

Levitazione magnetica

Obiettivo

Controllo tramite Arduino UNO della posizione di un magnete permanente attraverso il campo magnetico generato da una bobina percorsa da corrente.

Materiale occorrente

- Arduino UNO;
- MOSFET IRF520 (canale N) o equivalente;
- Filo di rame smaltato di diametro 0.7 mm;
- Rocchetto per bobina;
- Magneti permanenti (NdFeB di massa 2.4 g);
- Sensore di effetto Hall lineare SS49E.
- Resistenza da 220k Ω ;
- Diodo 1N4004 ;
- Saldatore e lega per saldature;
- Cavi di collegamento.

Descrizione

In base al segnale di un sensore di Hall posto sotto alla bobina, un controllo PID consente di modulare un segnale PWM di output adeguato per pilotare un MOSFET con funzione di interruttore e quindi variare la corrente media circolante nella bobina al fine di regolare il campo magnetico generato per poter mantenere il magnete permanente sospeso ad una distanza costante al di sotto di questa.

Il sensore ad effetto Hall, alimentato a 5 V da Arduino, permette di misurare intensità e segno del campo magnetico in corrispondenza del sensore. L'offset di tensione è pari alla metà della tensione con cui viene alimentato: ad un campo magnetico nullo corrisponde un output di 2.5 V. L'output analogico della risposta è proporzionale all'intensità del campo e aumenterà o diminuirà in base al segno della forza magnetica rilevata, partendo appunto dall'offset del segnale.

Il PID (Proporzionale-Integrativo-Derivativo) è un sistema di controllo che funziona in retroazione negativa. Confrontando un valore acquisito con un valore di riferimento, è in grado di reagire a un eventuale errore facendolo tendere verso il valore zero. Il sistema serve quindi per mantenere il magnete in una posizione di equilibrio misurandone il campo magnetico prodotto in corrispondenza del sensore, posto sotto alla bobina, che è correlato alla distanza tra magnete e sensore. In questo caso il controllo PID è eseguito sul valore della posizione in altezza del magnete permanente, legata alla lettura del sensore ad effetto Hall idealmente secondo la formula del campo magnetico generato da una bobina percorsa da corrente, quindi inversamente proporzionale alla radice cubica del campo.

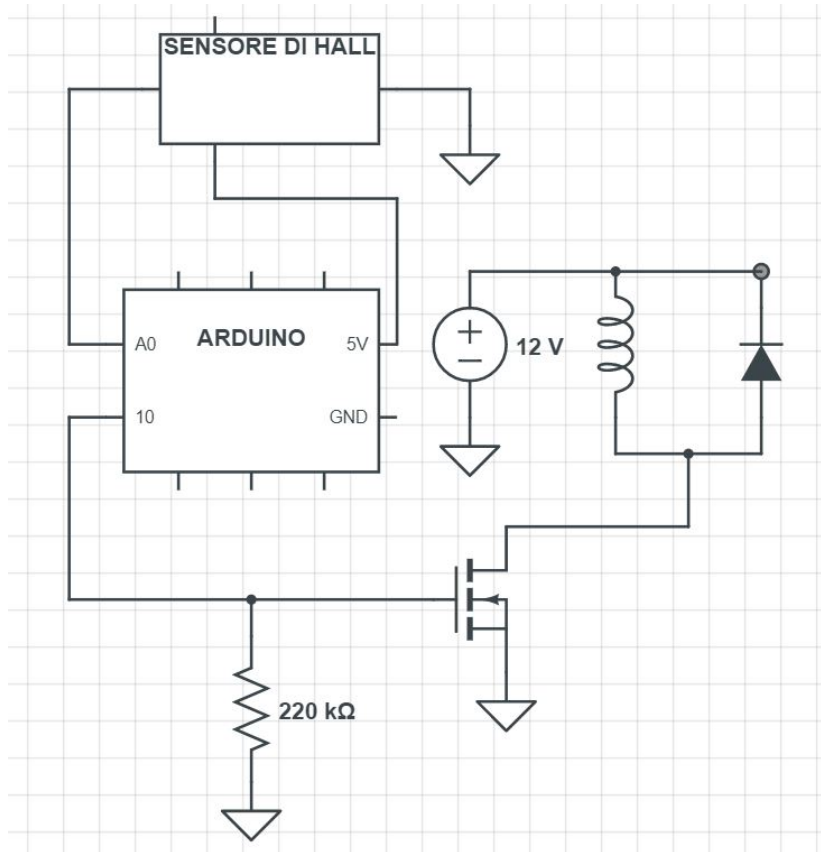


Figura 1: Circuito utilizzato per il progetto.

Il MOSFET (Metal Oxide Semiconductor FET) è un dispositivo controllato dalla tensione applicata al gate. È in funzione di questa che il circuito tra drain e source si comporta come una resistenza variabile, il cui valore dipende dalla tensione tra gate e source V_{GS} . Per il MOSFET utilizzato V_{GS} può variare fino a 20 V; se la tensione applicata è 0 V, la resistenza sarà molto alta e il MOSFET si comporterà come un interruttore aperto, mentre se la tensione applicata è superiore a un certo livello la resistenza sarà bassa (per il MOSFET a disposizione fino a 0.27Ω) e si comporterà come interruttore chiuso. Per controllare la corrente che scorre nell'avvolgimento, si è interessati al funzionamento come interruttore aperto/chiuso. Dal punto di vista del microcontroller o della logica che comanda il gate del MOSFET, esso si presenta come un condensatore; l'unica corrente richiesta è quella necessaria per caricare o scaricare questa capacità. Il MOSFET dissipa molta potenza quando passa da on a off e viceversa, quindi per avere alta efficienza è necessario che il MOSFET passi da interruttore on/off in poco tempo, quindi che il caricamento della capacità di gate sia veloce rispetto alla durata del periodo del PWM. Questa condizione è stata verificata con l'utilizzo di un oscilloscopio. Per la configurazione del circuito, il gate è alimentato da Arduino a 5 V, con il source collegato a ground; il drain è invece diretto alla bobina.

