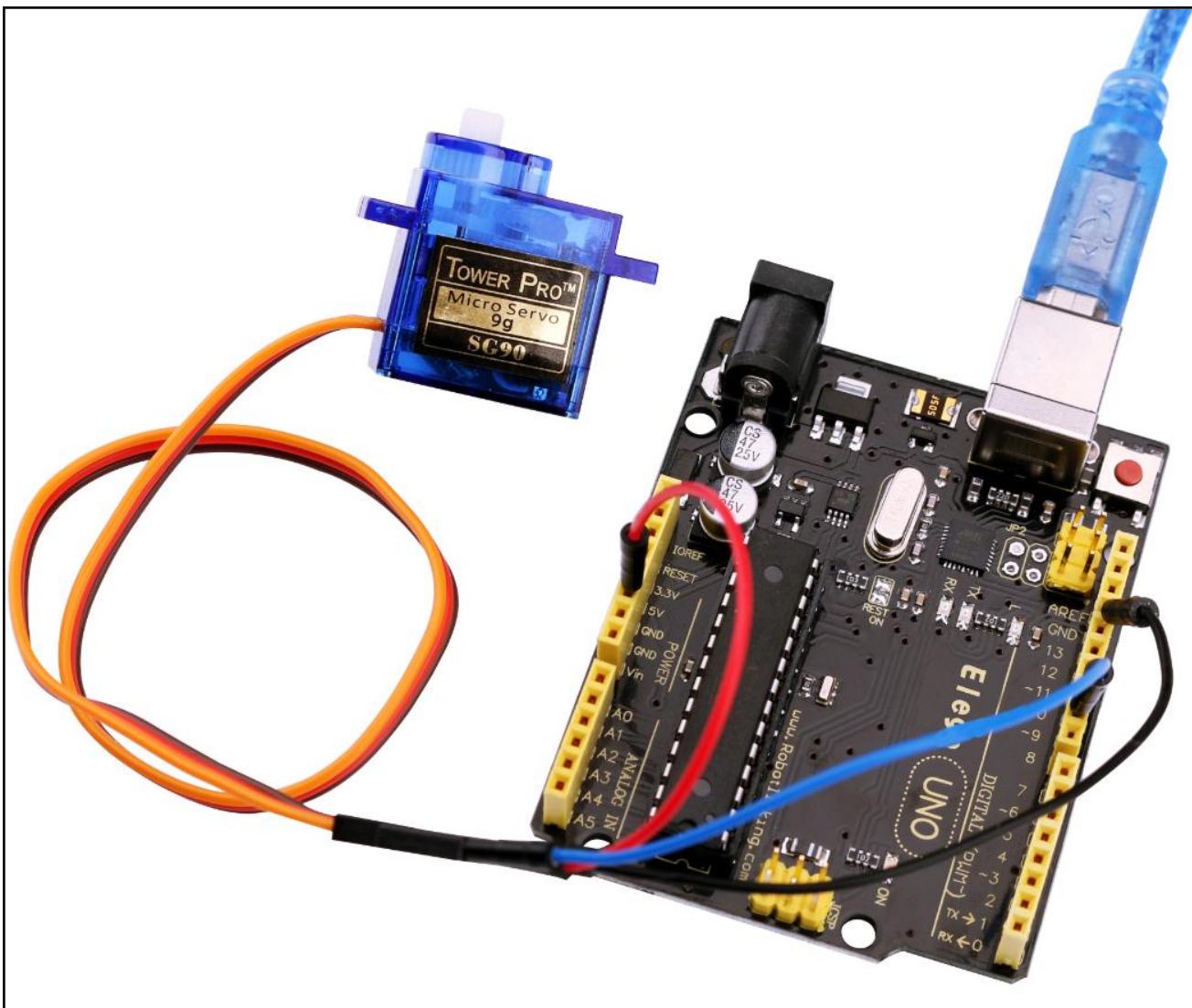


Schema di collegamento



Codice

Vedere il file di codice.



Lezione 16: LCD Display

Generali

In questa lezione si impara come collegare e utilizzare un display LCD alfanumerico.

Il display ha una retroilluminazione a LED e in grado di visualizzare due file con un massimo di 16 caratteri su ogni riga. Potete vedere i rettangoli per ogni personaggio sullo schermo ei pixel che compongono ogni personaggio. Il display è solo bianco su blu ed è destinato per mostrare il testo.

In questa lezione, ci verrà eseguito il programma di esempio Arduino per la libreria a cristalli liquidi, ma nella prossima lezione, avremo la nostra esposizione per mostrare il livello di temperatura e la luce, utilizzando i sensori.

Componenti necessari

- (1) Elegoo UNO R3
- (1) LCD1602 display
- (1) Potenziometro
- (1) Breadboard
- (16) F-M fili

Introduzione di Componente

LCD1602:

VSS: Un perno che collega a massa

VDD: Un perno che si connette a un + 5V alimentatore

VO: Un perno che regolare il contrasto del display di LCD 1602

RS: Un selezionato perno registro che controlla in cui nella memoria del display LCD si sta scrivendo i dati. È possibile selezionare il registro dei dati, che detiene ciò che accade sullo schermo, o un registro di istruzione, che è dove il controllore del display appare per le istruzioni su come procedere.

R/W: Un leggere/ scrittura perno che seleziona la modalità lettura o la scrittura del modello

E: Un perno consentendo che, a fonti di energia a basso livello, fa sì che il modulo di LDC di

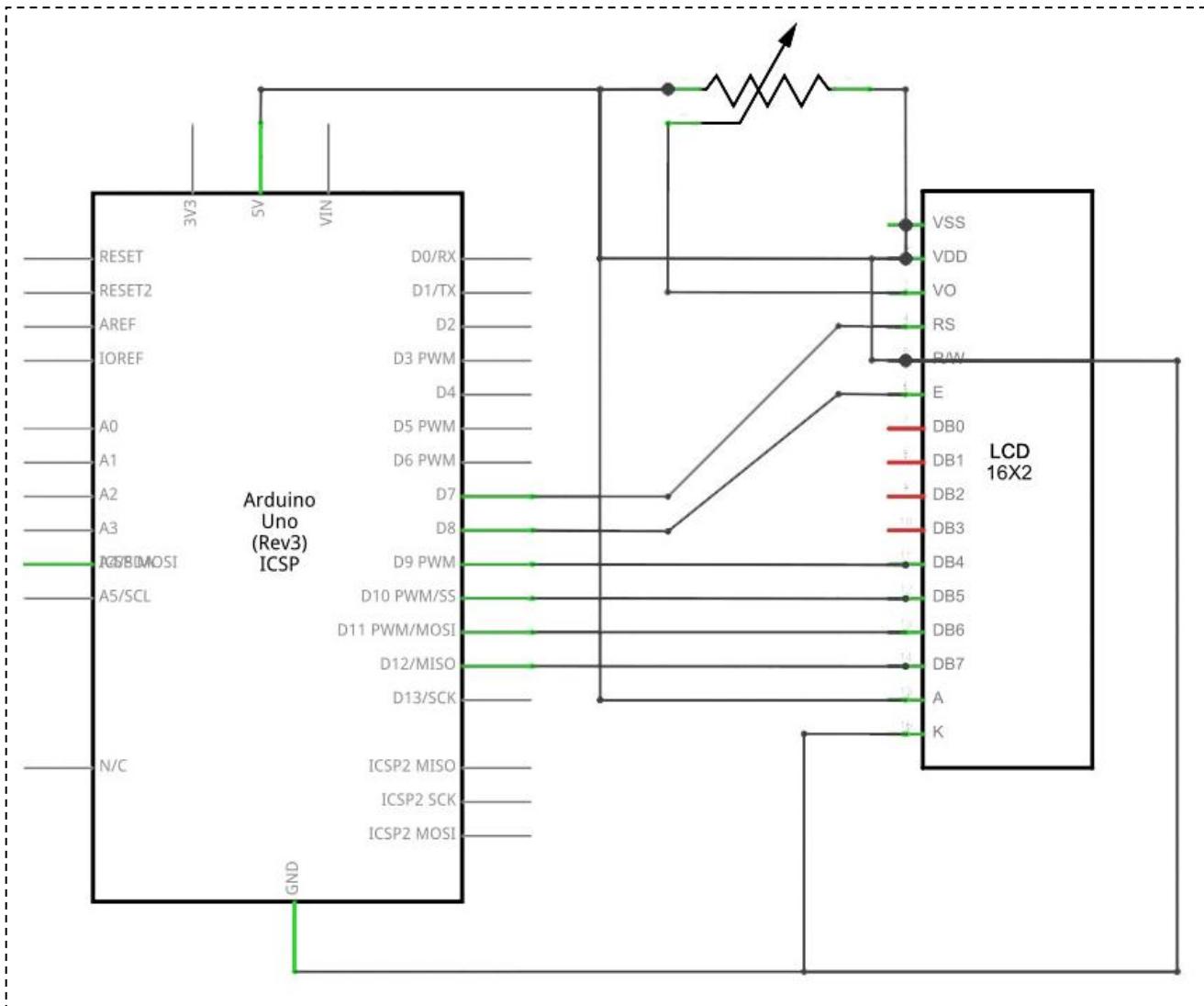
eseguire le relative istruzioni.

D0-D7: Penne che leggere e scrivere i dati

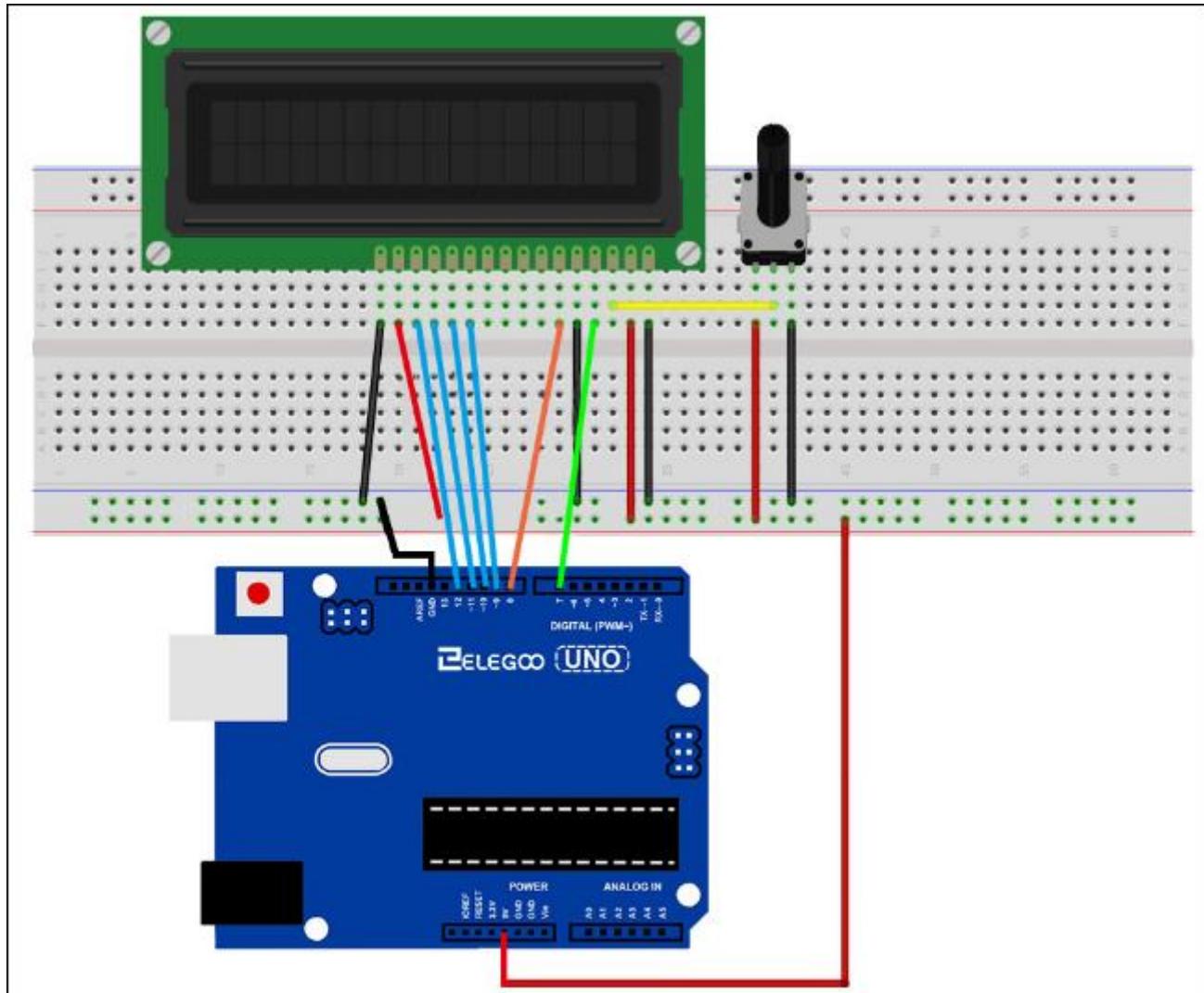
A e K: I perni che controllano la retroilluminazione a LED

Connessione

Schematico



Schema di collegamento



Il display LCD ha bisogno di sei perni Arduino, tutto pronto per essere uscite digitali. Ha anche bisogno di connessioni 5V e GND.

Ci sono un certo numero di connessioni da effettuare. Allineando il display con la parte superiore della basetta aiuta ad identificare i suoi piedini senza troppo contare, soprattutto se la basetta ha le sue righe numerate con la riga 1 come la riga superiore della scheda. Da non dimenticare, il lungo cavo giallo che collega il dispositivo di scorrimento del piatto al pin 3 del display. Il 'pot' è utilizzato per controllare il contrasto del display.

È possibile che il display è fornito senza perni di intestazione collegato ad esso. In tal caso, seguire le istruzioni nella sezione successiva.

