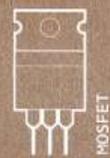


09



MOSFET



RESISTENZA DA 10 KILO OHM



DIODO 1N4007



MOTORE



PULSANTE



CONNETTORE PER BATTERIA



BATTERIA

INGREDIENTI

GIRANDOLA MOTORIZZATA

USA ARDUINO PER FAR RUOTARE UNA GIRANDOLA
COLORATA UTILIZZANDO UN MOTORE

Scopri: transistor, forti carichi di corrente e di tensione

Tempo: 45 MINUTI

Livello: ■■■■

Basato sui progetti: 1, 2, 3, 4

Per varie ragioni controllare motori con Arduino è molto più complicato che controllare i LED. Innanzitutto, i motori richiedono molta più corrente di quanta ne possano fornire i piedini d'uscita di Arduino e i motori possono generare corrente – attraverso un processo chiamato induzione – che può danneggiare il circuito. Comunque, i motori fanno muovere le cose, rendendo i progetti molto più interessanti. Ben vengano quindi le complicazioni!

Muovere le cose richiede molta energia, molta di più di quanto possa fornire Arduino. Alcuni motori hanno bisogno anche di una tensione più alta. All'inizio del movimento, e quando si ha un carico pesante attaccato, il motore assorbe quanta più energia possibile. Arduino può fornire solo 40 milliampere (mA) dai suoi piedini digitali, molto meno di quanto la maggior parte dei motori ha bisogno per lavorare.

I **transistor** sono componenti che permettono di controllare alti carichi di corrente e di tensione tramite la bassa corrente di uscita di un piedino di Arduino. Ce ne sono molti tipi diversi, ma tutti si basano sullo stesso principio. Puoi pensare a un transistor come a un interruttore digitale. Quando fornisci tensione a uno dei piedini del transistor, chiamato gate (porta), si chiude il circuito tra gli altri due piedini, chiamati source (sorgente) e drain (scarico). Così puoi accendere e spegnere un motore di elevata tensione o corrente con Arduino.

I **motori** sono dispositivi induttivi. L'induzione è un processo per cui una corrente elettrica variabile che scorre in un filo è in grado di generare un campo magnetico variabile. Quando si dà elettricità a un motore, una bobina di filo di rame avvolta strettamente all'interno del motore crea un campo magnetico. Questo campo fa girare l'asse, la parte che sporge dal motore.



È vero anche il contrario: un motore può generare elettricità quando l'albero motore viene fatto ruotare. Collega un LED ai due fili del motore, quindi fai girare l'albero con la mano. Se non succede niente, giralo dall'altra parte. Il LED si dovrebbe accendere. Hai appena realizzato un generatore con il tuo motore. Quando smetti di fornire energia elettrica al motore, questo continua a girare per inerzia. Girando, il motore genera una tensione nella direzione opposta alla corrente che hai fornito. Hai visto questo risultato quando il motore ha acceso il LED. Questa tensione inversa, a volte chiamata **controtensione**, può danneggiare il transistor. Per questa ragione, dovresti mettere un diodo in parallelo al motore, così la controtensione passa attraverso il diodo, che permette all'elettricità di scorrere in una direzione, proteggendo il resto del circuito.