Quando il MOSFET cambia stato logico, nella bobina si genera una f.e.m. autoindotta opposta alla tensione di alimentazione che può essere alta. Per questo motivo un diodo, polarizzato direttamente rispetto alla f.e.m., è posto in parallelo alla bobina, al fine di limitare le sovratensioni che possono danneggiare la componentistica del circuito.

Progettazione

Inizialmente si è stimato il numero di avvolgimenti della bobina necessari per produrre un campo magnetico in grado di sollevare il magnete permanente. Posizionando il magnete su una bilancia attraverso un piedistallo, una spira di filo di collegamento percorsa da 1 A è stata avvicinata a distanza di qualche centimetro. La spira è in grado di sollevare 0.03 g/A circa, quindi per i magneti a disposizione, di 2.4 grammi, occorrono almeno 80 spire/A. Avendo effettuato la stima con un filo di diametro minore rispetto a quello a disposizione per l'avvolgimento, si è quindi deciso di costruire una bobina con un filo di rame smaltato di 0.7 mm di diametro con circa 900 spire. Si è avvolta la bobina su un rocchetto di diametro e altezza pari a 3 cm, stampato appositamente con una stampante 3D di materiale PLA e disegnato con software *OpenScad*. Dalle dimensioni del rocchetto si stimano quindi circa 60 metri di filo occorrente. La resistenza misurata dell'avvolgimento è di 6.6 Ω .

Per quanto riguarda il codice implementato su Arduino, in una fase preliminare si opera una taratura del sistema misurando con il sensore di Hall il campo magnetico generato dalla sola bobina percorsa da corrente per ogni valore del duty cycle relativo al PWM. In seguito tali valori vengono sottratti alla misura rilevata dal sensore di Hall quando è presente anche il magnete permanente, in modo da avere una valutazione della distanza del magnete. Nella stesura del codice si è posta attenzione nel sincronizzare l'acquisizione dal sensore di Hall con il periodo del PWM. È stato poi inserito l'algoritmo per il controllo PID, comprendente un termine proporzionale, il quale si riferisce alla differenza tra il valore ottenuto per la posizione in altezza del magnete permanente e quello desiderato, uno integrativo, che si occupa di integrare nel tempo tramite metodo dei trapezi il termine precedente, ed uno derivativo, che invece ne fa la derivata vista come rapporto incrementale. Essendo l'input da stabilizzare una posizione, l'output del PID è una forza, che controbilancia quella generata dal dipolo del magnete permanente (proporzionale al gradiente del campo magnetico prodotto dalla bobina); per fare in modo che la forza sia indipendente dalla posizione, il valore di duty cycle per il PWM sarà quindi l'output del PID diviso per un fattore geometrico dipendente dalla posizione, in prima approssimazione proporzionale al campo magnetico.

L'aggiunta del collegamento all'oscilloscopio di ulteriori due pin di Arduino ha consentito una verifica del tempo di processo di acquisizione del segnale di campo magnetico (convertito poi in termini di distanza dalla bobina) e del calcolo dei valori del controllo PID. Il segnale visualizzato, che è un'onda quadra in modalità on quando i processi sono in corso e off al termine, è stata confrontata con quello relativo al pin collegato alla bobina, a cui arriva un segnale PWM di Arduino. I tempi dei processi di acquisizione e calcolo sono risultati inferiori alla durata del periodo del PWM e quindi adeguati per una corretta valutazione delle misure.

Infine per la ricerca delle costanti che fanno da peso ai tre termini del PID si è operato come di seguito:

- Per la parte proporzionale è stato aumentato gradualmente il valore della costante k_p fino a che il magnete fosse soggetto a oscillazioni con periodo più lungo possibile attorno ad un punto di equilibrio stabilito dal setpoint;
- La parte derivativa è necessaria per smorzare queste oscillazioni, quindi anch'essa è stata gradualmente aumentata in modo da ottenere oscillazioni rapidamente smorzate;
- La parte integrativa, utile ad eliminare ulteriori errori di posizionamento e stabilizzare ancora meglio il sistema, è stata impostata ad un valore basso rispetto alle costanti precedenti.

Osservazioni

Linearizzare l'output del controllo PID, effettuato sulla posizione, ha permesso di controllare bene il magnete permanente, in modo tale che anche per distanze piccole dalla bobina questo non venisse attratto con troppa intensità ma con la stessa forza. Anche le costanti del PID sono state trovate agevolmente e rendono perfettamente immobile il magnete per qualche minuto, dopo di che iniziano a presentarsi alcune oscillazioni; per un'ulteriore ottimizzazione di queste potrebbe essere utile modellizzare il sistema fisico, ad esempio con simulazioni in ambiente Matlab.