Codice

Ora che avete la configurazione fisica, tutti abbiamo bisogno ora è il codice.

Prima di poter eseguire questo, assicurarsi di aver installato il <LiquidCrystal> libreria o riinstallarlo, se necessario. In caso contrario, il codice non funzionerà

Questo esempio utilizza diverse penne a quelle che utilizziamo, in modo da trovare la riga di codice qui sotto:

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

e modificarla in:

LiquidCrystal lcd(7, 8, 9, 10, 11, 12);

Carica il codice per la vostra scheda Arduino e si dovrebbe vedere il messaggio 'ciao, mondo' visualizzato, seguita da un numero che conta fino da zero.

La prima cosa da notare nel disegno è la linea:

#incluso <LiquidCrystal.h>

Questo dice Arduino che vogliamo utilizzare la libreria di cristallo liquido.

Successiva è la linea che dobbiamo modificare. Questo definisce quali prodotti morbidi e Arduino devono essere collegati al quale prodotto è morbido e visualizzazione.

LiquidCrystal lcd(7, 8, 9, 10, 11, 12);

Gli argomenti di queste aree segue:

Display Perno Nome Display Perno Numero Arduino Perno (in questo esempio) RS 4 7 E 6 8 D4
11 9 D5 12 10 D6 13 11 D7 14 12

Dopo aver caricato il codice, accertarsi che la retroilluminazione è illuminato, e regolare il potenziometro tutta la strada fino a quando viene visualizzato il messaggio di testo.

Nella funzione 'setup', abbiamo due comandi:

lcd.inizio(16, 2);

lcd.stampare("Ciao mondo!");

Il primo racconta la libreria a cristalli liquidi il numero di colonne e righe il display ha. La seconda riga visualizza il messaggio che vediamo sulla prima riga della schermata.

Nella funzione 'loop', abbiamo anche due comandi:

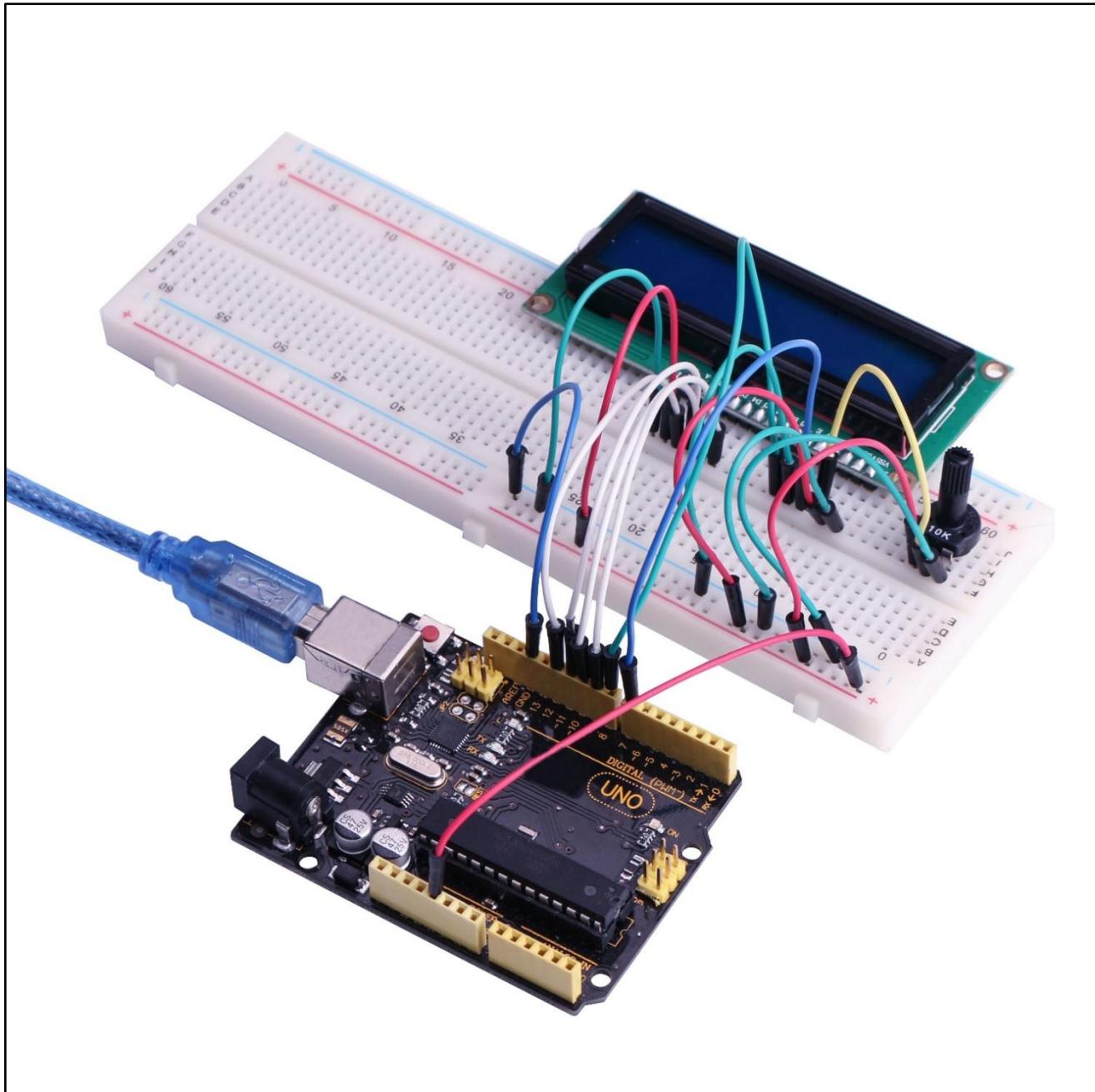
lcd.setCursor(0, 1);

Lcd. stampare (millis ()/1000);

La prima imposta la posizione del cursore (dove appare il testo accanto) a colonna 0 e riga 1.

Sia colonna e numeri di riga partono da 0 invece che 1.

La seconda riga viene visualizzato il numero di millisecondi dal momento che la Arduino è stato azzerato.



Lezione 17: Termometro

Generali

In questa lezione, si utilizzerà un display LCD per mostrare la temperatura.

Componenti necessari

(1) Elegoo UNO R3

(1) LCD1602 display

(1) 10k ohm resistore

(1) Termometro

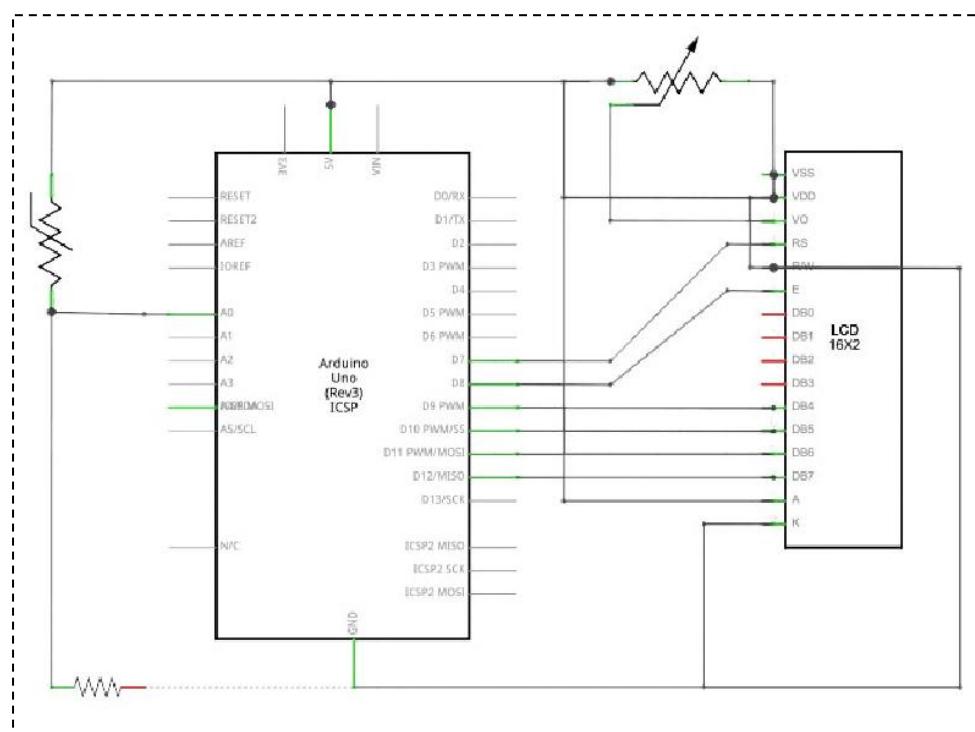
(1) Potenziometro

(1) Breadboard

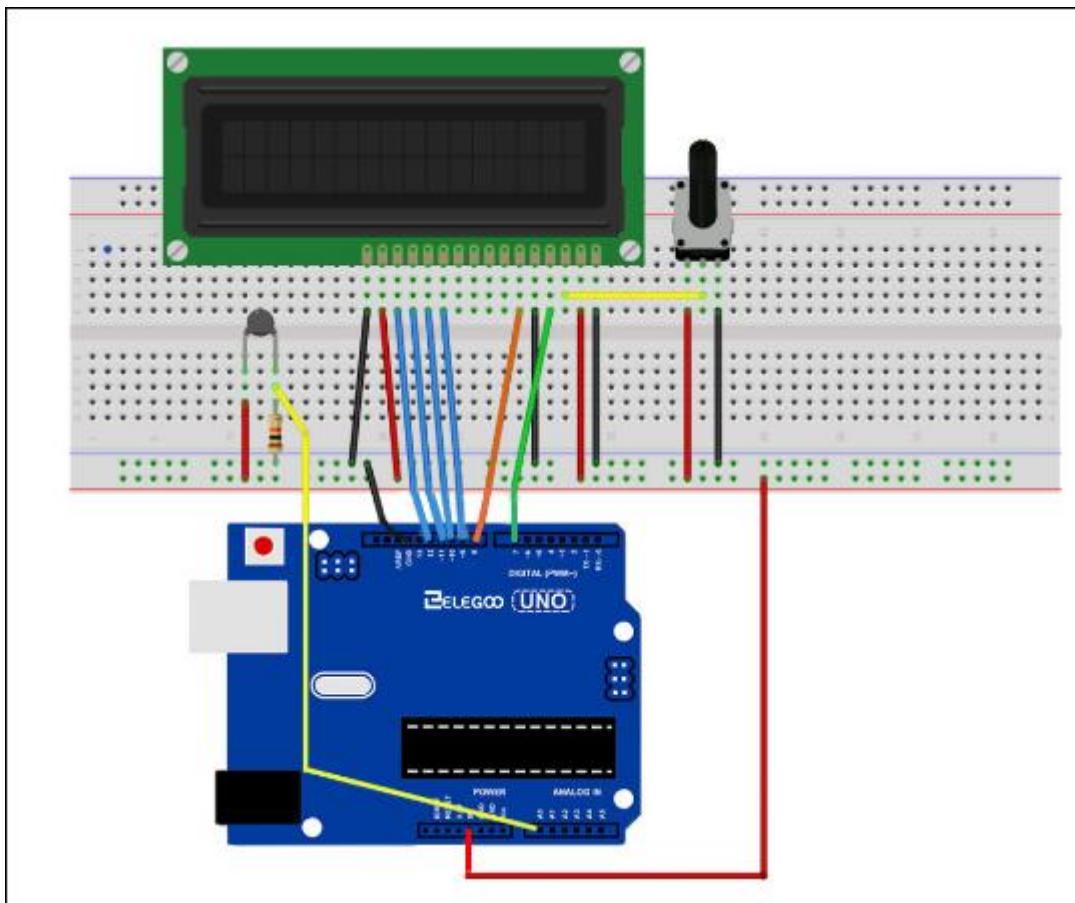
(16) F-M fili

Connessione

Schematico



Schema di collegamento



Il layout tagliere si basa sul layout dalla Lezione 16, in modo che semplificare le cose se avete ancora presente sul tagliere.

Ci sono alcuni ponticelli vicino il piatto che sono state spostate leggermente su questo layout.
La fotocellula, resistore di $1\text{ k}\Omega$ e termistore sono tutte nuove aggiunte alla scheda.

Codice

Prima di poter eseguire questo, assicurarsi di aver installato il <LiquidCrystal> biblioteca o ri-installarlo, se necessario. In caso contrario, il codice non funzionerà.

Lo schizzo per questo si basa su quello della lezione 14. caricarlo sul vostro Arduino e si dovrebbe trovare che il riscaldamento del sensore di temperatura mettendo il dito su di esso aumenterà la lettura della temperatura.

Trovo utile per mettere una riga di commento sopra il comando 'lcd'.

```
//           BS  E  D4 D5  D6 D7
```

```
LiquidCrystal lcd(7, 8, 9, 10, 11, 12);
```

Questo rende le cose più facili, se si decide di cambiare che i perni si utilizza.

Nella funzione 'loop' ora ci sono due cose interessanti in corso. In primo luogo dobbiamo convertire l'analogico dal sensore di temperatura in una temperatura effettiva, e in secondo luogo dobbiamo capire come visualizzarli.

Prima di tutto, diamo un'occhiata al calcolo della temperatura.

```
int tempLettura = analogLettura(tempPin);
float tempVolte = tempLettura * 5.0 / 1024.0;
float tempC = (tempVolte - 0.5) * 100.0;
float tempF = tempC * 9.0 / 5.0 + 32.0;
```

La lettura RAW dal sensore di temperatura viene prima moltiplicato per 5 e poi diviso per 1024 per darci la tensione (tra 0 e 5) in ingresso analogico 'tempPin'.

Per convertire la tensione proveniente dal TMP36 in una temperatura in gradi C, si deve sottrarre 0.5V dalla misura e poi moltiplicare per 100.

