



**UNIVERSITÀ DEGLI STUDI DI ROMA
TOR VERGATA**

FACOLTÀ DI INGEGNERIA

CORSO DI LAUREA IN INGEGNERIA DELL'AUTOMAZIONE

A.A. 2020/2021

Tesi di Laurea

Sviluppo di algoritmi di controllo delle correnti nelle bobine poloidali di macchine per la fusione Tokamak, con riguardo al design sistemico per la cooperazione tra sistemi embedded per l'attuazione, misurazione e centrali di controllo.

RELATORE

Daniele Carnevale

CANDIDATO

Emanuele Alfano

CORRELATORI

Marco Passeri

Dedico questa tesi ai miei cari nonni.

Grazie per aver sempre creduto in me.

Indice

Ringraziamenti	1
Introduzione	3
1 Elementi Costitutivi	4
1.1 Trasformatore	4
1.2 Sensore di Corrente	5
1.2.1 Metodo di acquisizione	5
1.3 Driver di Corrente - IBT-2	6
1.3.1 Connessione di Controllo	6
1.3.2 Benchmark Driver	8
2 Catena di Acquisizione	10
2.1 Schema di acquisizione	10
2.1.1 Sample and Hold alla frequenza di campionamento	10
2.1.2 Storage su file delle informazioni	10
2.2 Embedded Message Pack (EMP)	10
2.2.1 Metodo di codifica	10
2.2.2 Struttura del codice	10
2.2.3 Benchmark	10
2.3 Post Elaborazione con Matlab	10

3	Modello teorico	11
3.1	Modellazione Fisica	11
3.2	Funzione di trasferimento	11
4	Controllo in Retroazione dall'uscita	12
4.1	Modello di controllore	12
4.2	Esperimenti	12
5	Conclusioni e sviluppi futuri	13
	Appendice A - Codice Arduino	14
5.1	Setup Registri	14
5.2	Generatore di Segnale	14
5.2.1	Segnali Base	14
5.2.2	Segnali Composti	15
	Appendice B - Codice EMP	16
	Appendice C - Matlab Post Elaboration	17
	Elenco delle figure	18

Ringraziamenti

Questa tesi è stata resa possibile dal contributo nella mia vita di tante persone, che giorno per giorno mi hanno sempre dato il loro sostegno, a voi dedico questa mia Tesi.

Un ringraziamento speciale alla mia famiglia, in particolare a mia **Madre** e mio **Padre**: è grazie al vostro sostegno e incoraggiamento se oggi sono riuscito a raggiungere questo traguardo.

La forza di arrivare qui, oggi, però non è dovuta solo a loro, devo per forza ringraziare dell'affetto e il sostegno speciale da parte dei miei cari amici, che ogni giorno hanno condiviso con me gioie, sacrifici e successi, senza voltarmi mai le spalle, mi hanno dato la forza di arrivare a questo prezioso traguardo. **Filippo, Gabriele, Marta**, grazie di TUTTO.

Un pensiero in particolare vola verso la mia dolce **Nicoleta**, è sicuramente grazie all'affetto e le attenzioni che mi hai donato che sono riuscito a tenere dritto il timone ed arrivare qui oggi. Per terminare voglio ringraziare tutti i professori che negli ultimi 18 anni hanno guidato il mio cammino, loro che hanno sempre creduto in me e nelle mie capacità. Un ringraziamento più speciale va però alla mia professoressa e mentore **Beniamina Rauch** che fu la prima a vedere il mio potenziale e coltivarlo.

Oltre a lei ringrazio il mio relatore **Daniele Carnevale** che in questi anni universitari, da quando mi ha conosciuto, ha sempre creduto in me e mi ha permesso di fare esperienze che mai avevo immaginato.

Un sentito grazie a tutti voi.

Introduzione

Il capitolo introduttivo è generalmente lungo tre pagine (almeno due). Una buona introduzione può essere preparata secondo il seguente schema caratterizzante tre blocchi consecutivi:

1. *Introduzione generale all'ambito in cui si colloca la tesi* (più o meno partendo da "caro amico").

Ad esempio: "La robotica nasce dall'esigenza di sostituire l'uomo in quei lavori che... " eccetera.

2. *Collocazione della tesi nell'ambito generale sopra descritto.* Ad esempio: "Questo lavoro di tesi si colloca nel contesto dell'automazione domestica. In particolare, con riferimento a quanto sopra accennato, l'esigenza di".