CONSTRUISCI IL CIRCUITO

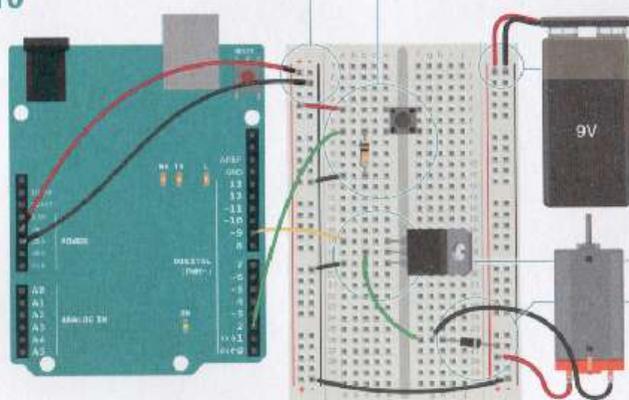


Fig. 1

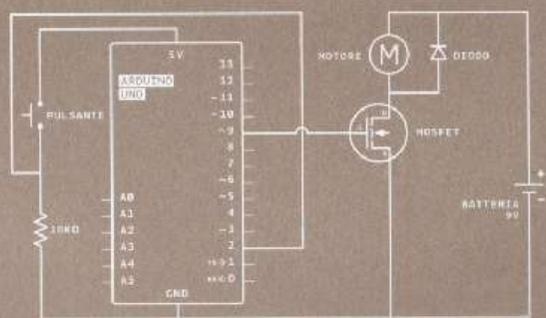


Fig. 2

- 1 Collega alimentazione e massa alla breadboard attraverso Arduino.
- 2 Aggiungi alla scheda un pulsante, connettendo un lato all'alimentazione e l'altro lato al piedino digitale 2 di Arduino. Aggiungi una resistenza pull-up da 10 kilo ohm a massa sul piedino di uscita dell'interruttore.
- 3 Quando usi circuiti con tensioni differenti, devi collegare tra loro le masse per creare una massa comune. Collega il connettore per batteria 9V alla breadboard. Collega la massa della batteria alla massa di Arduino sulla breadboard con un ponticello come mostrato in Fig. 1. Poi attacca l'estremità libera del motore all'alimentazione 9V.
- 4 Metti il transistor sulla breadboard. Verifica che la linguetta di metallo sia dalla parte opposta a te. Collega il piedino digitale 9 al piedino di sinistra del transistor. Questo piedino è chiamato **gate (porta)**. Una tensione sul gate crea un collegamento tra gli altri due piedini. Collega un'estremità del motore al piedino centrale del transistor. Questo piedino è chiamato **drain (scarico)**. Quando Arduino attiva il transistor, fornendo tensione al gate, questo piedino si collega al terzo piedino, chiamato **source (sorgente)**. Collega source a massa.
- 5 Poi, collega l'alimentatore del motore al motore e alla breadboard. L'ultimo componente da aggiungere è il diodo. Il diodo è un componente polarizzato: va inserito nel circuito solo in una specifica direzione. Nota che il diodo ha una striscia su una estremità: è il polo negativo, o catodo, del diodo. L'altro è il polo positivo o anodo. Collega l'anodo del diodo a massa del motore e il catodo del diodo all'alimentazione del motore. Guarda la Fig. 1. Sembra al rovescio, e infatti lo è. Il diodo ti aiuta a prevenire ogni controtensione generata dal motore che rientra nel circuito. Ricorda, la controtensione viene generata nella direzione opposta alla tensione che fornisci.



Anche i LED sono diodi, nel caso ti stupissi del fatto che anche le loro estremità si chiamano anodi e catodi. Ci sono molti tipi di diodi, ma hanno tutti in comune un elemento: permettono alla corrente di fluire dall'anodo al catodo, ma non il contrario.

IL CODICE

Dai un nome alle costanti e alle variabili

Il codice è molto simile a quello che hai usato la prima volta per accendere un LED. Prima di tutto, imposta le costanti per il pulsante e i piedini del motore e una variabile chiamata `switchState` per memorizzare il valore dello switch (pulsante).

Dichiara la direzione dei piedini

Nel `setup()`, dichiara il `pinMode()` dei piedini del motore (`OUTPUT`) e dell'interruttore (`INPUT`).

Leggi l'ingresso, attiva l'uscita se viene premuto

Il `loop()` è semplice. Controlla lo stato di `switchPin` con `digitalRead()`.

Se è premuto l'interruttore, imposta il motorPin a `HIGH`. Se non è premuto, imposta il piedino a `LOW`. Quando è `HIGH`, il transistor si attiva, completando il circuito. Quando è `LOW`, il motore è spento.

```
1 const int switchPin = 2;
2 const int motorPin = 9;
3 int switchState = 0;
```

```
4 void setup() {
5   pinMode(motorPin, OUTPUT);
6   pinMode(switchPin, INPUT);
7 }
```

```
8 void loop(){
9   switchState = digitalRead(switchPin);
10  if (switchState == HIGH) {
11    digitalWrite(motorPin, HIGH);
12  }
13  else {
14    digitalWrite(motorPin, LOW);
15  }
16 }
```



I motori hanno una tensione di esercizio ottimale. Lavorano anche con meno del 50% della tensione nominale e fino al 50% in più rispetto a quel valore. Se si varia la tensione, è possibile modificare la velocità di rotazione del motore. Non variarla troppo, però, o si brucia il motore.

I motori richiedono una particolare attenzione quando sono controllati da un microcontrollore; questo, infatti, non può fornire abbastanza corrente e/o tensione per alimentare un motore. Inoltre è bene utilizzare diodi per evitare di danneggiare il circuito.



I transistor sono dispositivi a stato solido, non hanno parti in movimento. Per questo motivo, è possibile accenderli e spegnerli molto rapidamente. Prova a collegare un potenziometro a un ingresso analogico e usalo per modulare l'ampiezza di impulso (PWM) del piedino che controlla il transistor. Cosa pensi succeda alla velocità del motore se si varia la tensione? Utilizzando vari motori a disco, si possono ottenere diversi effetti visivi?