Per convertire questo in una temperatura in gradi Fahrenheit, è quindi necessario moltiplicare per 9/5 e quindi aggiungere 32.

Visualizzazione letture cambiano su un display LCD può essere difficile. Il problema principale è che la lettura non può essere sempre lo stesso numero di cifre. Quindi, se la temperatura passa da 101.50 al 99.00 quindi la cifra supplementare dal vecchio lettura rischia di rimanere sul display.

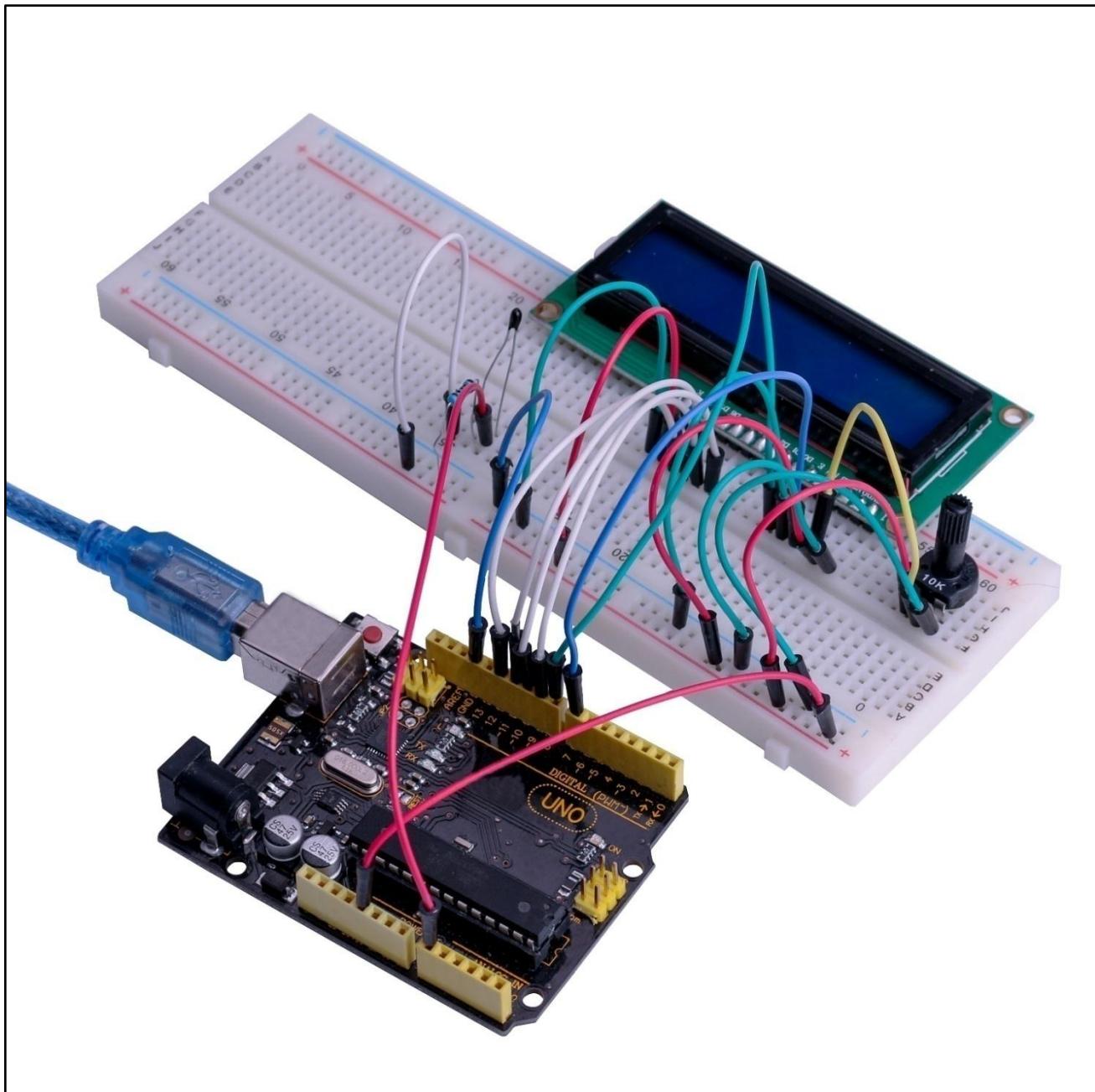
Per evitare questo, scrivere l'intera riga del display LCD ogni iterazione del ciclo.

```
// -----
lcd.stampare("Temp      F ");
lcd.setCursor(6, 0);
lcd.stampare(tempF);
```

La piuttosto strano commento serve per ricordare le 16 colonne del display. È quindi possibile stampare una stringa di lunghezza che con gli spazi in cui la lettura effettiva andrà.

Per riempire gli spazi vuoti, impostare la posizione del cursore per cui dovrebbe comparire la lettura e quindi stamparlo.

Esattamente lo stesso approccio viene utilizzato per la visualizzazione del livello di luce. Non ci sono unità per il livello di luce, abbiamo appena visualizzare la lettura grezza dalla lettura analogica.



Lezione 18: Modulo sensore ad ultrasuoni

Generali

sensore ad ultrasuoni sono l'ideale per tutti i tipi di progetti che necessitano di misure di distanza, evitando gli ostacoli come esempi.

L'HC-SR04 sono poco costoso e facile da usare dato che useremo una libreria specificamente progettato per questi sensori.

Componenti necessari

- (1) Elegoo UNO R3
- (1) Modulo sensore ad ultrasuoni
- (4) F-M fili

Introduzione di Componente

Sensore ad ultrasuoni

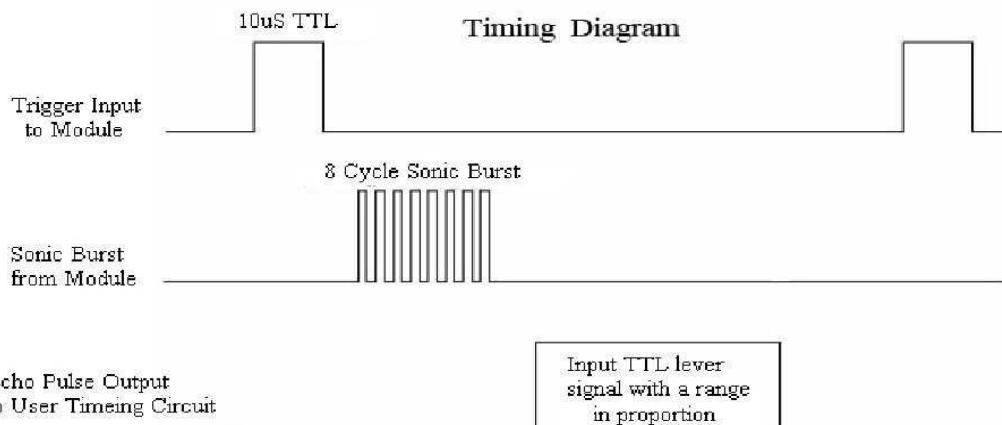
Ultrasuoni che vanno modulo HC - SR04 fornisce due centimetri - Funzione 400cmnon-contactmeasurement, la precisione che vanno può arrivare a 3 mm. I moduli comprende trasmettitori a ultrasuoni, ricevitore e circuito di controllo. Il principio di base del lavoro:

- (1) Utilizzo IO trigger per segnale a livello alto almeno 10us,
- (2) Il modulo invia automaticamente otto 40 kHz e rileva se vi è un segnale di impulso indietro.
- (3) Se il segnale posteriore, attraverso di alto livello, il tempo di durata ad alto rendimento IO è il momento di inviare ad ultrasuoni strappato girando.

Prova della distanza = (tempo di alto livello × velocità del suono (340M / S) / 2

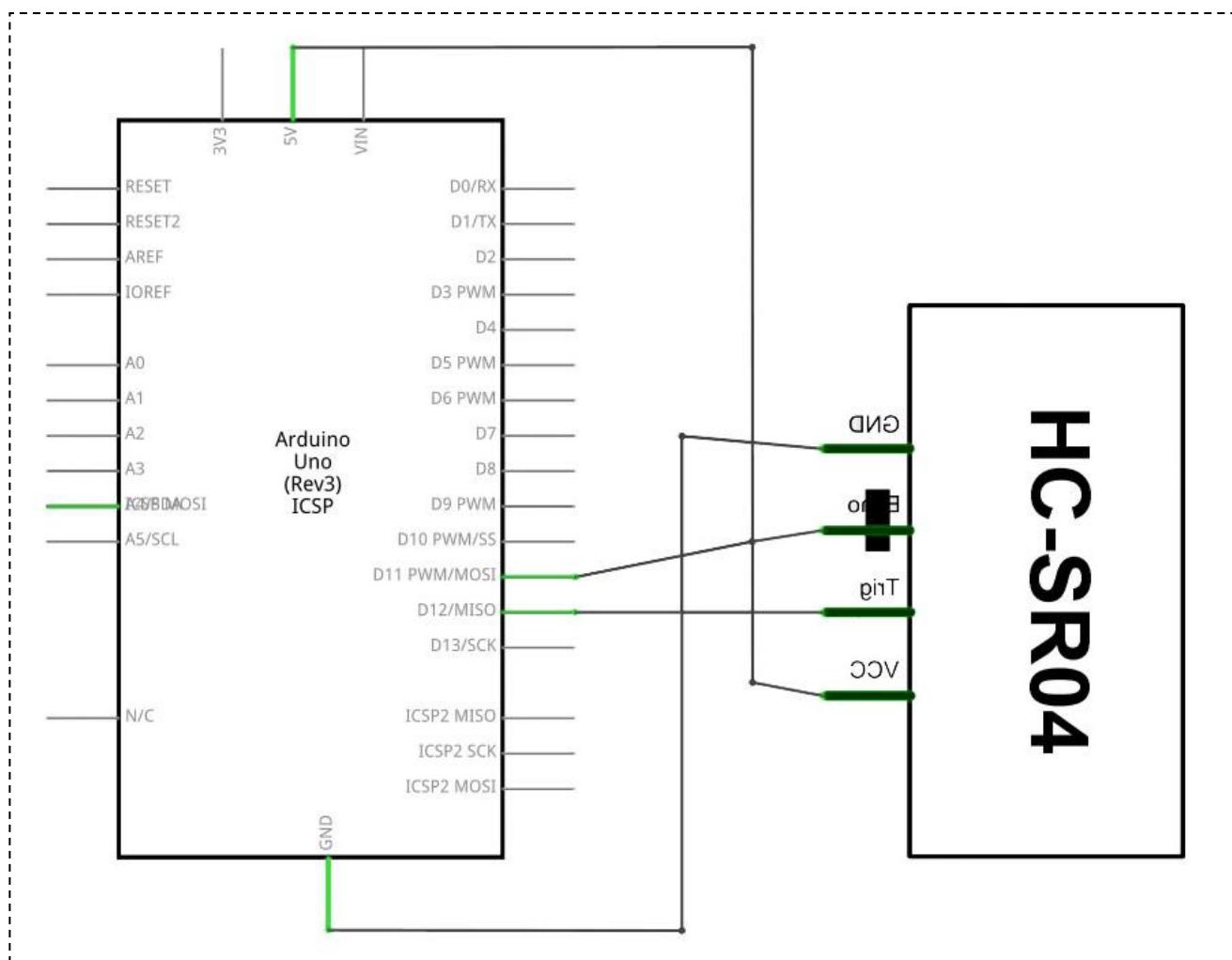
Diagramma dei tempi è mostrato di seguito. Hai solo bisogno di fornire una short10uSpulse per l'ingresso trigger per avviare il respiro, e quindi il modulo inviare una raffica 8 ciclo di ultrasuoni a 40 kHz ed aumentare la sua eco. All'Echo è un oggetto distanza che è la larghezza di impulso e l'intervallo in proporzione .You può calcolare la portata attraverso l'intervallo di tempo tra l'invio del segnale di innesco e la ricezione del segnale di eco. Formula:
 $US / 58 = \text{centimetri} \quad o \quad \text{siamo} / 148 = \text{pollici}; \quad o: \text{la velocità range} = \text{alto livello tempo} * (340M / S)$

/ 2; si consiglia di utilizzare ciclo over60msmeasurement, al fine di evitare che il segnale di attivazione al segnale di echo.