3. *Descrizione schematica della struttura della relazione* (un paragrafo o poco più). Ad esempio: "La tesi è strutturata come segue: nel Capitolo ?? viene discussa una ...,

Capitolo 1

Elementi Costitutivi

Se lo si desidera, utilizzare questo spazio per inserire un breve riassunto di ciò che verrà detto in questo capitolo. Inserire solo i punti salienti.

1.1 Trasformatore

La tesi va scritta usando la terza persona, per quanto possibile, tranne casi veramente eccezionali.

In inglese è piuttosto standard usare la prima persona (plurale) in testi tecnici. In italiano no.

1.2 Sensore di Corrente

In questa sezione vengono commentate alcune questioni estetiche e di forma legate alla tesi.

1.2.1 Metodo di acquisizione

asdasd

1.3 Driver di Corrente - IBT-2

Per l'attuazione del controllo di corrente nella bobina primaria del trasformatore, è stato usato il driver di corrente Handsontec, *BTS7960 High Current 43A H-Bridge Motor Driver* .

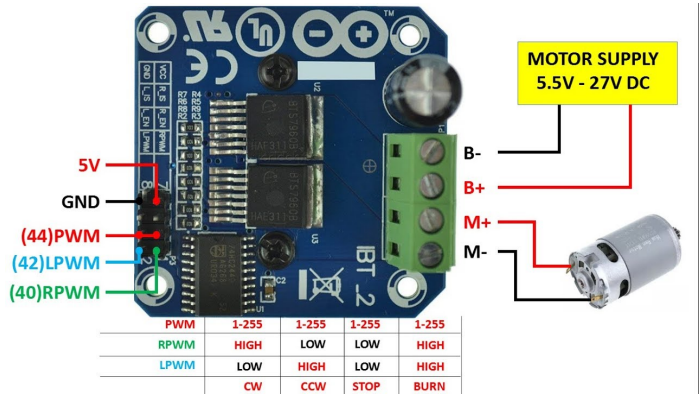


Figura 1.1: IBT-2 TopView.

Esso non è un comune Ponte-H integrato, per poter gestire potenze superiori è stato costruito usando 2 Half-Bridge collegati insieme mediante una opportuna logica per ricreare un normale Ponte-H. Questa scheda in particolare ha prestazioni interessanti per gli scopi di questa tesi, i principali sono elencati di seguito:

Power Input Voltage:	6 – 27 V
Peak current:	43 A
Massima Frequenza di PWM:	25 kHz
Protezione Sovra Tensioni	
Disaccoppiamento Ingresso di Potenza/Logica di controllo	

Di particolare interesse per l'esperimento è proprio la corrente di picco gestibile: avendo le dinamiche del sistema tempi inferiori ai 5 secondi, poter reggere correnti di picco così elevate rende la scheda perfetta per i nostri scopi.

1.3.1 Connessione di Controllo

Il driver permette 2 modalità di funzionamento:

Doppio PWM Modalità operativa che richiede l'uso di 2 PWM

Ciascun PWM controlla uno dei 2 Half-Bridge, e per evitare di bruciare i driver devono essere controllati singolarmente, il vantaggio di questa configurazione è la possibilità di usare 2 frequenze di controllo diverse.

Singolo PWM Modalità operativa classica di un normale Ponte-H

In questa modalità, la porta nand presente sulla scheda attua la logica di controllo opportuna per governare i 2 Half-Brige come fossero un normale Ponte-H.

Per il nostro esperimento si è scelto di usare il collegamento Singolo PWM così da evitare spiacevoli sorprese e avere il PWM di controllo sempre sincronizzato.

1.3.2 Benchmark Driver

Il driver sulla carta à buone prestazioni, ma non sono descritte le sue Non-Linearità, per farle risaltare si sono effettuati 2 esperimenti usando differenti input di controllo:

1. Onda Triangolare Periodica

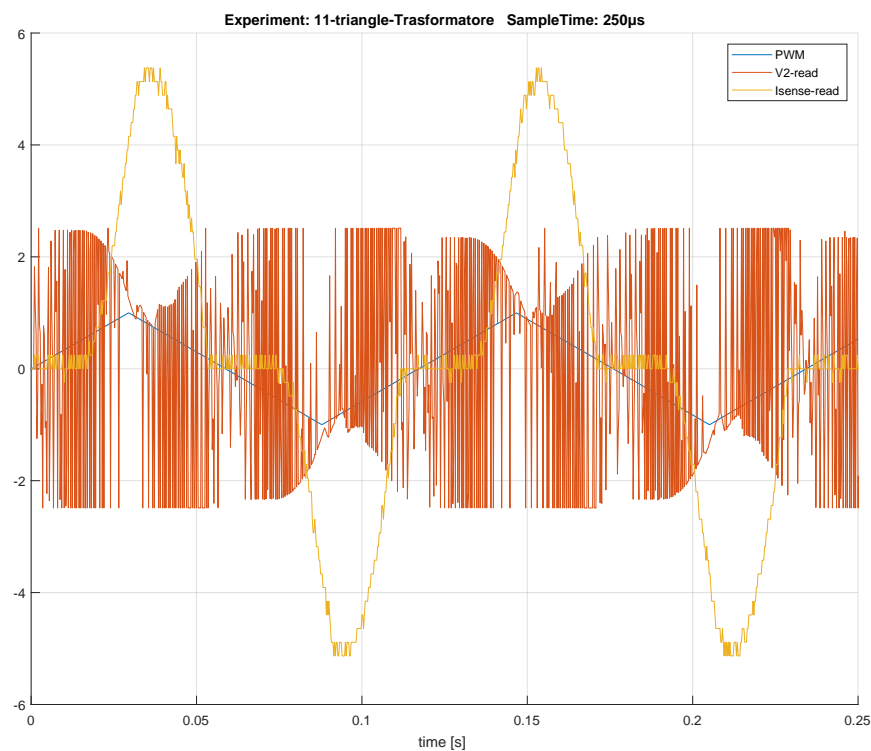


Figura 1.2: Onda Triangolare

Molto utile per far risaltare la Dead-Zone inferiore del driver, dovuta ai tempi di ritardo introdotti dai 2 Half-Bridge per evitare di cortocircuitare le alimentazioni.

2. Onda Trapezoidale Periodica

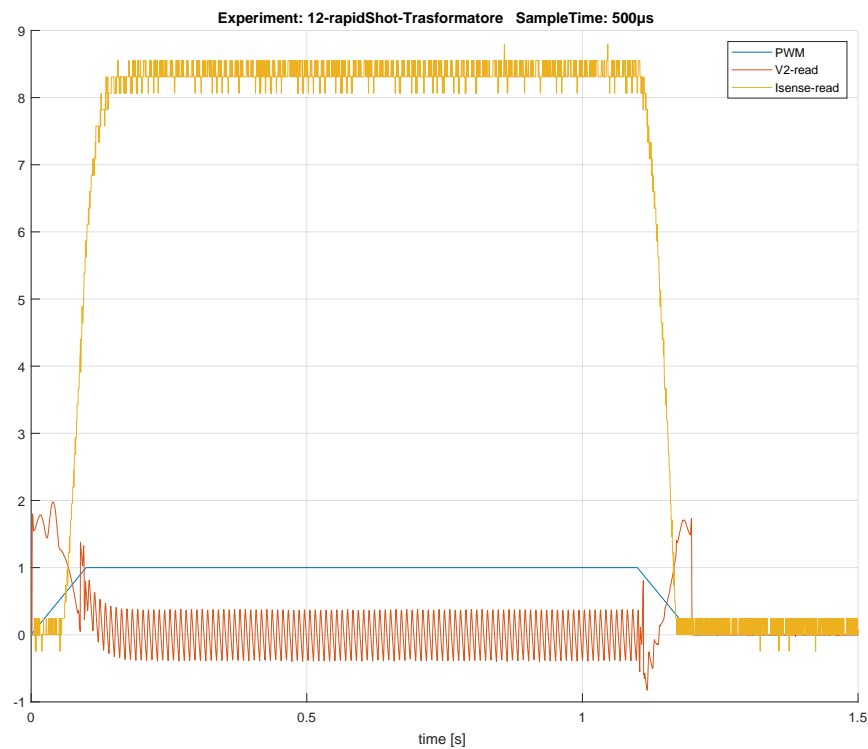


Figura 1.3: Onda Trapezoidale

Vediamo ora di analizzare i 2 segnali di controllo e come si riflettono sul sistema:

Deadzone

Nel prim

Disturbo della 50Hz

Capitolo 2

Catena di Acquisizione

Se lo si desidera, utilizzare questo spazio per inserire un breve riassunto di ciò che verrà detto in questo capitolo. Inserire solo i punti salienti.

In cosa consiste la catena di acquisizione

2.1 Schema di acquisizione

2.1.1 Sample and Hold alla frequenza di campionamento

Descrivivo come a ogni tic i dati vengono catturati e inviati al computer

2.1.2 Storage su file delle informazioni

2.2 Embedded Message Pack (EMP)

2.2.1 Metodo di codifica

2.2.2 Struttura del codice

2.2.3 Benchmark

2.3 Post Elaborazione con Matlab

Capitolo 3

Modello teorico

Se lo si desidera, utilizzare questo spazio per inserire un breve riassunto di ciò che verrà detto in questo capitolo. Inserire solo i punti salienti.