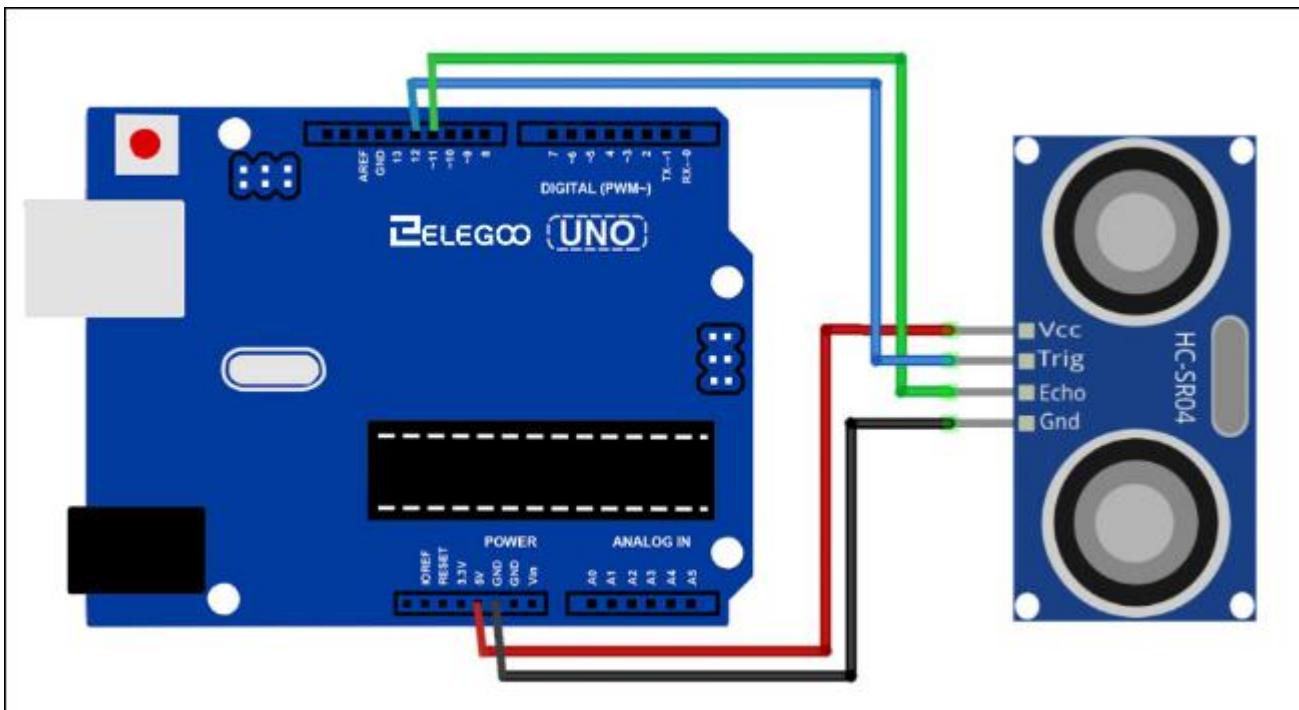


Connessione

Schematico



Schema di collegamento



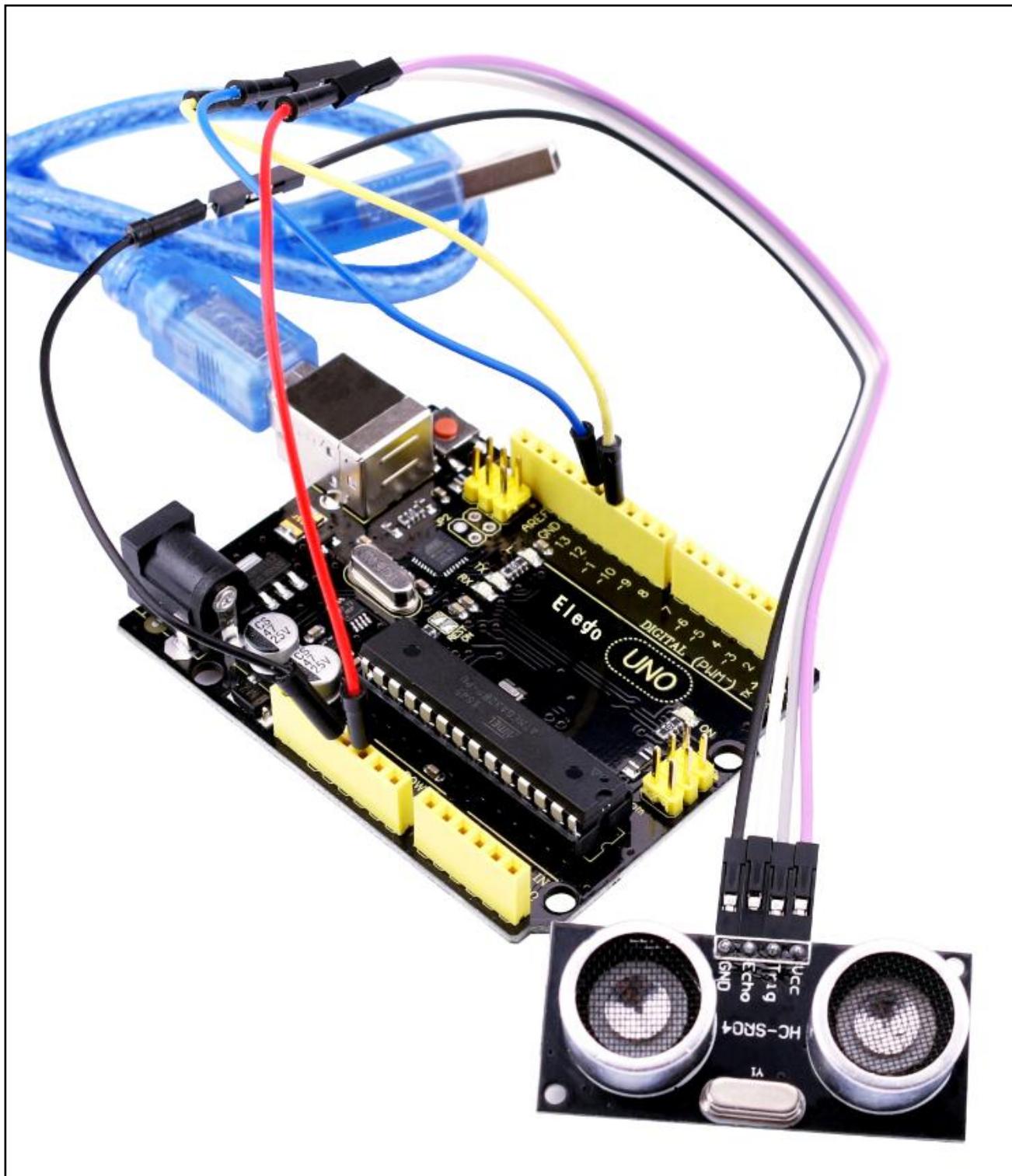
Codice

Utilizzando una libreria progettata per questi sensori farà il nostro codice breve e semplice.

Includiamo la libreria all'inizio del nostro codice, e quindi utilizzando semplici comandi possiamo controllare il comportamento del sensore.

Ora che abbiamo la configurazione fisica, tutti abbiamo bisogno ora è il codice.

Prima di poter eseguire questo, è necessario assicurarsi di avere installare il <HC-SR04_Library> biblioteca. Oppure è necessario installare di nuovo. Se non si esegue questa operazione, il codice non funzionerà.



Lezione 19: Temperatura e umidità di sensore di DHT11

Generali

In questa esercitazione impareremo come usare un DHT11 temperatura e umidità di sensore.

È abbastanza preciso per la maggior parte dei progetti che hanno bisogno di tenere traccia delle letture di umidità e temperatura.

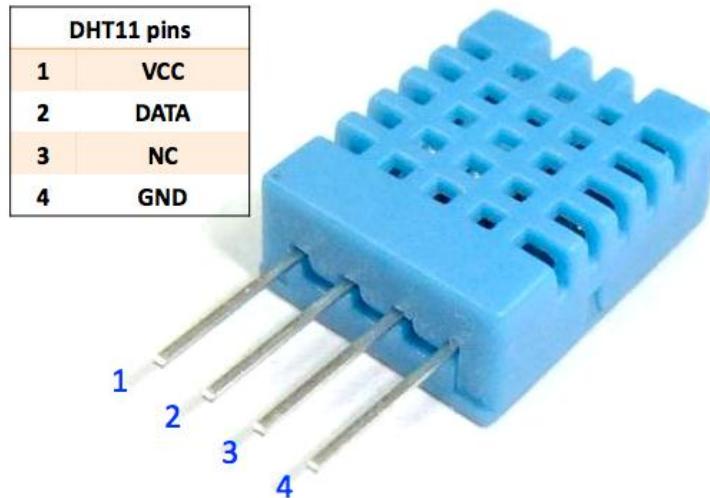
Anche in questo caso useremo una libreria appositamente per questi sensori che renderanno il nostro codice breve e facile da scrivere.

Componenti necessari

- (1) Elegoo UNO R3
- (1) DHT11 modulo
- (3) F-M fili

Introduzione di Componente

Temperatura e umidità di sensore:



Temperatura e umidità di sensore di DHT11 è un sensore composito contiene un segnale di uscita digitale calibrata della temperatura e dell'umidità. Applicazione di una unità digitali tecnologia di raccolta dedicato e la tecnologia di rilevamento di temperatura e umidità, per garantire che il prodotto ha alta affidabilità ed eccellente stabilità a lungo termine. Il sensore

comprende un senso resistivo di componenti bagnati e un NTC dispositivo di misurazione della temperatura, e collegati con un alto rendimento microcontrollore a 8 bit.

Applicazioni: HVAC, deumidificatore, di prova e di apparecchiature di controllo, dei beni di consumo, automotive, controllo automatico, data logger, stazioni meteo, elettrodomestici, regolatore di umidità, di misurazione dell'umidità medico e l'altro e di controllo.

Parametri di prodotti

Umidità relativa:

Risoluzione: 16Bit

Ripetibilità: $\pm 1\%$ RH

Precisione: At 25°C $\pm 5\%$ RH

Intercambiabilità: completamente intercambiabili

Tempo di risposta: 1 / e (63%) di 25°C 6s

1m / s aria 6s

Hysteresis: $<\pm 0.3\%$ RH

Stabilità a lungo termine: $<\pm 0.5\%$ RH / yr in

Temperatura:

Risoluzione: 16Bit

Ripetibilità: $\pm 0.2^\circ\text{C}$

Variabile: At 25°C $\pm 2^\circ\text{C}$

Tempo di risposta: 1 / e (63%) 10S

Caratteristiche Elettriche

Alimentazione elettrica: DC 3.5~5.5V

Corrente di alimentazione misura 0.3mA attesa 60 μ A

Periodo di campionamento: più di 2 secondi

Descrizione di perno:

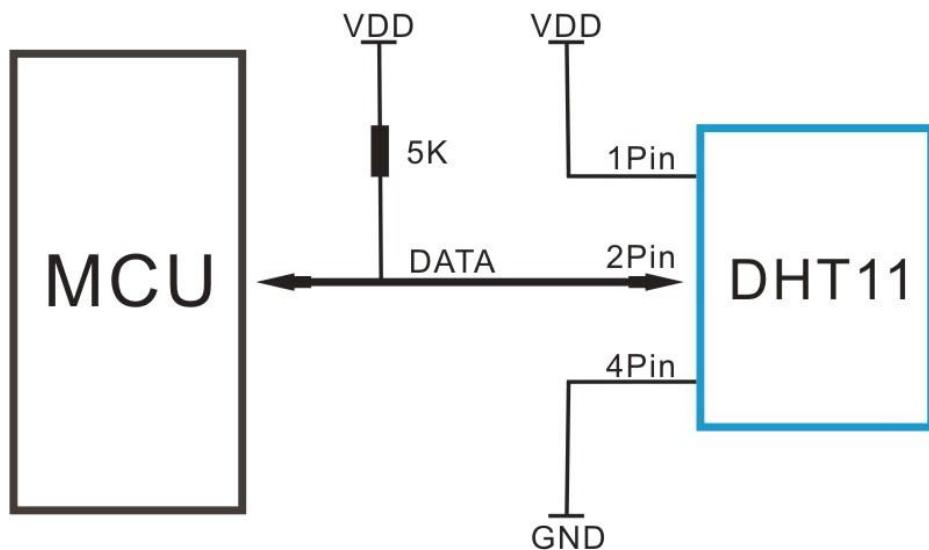
1, Alimentazione VDD 3.5~5.5V DC

2, dati seriale DATA, un singolo bus

3, NC, perno vuoto

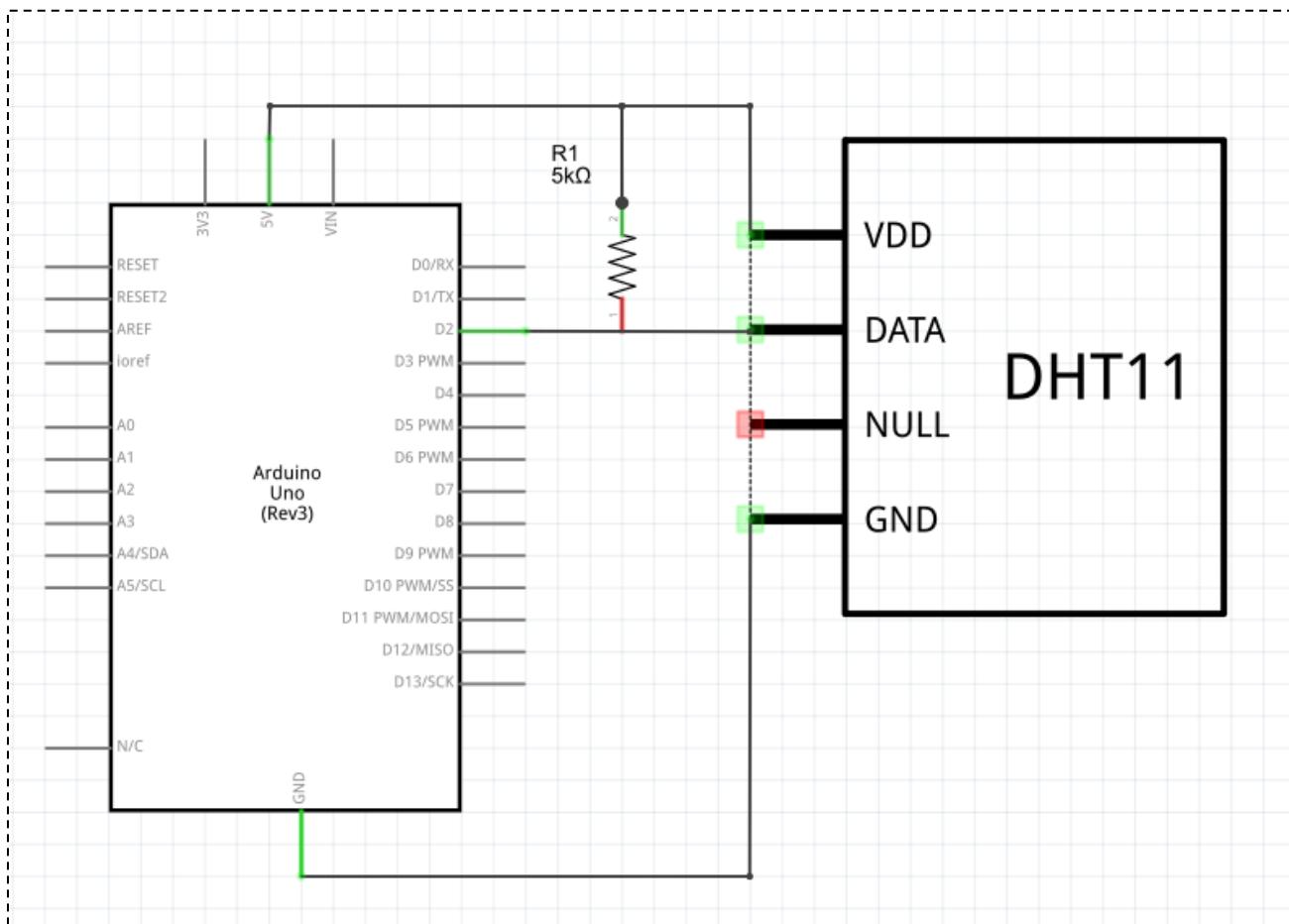
4, GND terra, il potere negativo

Applicazione tipica

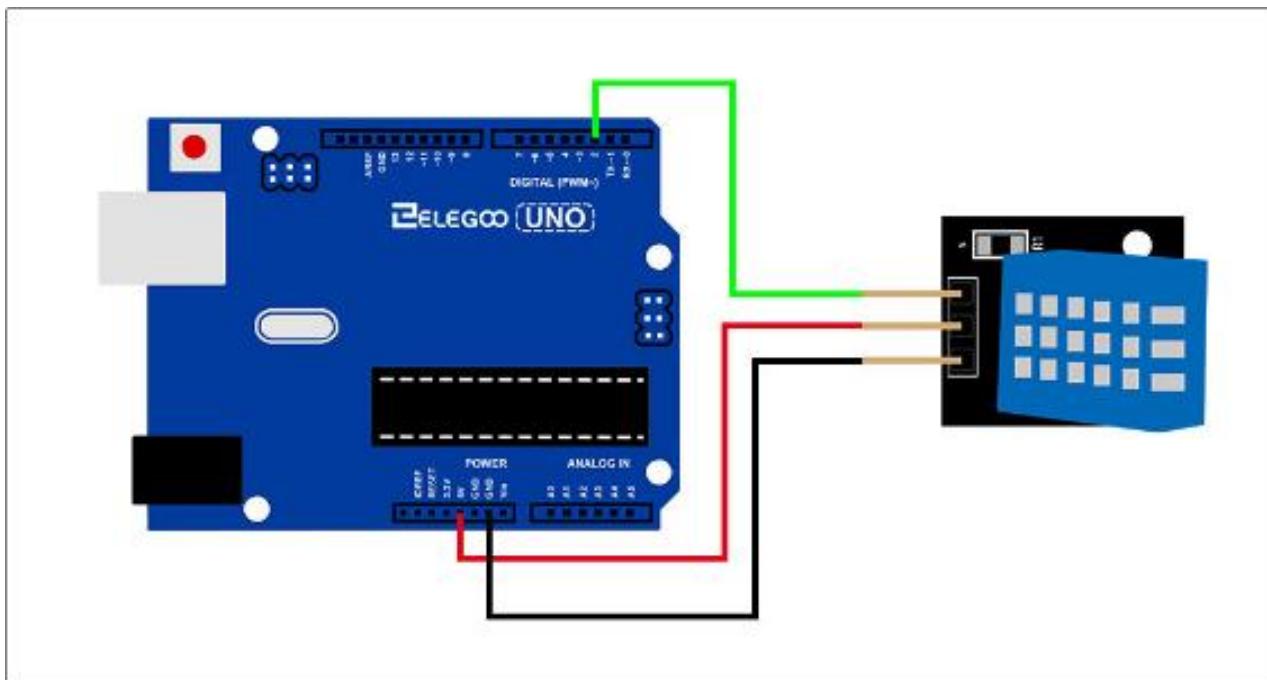


Connessione

Schematico



Schema di collegamento



Come potete vedere abbiamo solo bisogno di 3 connessioni al sensore, dal momento che uno del perno non viene utilizzato.

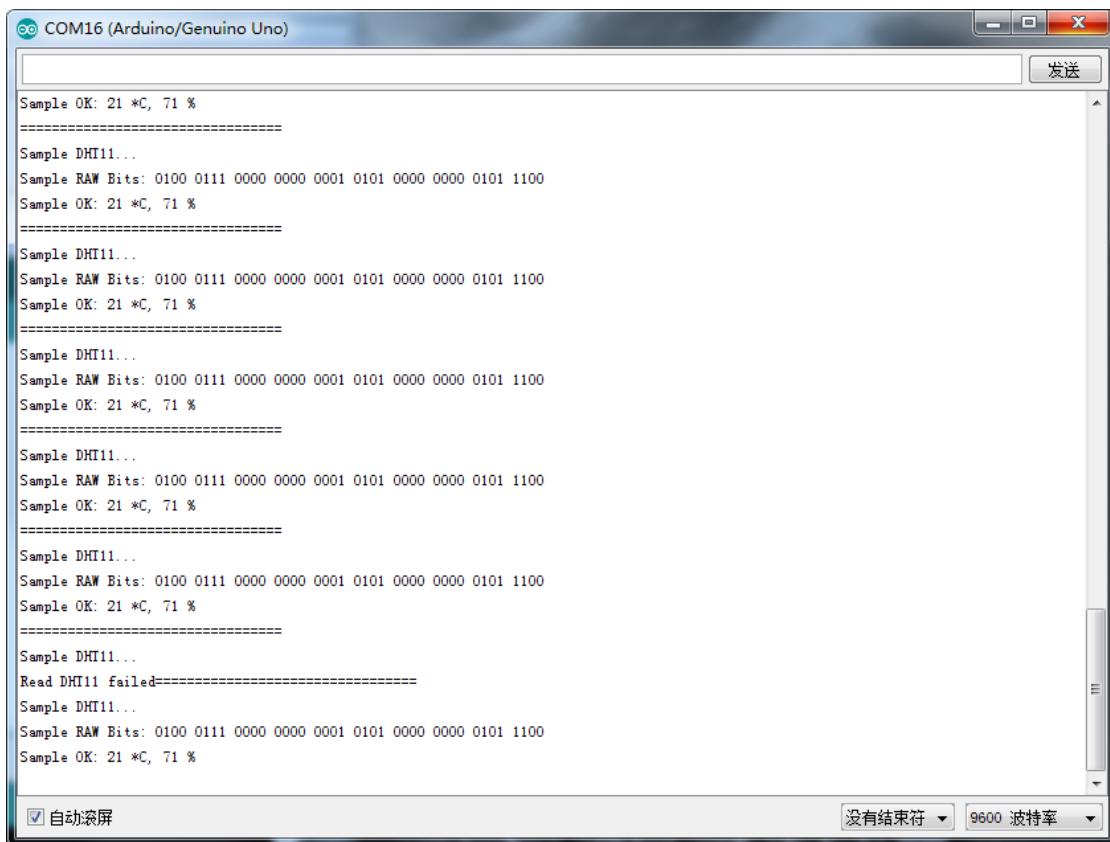
Il collegamento sono: tensione, a terra e di segnale che può essere collegato a qualsiasi analogico Pin sul nostro UNO.

Codice

Ora che abbiamo la configurazione fisica, tutti abbiamo bisogno ora è il codice.

Prima di poter eseguire questo, è necessario assicurarsi di avere installare il <simpleDHT> libreria. Oppure è necessario installare di nuovo. Se non si esegue questa operazione, il codice non funzionerà.

Carica il programma quindi aprire il monitor, possiamo vedere i dati, come di seguito: (Indica la temperatura dell'ambiente, possiamo vedere che è di 21 gradi)



```
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Read DHT11 failed=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
```

Lezione 20: Modulo di joystick analogico

Generali

joystick analogici sono un ottimo modo per aggiungere un certodi controllo nei vostri progetti.

In questa esercitazione impareremo come usare il modulo joystick analogico.

Componenti necessari

(1) Elegoo UNO R3

(1) Modulo di Joystick

(5) F-M fili

Introduzione di Componente

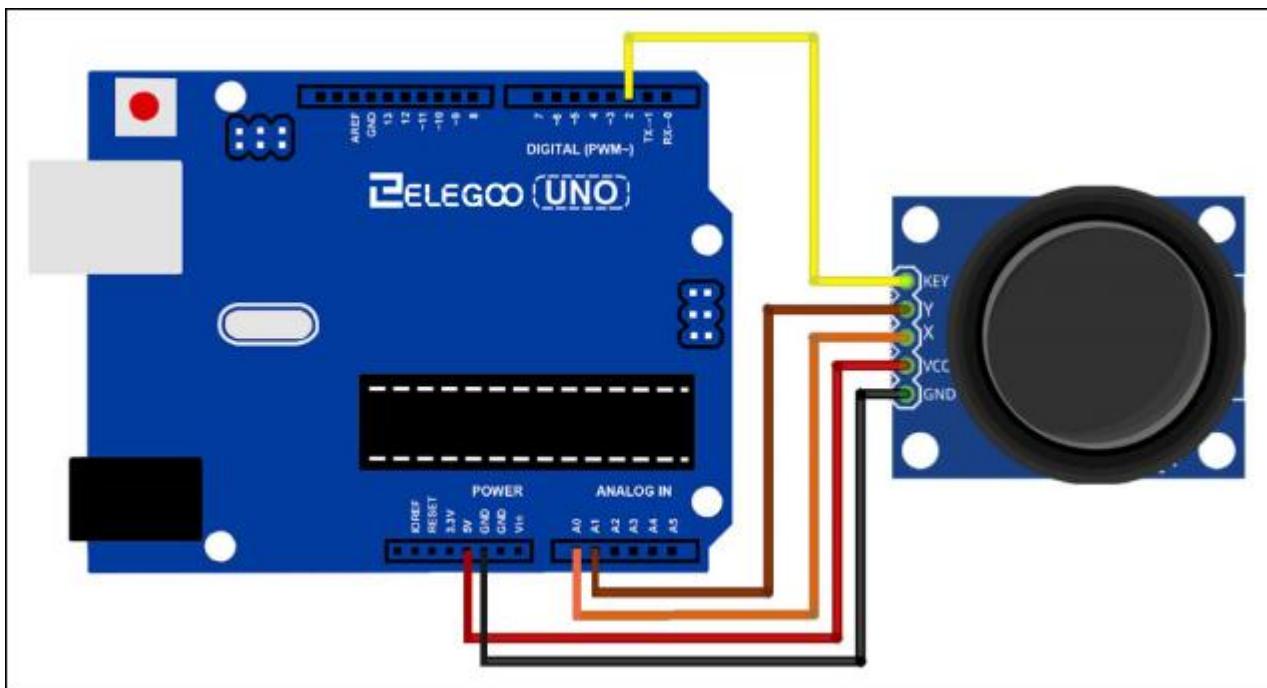
Joystick

Il modulo dispone 5 perni: Vcc, a terra, X, Y, chiave. Si noti che le etichette sul tuo potrebbero essere leggermente diversi, a seconda di dove hai preso il modulo da. La puntina è analogico e dovrebbe fornire letture più accurate di semplice 'direzionale' Joystick usare tattò alcune forme di pulsanti o interruttori meccanici. Inoltre, è possibile premere il joystick verso il basso (piuttosto duro sul mio) per attivare un 'premere per selezionare' pulsante.

Dobbiamo usare perni Arduino analogici per leggere i dati dal X / Y perni, ed un perno digitale per leggere il tasto. Il perno chiave è collegato a massa, quando viene premuto il joystick, e galleggia altrimenti. Per ottenere letture stabili dalla chiave / Scegli perno, ha bisogno di essere collegato a Vcc attraverso una resistenza di pull-up. Il costruito nel resistori sui perni digitali Arduino può essere usato. Per un tutorial su come attivare i resistori pull-up per i perni Arduino, configurati come ingressi

Connessione

Schema di collegamento



Abbiamo bisogno di 5 collegamenti con il joystick.

Il collegamento sono: Chiave, Y, X, di tensione e di terra.

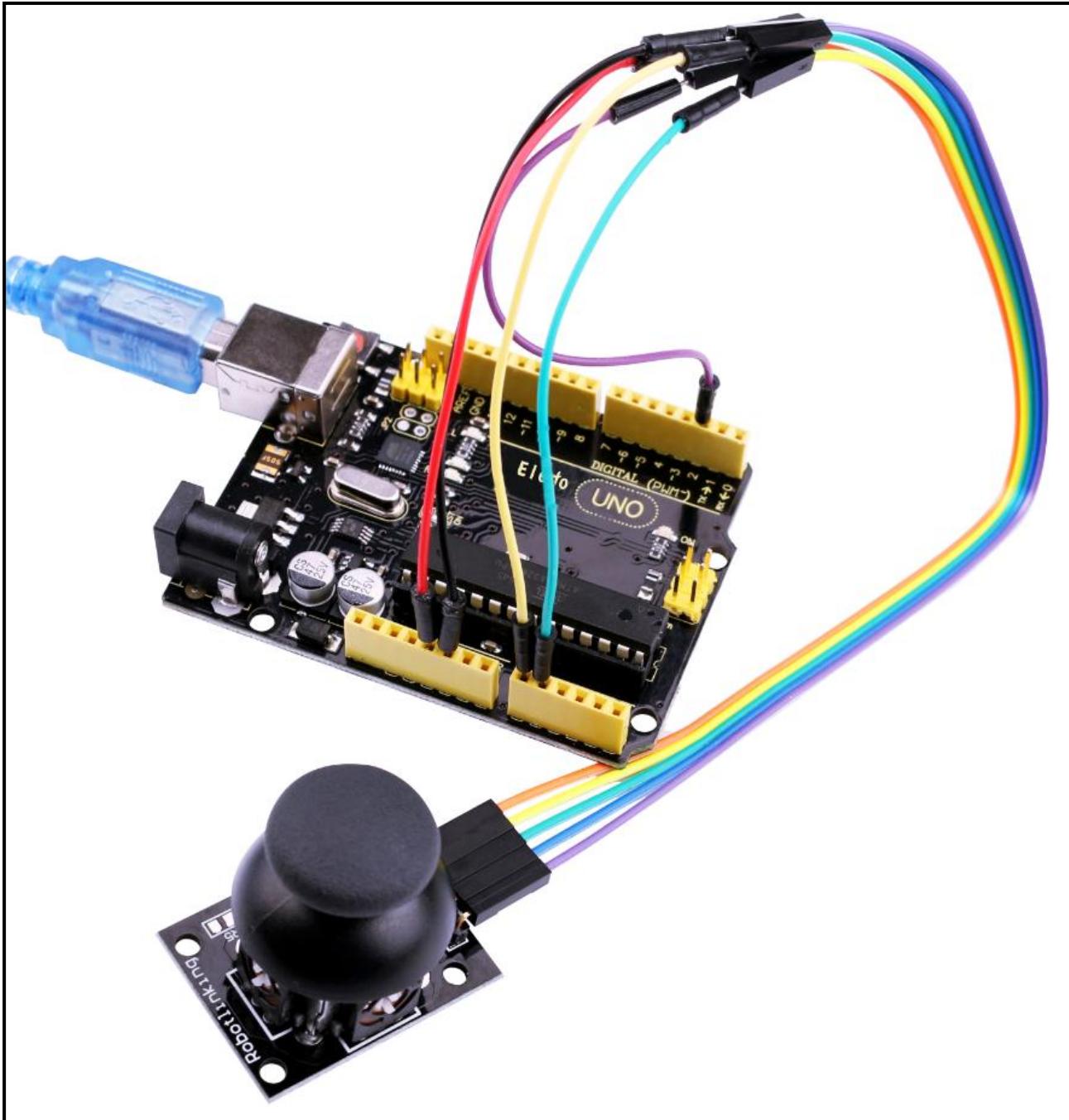
"Y e X" sono analogici e "Chiave" è digitale. Se non è necessario l'interruttore, è possibile utilizzare solo 4 perni.

Codice

Joystick analogici sono fondamentalmente Potenziometri così tornano valori analogici.

Quando il joystick è in posizione di riposo o medio, dovrebbe restituire un valore di circa 512.

L'intervallo di valori compresi tra 0 e 1024.



Lezione 21: Modulo di IR Ricevitore

Generali

Uso di un telecomando a infrarossi è un ottimo modo per avere il controllo wireless del vostro progetto.

telecomandi infrarossi sono semplici e facili da usare. In questo tutorial ci sarà il collegamento del ricevitore IR per l'ONU, e quindi utilizzare una libreria che è stata progettata per questo particolare sensore.

Nel nostro schizzo avremo tutti i codici IR esadecimali che sono disponibili su questo telecomando, ci sarà anche rilevare se il codice è stato riconosciuto e anche se stiamo tenendo premuto un tasto.

Componenti necessari

- (1) Elegoo UNO R3
- (1) IR Modulo Ricevitore
- (1) IR remote
- (3) F-M fili

Introduzione di Componente

IR RICEVITORE SENSOR:

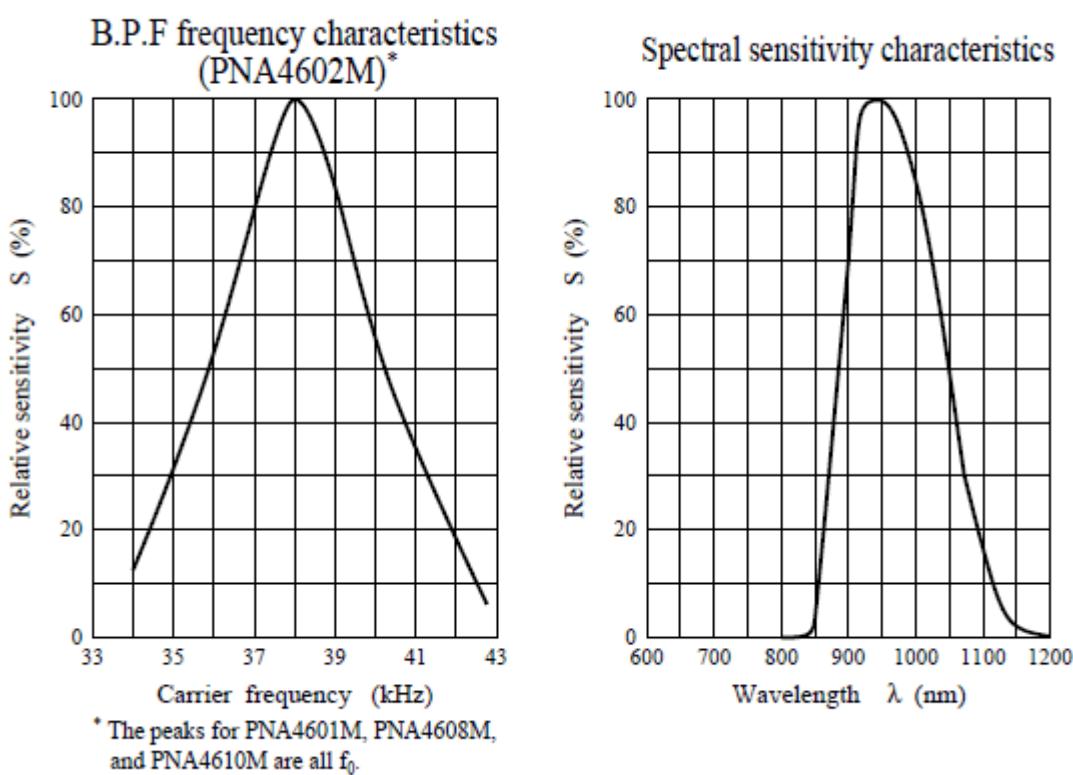
Rivelatori IR sono piccoli microchip con una fotocellula che sono sintonizzati per ascoltare la luce a infrarossi. Sono quasi sempre utilizzato per il rilevamento telecomando - ogni piatto e lettore DVD ha uno di questi nella parte anteriore per ascoltare il segnale IR dal clicker.

All'interno del telecomando è un LED IR di corrispondenza, che emette impulsi IR per dire al televisore per attivare, disattivare o cambiare canale. Luce a infrarossi non è visibile all'occhio umano, il che significa che ci vuole un po 'di lavoro per testare una configurazione.

Ci sono un paio di differenze tra queste e dicono un CDS fotocellule: rivelatori IR sono appositamente filtrati per la luce infrarossa, non sono bravi a rilevare la luce visibile. D'altra parte, fotocellule sono bravi a rilevare la luce visibile di colore giallo / verde, non è bravo a luce IR.

- rivelatori IR hanno un demodulatore interno che cerca IR modulato a 38 KHz. Proprio brillante non verrà rilevato un LED a infrarossi, deve essere PWM lampeggiare a 38KHz. Le fotocellule non hanno alcun tipo di demodulatore e possono rilevare qualsiasi frequenza (compreso DC) entro la velocità di risposta della fotocellula (che è di circa 1 KHz)
- rivelatori IR sono digitali fuori - o rilevano segnali 38KHz IR e l'uscita bassa (0 V) o che non rilevano alcuna e l'uscita alta (5V). Fotocellule comportano come resistori, la resistenza cambia a seconda della quantità di luce a cui sono esposti.

Che cosa potete Misurare

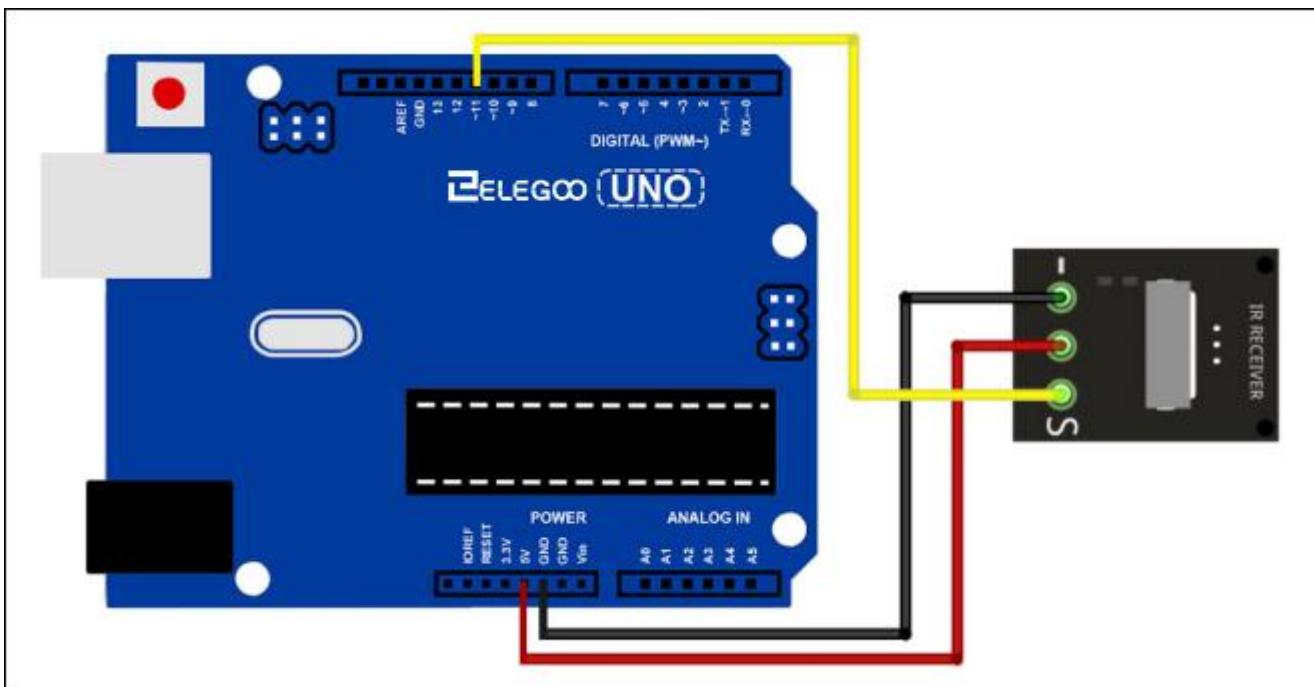


Mentre si può vedere da questi grafici Foglio dati, la rilevazione del picco di frequenza è a 38 KHz e il colore del LED di picco è 940 nm. È possibile utilizzare da circa 35 KHz a 41 KHz, ma la sensibilità sarà scendere in modo che non rileverà pure da lontano. Allo stesso modo, è possibile utilizzare 850 a 1100 LED nm ma non funziona così come 900 a 1000nm in modo da assicurarsi di ottenere i LED corrispondenti! Controllare la scheda per il vostro IR LED per verificare la lunghezza d'onda.

Cercare di ottenere un 940nm - ricordate che non è 940nm luce visibile (la sua Infra Rossa)!

Connessione

Schema di collegamento



Ci sono 3 connessioni al ricevitore IR.

Le connessioni sono: segnale, tensione e di terra.

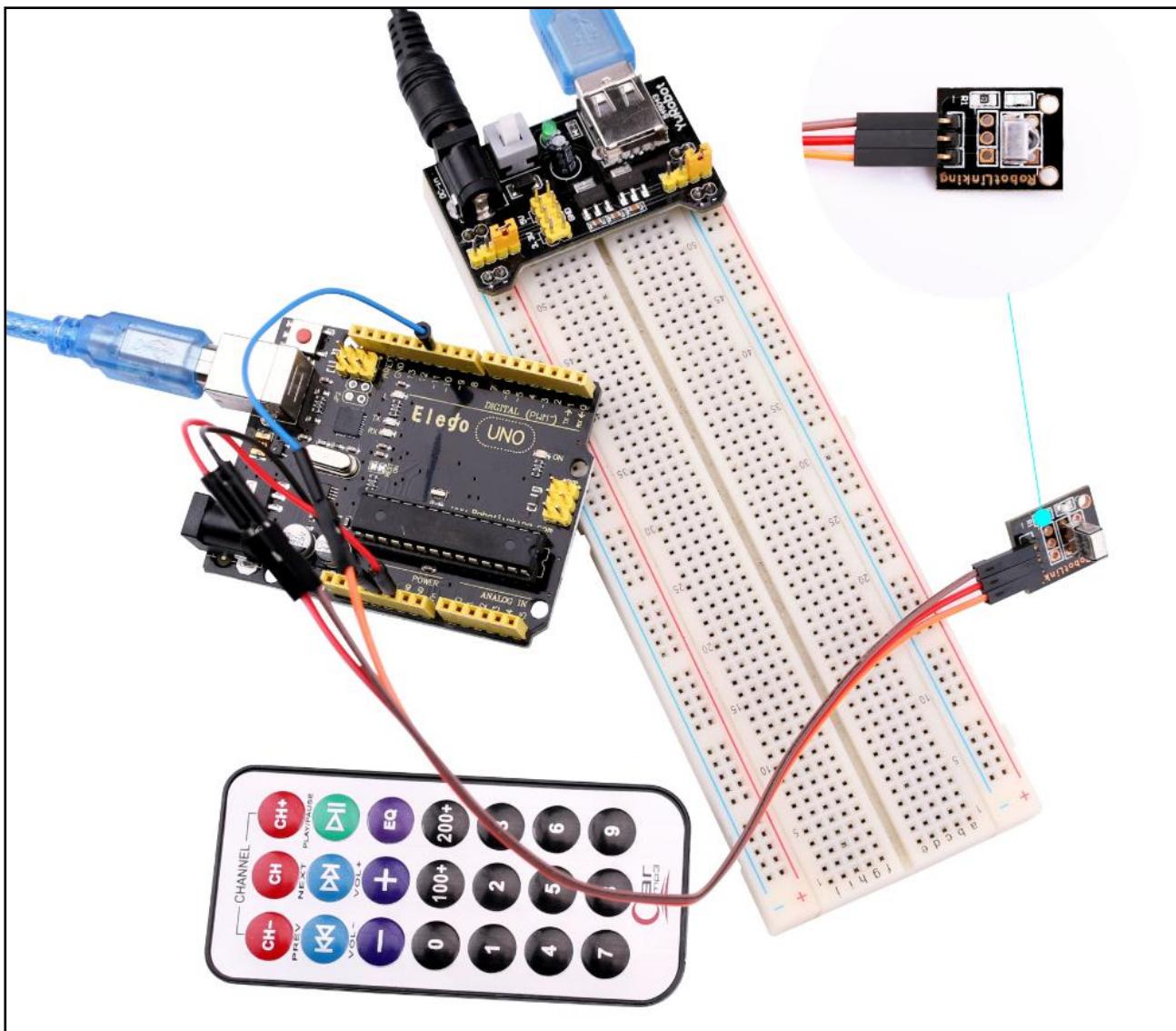
Il "-" è la terra, "S" è il segnale, e il perno centrale è tensione 5V.