3.1 Modellazione Fisica

3.2 Funzione di trasferimento

Capitolo 4

Controllo in Retroazione dall'uscita

Se lo si desidera, utilizzare questo spazio per inserire un breve riassunto di ciò che verrà detto in questo capitolo. Inserire solo i punti salienti.

4.1 Modello di controllore

4.2 Esperimenti

Capitolo 5

Conclusioni e sviluppi futuri

Inserire qui le conclusioni trovate con la tesi, ed eventualmente eventuali idee per sviluppi futuri.

Appendice A

Arduino Code

5.1 Setup Registri

5.2 Generatore di Segnale

Per generare i segnali di controllo in Feed-Forward usati nel sistema, sono stati usati 2 diversi livelli di programmazione.

Un primo livello segnali di base, definiti su tutto \mathbb{R} , e usabili a piacere, e dei segnali compositi e periodici da mandare durante l'esperimento. Tutti i segnali sono pensati per andare da -100% <-> 100%, è compito dell'attuazione eliminare le deadzone e traslare il controllo al valore più opportuno

5.2.1 Segnali Base

Rampa

```
1 int ramp(uint64_t t, int vStart, uint64_t tStart, int vEnd, uint64_t tEnd) {
2     // Saturazione
3     if (t < tStart)
4         return vStart;
5     else if (t > tEnd)
6         return vEnd;
7     // Retta
8     unsigned int dt = t - tStart;
9     return vStart + int((vEnd - vStart) / float(tEnd - tStart) * dt);
10 }
```

Listing 5.1: Rampa Saturata

La rampa è descritta come una retta nell'intervallo di interesse, saturata prima e dopo il tempo desiderato

$$RampaSat(t) = \begin{cases} v_{start} + \frac{v_{end} - v_{start}}{t_{end} - t_{start}} * (t - t_{start}) & \forall t \in [t_{start}, t_{end}] \\ v_{start} & t < t_{start} \\ v_{end} & t > t_{start} \end{cases}$$

5.2.2 Segnali Composti

Onda Triangolare

```

1 int triangleSignal(uint64_t t, int msQuartPeriod) {
2     static uint64_t startTic = 0;
3     int dTic = t - startTic;
4     int pwm = 0;
5     if (dTic < ticConvert(msQuartPeriod))
6         pwm = ramp(dTic, 0, 0, 100, ticConvert(msQuartPeriod));
7     else if (dTic < (ticConvert(msQuartPeriod) * 3))
8         pwm = ramp(dTic, 100, ticConvert(msQuartPeriod), -100, ticConvert(msQuartPeriod)
9             * 3);
10    else if (dTic < (ticConvert(msQuartPeriod) * 4))
11        pwm = ramp(dTic, -100, ticConvert(msQuartPeriod) * 3, 0, ticConvert(msQuartPeriod)
12            * 4);
13    else {
14        pwm = 0;
15        startTic = t;
16    }
17    return pwm;
18 }
```

Listing 5.2: Onda Triangolare Periodica

Onda Trapezoidale

```

1 int rapidShot(uint64_t t) {
2     static uint64_t startTic = 0;
3     int pwmRapidShot;
4     long dTic = t - startTic;
5     if (dTic > t4) {
6         startTic = t;
7         pwmRapidShot = 0;
8         dTic = t - startTic;
9     }
10
11    if (dTic <= t1) {
12        pwmRapidShot = ramp(dTic, 0, 0, 100, t1);
13    } else if (dTic <= t2) {
14        pwmRapidShot = 100;
15    } else if (dTic <= t3) {
16        // falling ramp
17        pwmRapidShot = ramp(dTic, 100, t2, 0, t3);
18    } else if (dTic <= t4) {
19        pwmRapidShot = 0;
20    }
21    return pwmRapidShot;
22 }
```

Listing 5.3: Onda Trapezoidale Periodica

Appendice B

EMP Code

Appendice C

Matlab Post Elaborazione

Elenco delle figure

1.1	Driver Motori IBT-2 TopView & PinOut	6
1.2	Esperimento con Onda Triangolare	8
1.3	Esperimento con Onda Trapezoidale	9

Bibliografia

Handsontec: BTS7960 High Current 43A H-Bridge Motor Driver

IBT-2

Handsontec. *BTS7960 High Current 43A H-Bridge Motor Driver*. Datasheet. 18 Sep 2019. URL: <https://www.handsontec.com/dataspecs/module/BTS7960%20Motor%20Driver.pdf>.