Codice

Prima di poter eseguire questo, assicurarsi di aver installato il <IRremote> biblioteca o ri-installarlo, se necessario. In caso contrario, il codice non funzionerà.

Successivo ci si sposterà il <RobotIRremote> fuori dalla cartella Libreria, facciamo questo perché che i conflitti di libreria con quello che useremo. Si può semplicemente trascinare indietro all'interno della cartella della libreria, una volta che hai finito la programmazione del microcontrollore.

Dopo aver installato la libreria, basta andare avanti e riavviare il software IDE.



Lezione 22: DC Motori

Generali

In questa lezione, imparerete come controllare un piccolo Motore DC con un UNO R3 e un transistor.

Componenti Necessari

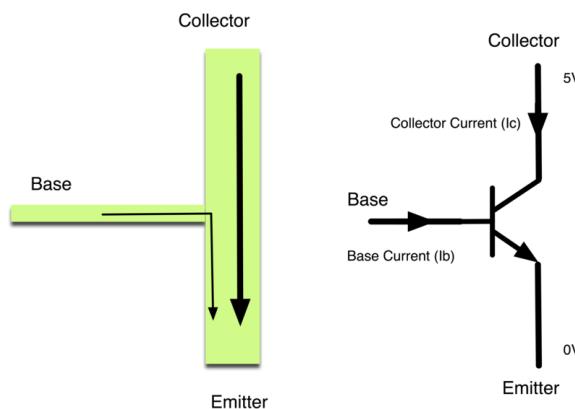
- (1) Elegoo UNO R3
- (1) Breadboard
- (1) 6v DC motore
- (1) PN2222
- (1) 1N4007
- (1) 220 ohm resistore
- (3) M-M fili

Introduzione di Componente

TRANSISTORI:

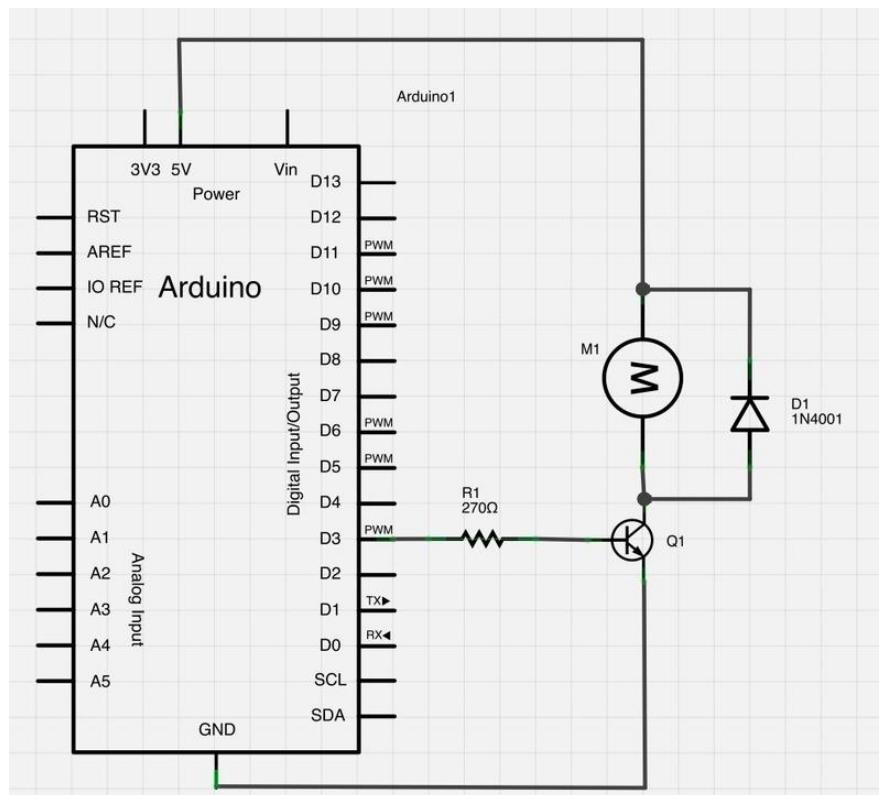
Il piccolo Motore DC è probabile che l'uso più potere di una scheda di UNO R3 uscita digitale in grado di gestire direttamente. Se abbiamo provato a collegare il motore direttamente a un perno bordo UNO R3, c'è una buona possibilità che potrebbe danneggiare la scheda di UNO R3.

Un piccolo transistor come il PN2222 può essere utilizzato come un interruttore che utilizza solo un po corrente dall'uscita digitale scheda UNO R3 per controllare la corrente molto più grande del motore.



Il transistor ha tre conduttori. La maggior parte dell'energia elettrica fluisce dal collettore al trasmettitore, ma questo avverrà solo se una piccola quantità scorre nella connessione di base. Questa piccola corrente è fornita dall'uscita digitale scheda UNO R3.

Qui di seguito viene chiamato un diagramma schematico del. Come un layout basetta, è un modo di mostrare come le parti di un progetto elettronico sono collegati insieme.



La D3 perno della scheda UNO R3 è collegato alla resistenza. Proprio come quando si utilizza un LED, questo limita la corrente che scorre nel transistore attraverso la base.

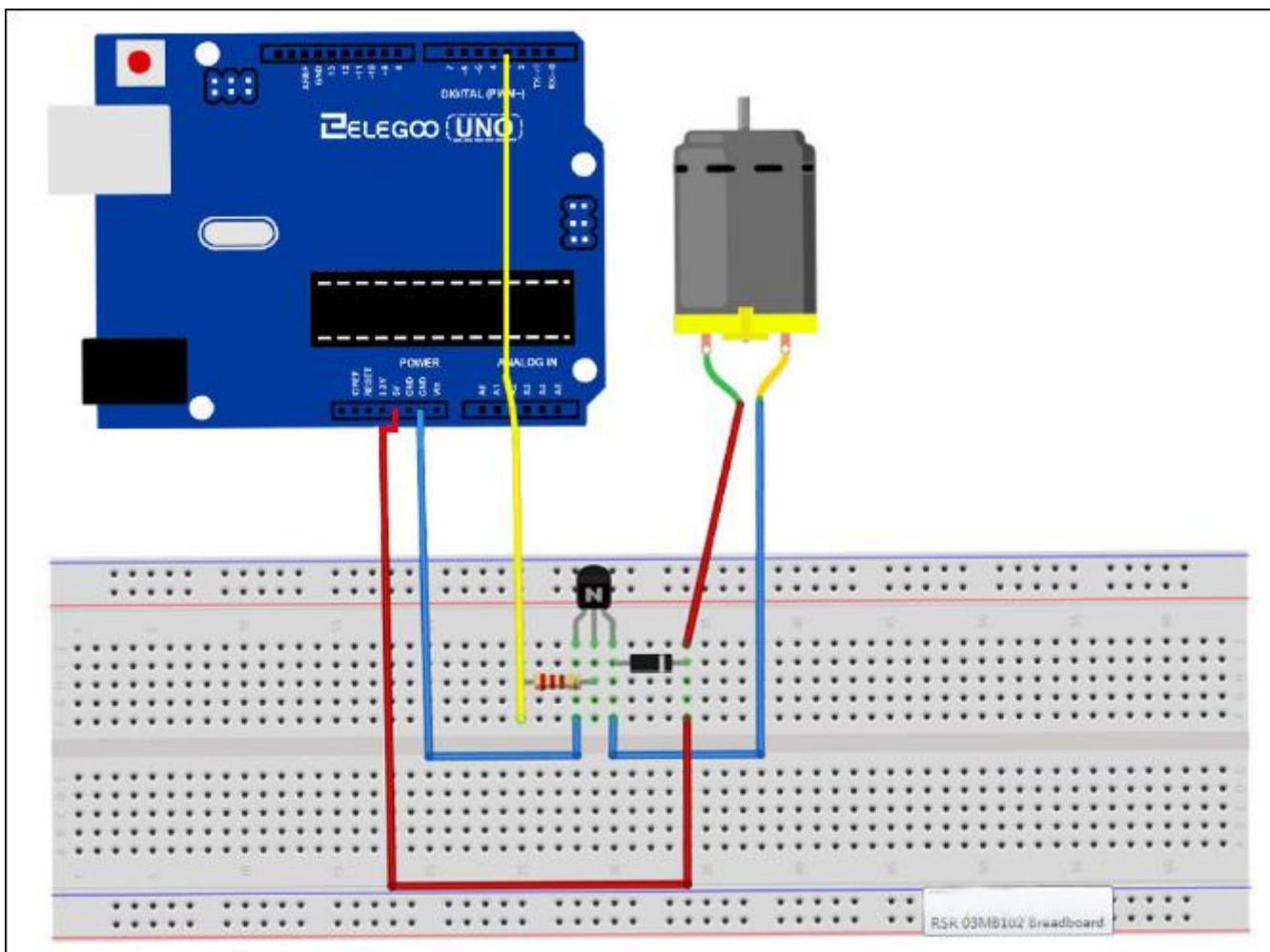
C'è un diodo collegato attraverso i collegamenti del motore. Diodi consentono solo energia elettrica di fluire in una direzione (direzione della loro freccia).

Quando si accende l'alimentazione ad un motore, si ottiene un picco negativo di tensione, che può danneggiare la scheda UNO R3 o il transistor. Il diodo protegge da questo, un cortocircuito dei ogni attuale tale inversione dal motore.

Attraverso questo esperimento, si è appreso come guidare LCD1602. Ora è possibile creare i propri messaggi da visualizzare! Si può anche provare lasciando i vostri LCD 1602 numeri di visualizzazione.

Connessione

Schema di collegamento



Si utilizzerà un'uscita scheda analogica UNO R3 (PWM) per controllare la velocità del motore inviando un numero compreso tra 0 e 255 dal Serial Monitor.

Quando si mettono insieme la basetta, ci sono due cose da guardare fuori per.

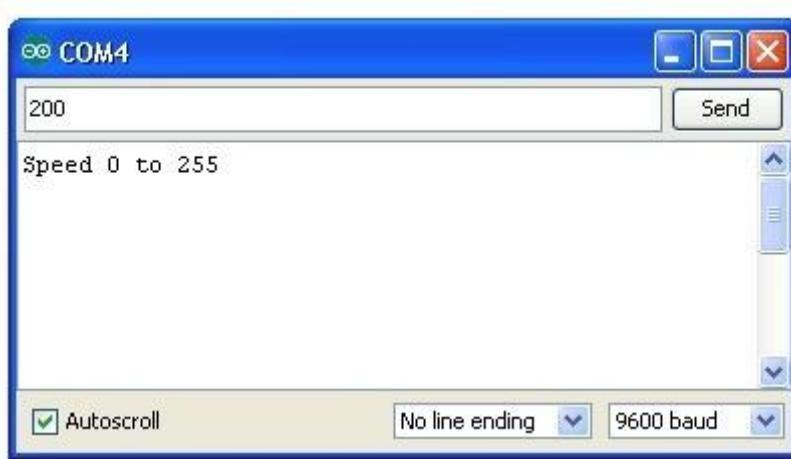
Innanzitutto, assicurarsi che il lato piatto del transistore si trova sul lato destro della basetta.

In secondo luogo, la fine strisce del diodo dovrebbe essere verso la linea + 5V - vedere l'immagine qui sotto!

Il motore che viene fornito con il kit non disegna più di 250mA, ma se si dispone di un motore diverso, si potrebbe facilmente trarre 1000mA, più di una porta USB in grado di gestire! Se non si è sicuri di assorbimento di corrente di un motore, alimentare la scheda UNO R3 da un adattatore a muro, non solo USB.

Il transistor si comporta come un interruttore, il controllo della potenza al motore, UNO R3 perno bordo 3 viene utilizzato per attivare il transistor e si spegne e viene dato il nome di 'motorPin' nel disegno.

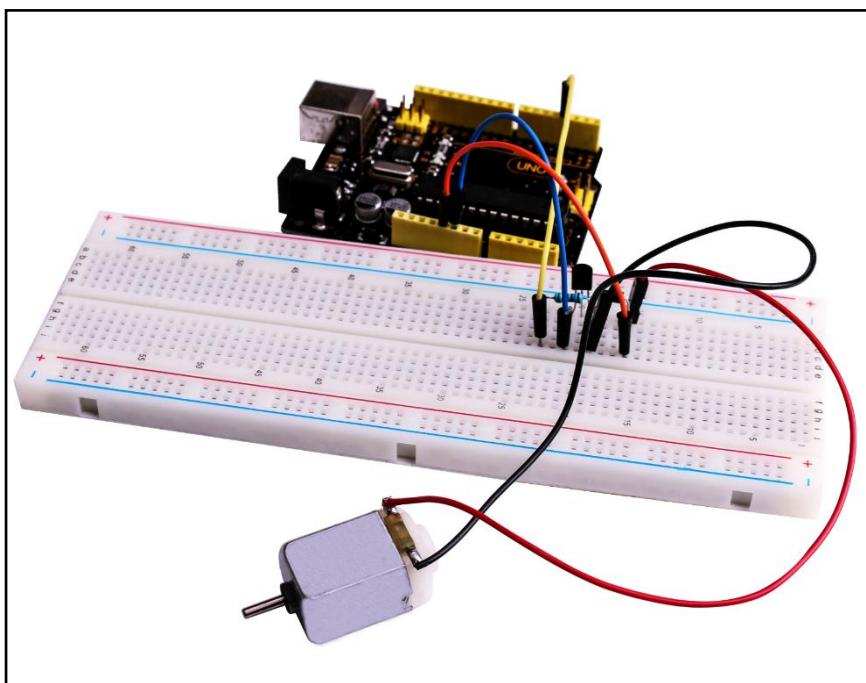
Quando il disegno viene avviato, induce a ricordare che per controllare la velocità del motore, è necessario immettere un valore compreso tra 0 e 255 nel monitor di serie.



Nella funzione 'loop', il comando 'Serial.parseInt' viene utilizzato per leggere il numero inserito come testo nel monitor seriale e convertirlo in un 'int'.

È possibile digitare qualsiasi numero qui, in modo che il 'se' dichiarazione sulla riga successiva fa solo una scrittura analogico con questo numero se il numero è compreso tra 0 e 255.

Codice



Lezione 23: Motore passo-passo

Generali

In questa lezione, si impara un modo divertente e facile da guidare un motore passo-passo.

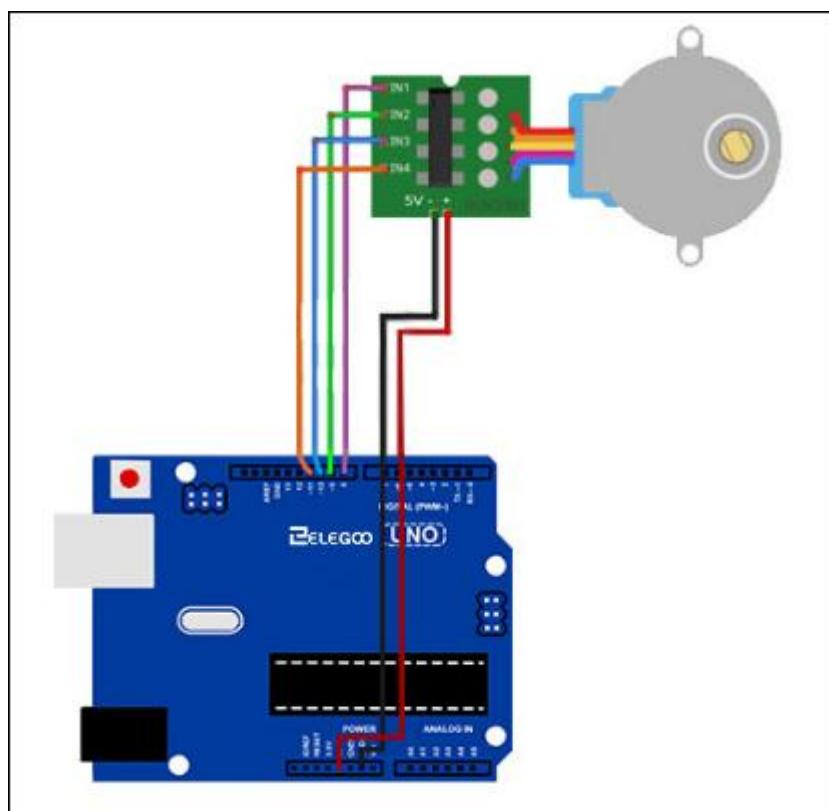
Il passo-passo che stiamo usando è dotato di un proprio bordo di driver che lo rende facile da collegare al nostro UNO.

Componenti necessari

- (1) Elegoo UNO R3
- (1) ULN2003 modulo del driver motore passo-passo
- (1) Motore passo-passo
- (9) F-M fili

Connessione

Schema di collegamento



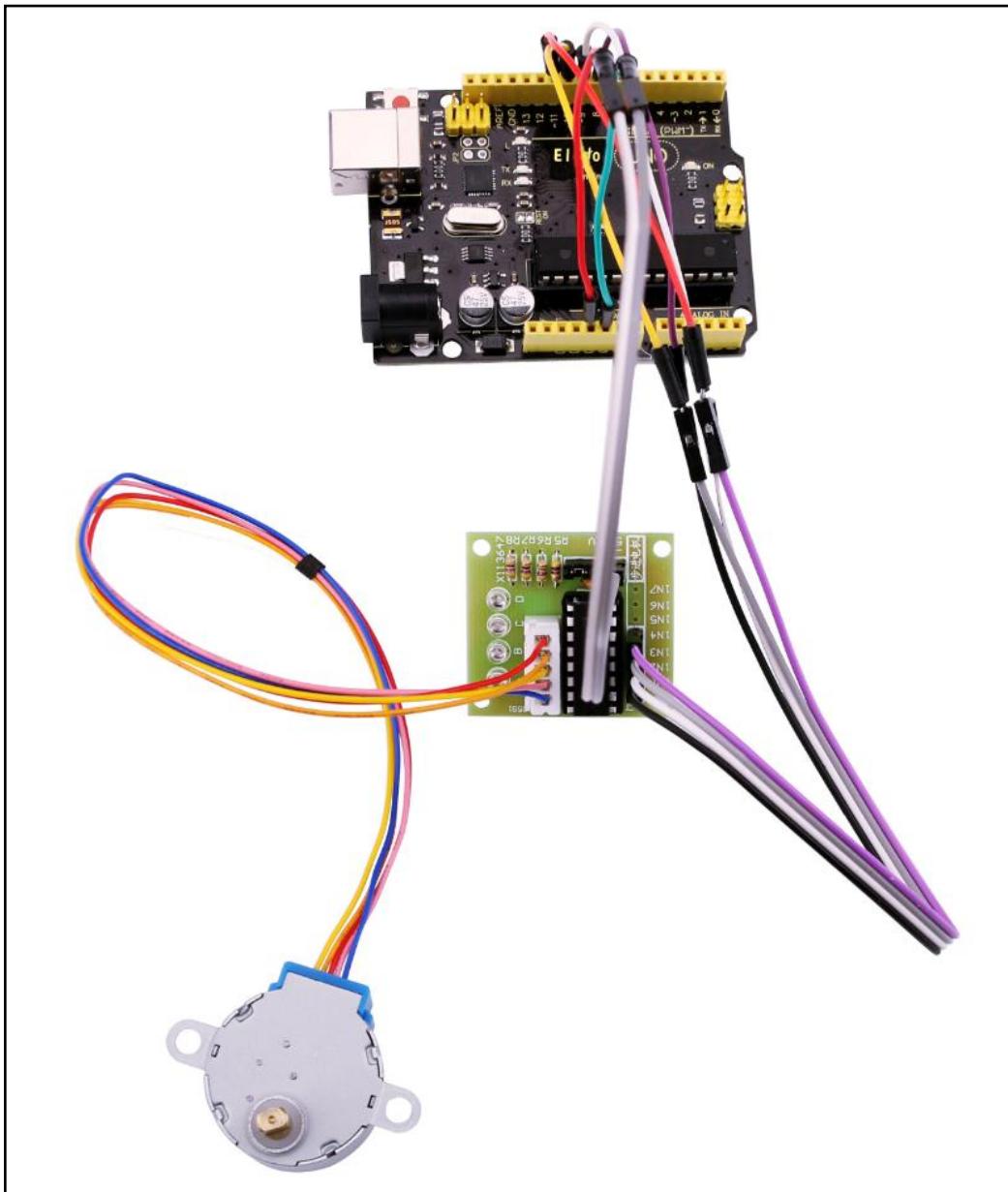
Stiamo usando 4 pin per controllare il passo-passo.

I perni 8-11 controllano il motore passo-passo.

Collegiamo il 5V e terra dal UNO al motore passo-passo.

Codice

Prima di poter eseguire questo, assicurarsi di aver installato il <Stepper_Motor_Control> biblioteca e reinstallarlo, se necessario. In caso contrario, il codice non funzionerà.



Lezione 24: Il controllo del motore passo-passo con telecomando

Generali

In questa lezione, si impara un modo divertente e facile da controllare un motore passo-passo a distanza con un telecomando a infrarossi.

Il passo-passo che stiamo usando è dotato di un proprio bordo di driver che lo rende facile da collegare al nostro UNO.

Dal momento che non vogliamo per azionare il motore direttamente dal UNO, useremo un alimentatore poco basetta economico che si inserisce a destra nella nostra basetta e il potere con un alimentatore da 9V 1Amp.

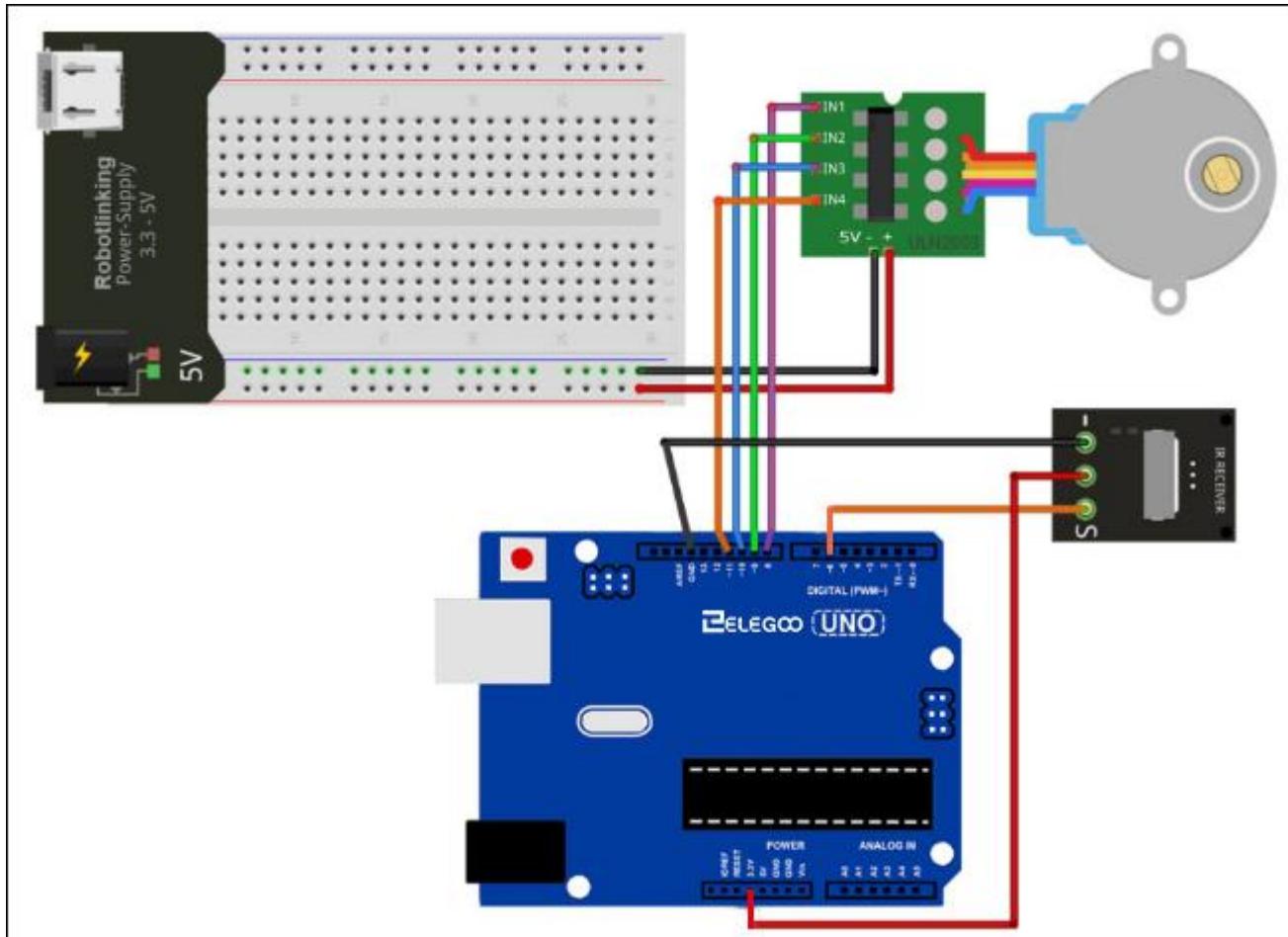
Il sensore IR è collegato al UNO direttamente poiché usa quasi nessun potere.

Componenti necessari

- (1) Elegoo UNO R3
- (1) Breadboard
- (1) Modulo ricevitore
- (1) IR remoto
- (1) modulo del driver motore passo-passo di ULN2003
- (1) Motore passo-passo
- (1) Modulo di alimentazione
- (1) Adattatore
- (9) F-M fili

Connessione

Schema di collegamento



Stiamo usando 4 perno per controllare il passo-passo e 1 perno per il sensore IR.

I perni 8-11 controllano il motore passo-passo e il perno 6 sta ricevendo le informazioni IR.

Colleghiamo il 5V e terra dal UNO al sensore. Come precauzione, usare un potere basetta di alimentazione per alimentare il motore passo-passo poiché può utilizzare più energia e non vogliamo danneggiare l'alimentazione del UNO.

Codice

Prima di poter eseguire questo, assicurarsi di aver installato il <IRremote>

<Stepper_Motor_Control> libreria e reinstallarlo, se necessario. In caso contrario, il codice non funzionerà.

