



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DIPARTIMENTO DI INGEGNERIA DELL'ENERGIA ELETTRICA E
DELL'INFORMAZIONE

CORSO DI LAUREA MAGISTRALE IN INGEGNERIA DELL'ENERGIA
ELETTRICA

SVILUPPO DI UN SISTEMA POWER HARDWARE IN THE LOOP PER L'EMULAZIONE DI MACCHINE ELETTRICHE

Relatore

Dott. Ing. Gabriele Rizzoli

Presentata da

Leonardo Vita

Correlatore

Prof. Michele Mengoni

Dott. Ing. Luca Vancini

Sessione IV

Anno Accademico 2022/2023

Sommario

Indice delle figure	4
INTRODUZIONE	6
CAPITOLO 1 POWER HARDWARE IN THE LOOP	7
1.1 Hardware in the loop	7
1.2 Amplificatore di potenza e accoppiatore.....	9
1.3 Configurazione dell'emulatore	10
CAPITOLO 2 MODELLAZIONE DELL'EMULATORE	12
2.1 Specifiche del motore emulato	12
2.2 Modellazione matematica del motore sincrono	13
2.3 Modello tempo continuo motore sincrono trifase	17
2.3.1 Motore reale	17
2.3.2 Emulatore.....	27
2.3.3 Risultati della simulazione	29
2.4 Modello tempo discreto del motore sincrono	31
2.4.1 Discretizzazione di Tustin.....	32
2.4.2 Modello tempo discreto	34
2.4.3 Risultati della simulazione	38
CAPITOLO 3 DISPOSITIVO DI CONDIZIONAMENTO DEI SEGNALE	40
CAPITOLO 4 IMPLEMENTAZIONE DEL MODELLO SU DSP	44
4.1 Controllo dell'emulatore.....	46
4.1.1 Conversioni Analogico Digitali	47
4.1.2 Macchina a stati	48
4.1.3 Protezioni	49
4.2 Controllo dell'inverter sotto test	50
4.2.1 Conversione Digitale Analogico.....	50
CAPITOLO 5 RISULTATI SPERIMENTALI	51
CONCLUSIONI.....	59
Bibliografia.....	60

Indice delle figure

Figura 1: Panoramica generale dell'utilizzo delle FPGA, DSP o SoCP.....	8
Figura 2: dSpace HIL test system.....	8
Figura 3: Architettura generale PHIL	9
Figura 4: Modulo di espansione BOOSTXL-DRV8305EVM	11
Figura 5: Scheda di sviluppo Delfino F28379D con il modulo della figura precedente	11
Figura 6: Motore BTSS 1524 preso come riferimento	12
Figura 7: Circuito equivalente del motore	16
Figura 8: visione d'insieme del modello tempo continuo.....	17
Figura 9: Modello tempo continuo del motore reale.....	17
Figura 10: Stati della modulazione PWM unipolare a sette intervalli	18
Figura 11: Modulazione PWM.....	20
Figura 12: Modello matematico del motore.....	21
Figura 13: Modello circuitale implementato del motore	21
Figura 14: Modellazione del carico del motore	22
Figura 15: Margine di ampiezza e di fase nei diagrammi di Bode	24
Figura 16: Schema di principio di un regolatore PI	25
Figura 17: Schema di controllo in velocità di un azionamento	25
Figura 18: Schema del controllo in velocità del motore	26
Figura 19: Modello tempo continuo dell'emulatore	27
Figura 20: Controllo dell'emulatore	29
Figura 21: Risultati della simulazione del motore reale. In ordine sono mostrate tensioni, coppia, correnti e velocità.....	29
Figura 22: Risultati della simulazione del motore emulato. In ordine sono mostrate tensioni, coppia, correnti e velocità.....	30
Figura 23 : Confronto fra le correnti circolanti nel modello del motore reale (in verde) e di quello emulato (in rosso).....	30
Figura 24: Zoom sulle correnti presentate nella figura precedente	31
Figura 25: Area sottesa dalla funzione $y(t)$ nel periodo di campionamento	33
Figura 26: Schema del sistema discretizzato	34
Figura 27: Schema di controllo del DUT, discreto	35
Figura 28: Schema di controllo dell'EMU e carico meccanico, discreto	36
Figura 29: Induttori utilizzati per la realizzazione del dispositivo	37
Figura 30: Risultati della simulazione del motore reale	38
Figura 31: Confronto delle correnti del modello reale e quello emulato, discreto	38
Figura 32: Risultati della simulazione del motore emulato, discreto	38
Figura 33: Zoom sulle correnti della figura precedente.....	39
Figura 34: Configurazione filtro Sallen-Key passa basso	41
Figura 35: Circuito di misura delle correnti	41
Figura 36: Circuito di misura delle tensioni stellate.....	42
Figura 37: Risultati dell'analisi al piccolo segnale per lo stadio di misura delle tensioni	43
Figura 38: Scheda di condizionamento dei segnali realizzata	43
Figura 39: Dashboard del dispositivo	45

Figura 40: Schema di controllo dell'EMU implementato su DSP	46
Figura 41: Blocco di acquisizione delle misure	47
Figura 42: Conversione, offset e taratura delle misure	47
Figura 43: Macchina a stati.....	48
Figura 44: Blocco relativo alle protezioni	49
Figura 45: Schema di controllo del DUT	50
Figura 46: Blocco relativo al DAC	50
Figura 47: Setup dell'emulatore	51
Figura 48: Correnti nelle tre fasi	52
Figura 49: Corrente della fase a	52
Figura 50: Zoom sull'effetto di quantizzazione.....	52
Figura 51: Analisi in frequenza sulla forma d'onda della corrente.....	53
Figura 52: Ripple del DUT, corrente i_a filtrata a 8 kHz.....	53
Figura 53: Ripple dell'EMU, corrente i_a filtrata a 20 MHz.....	53
Figura 54: Tensione della fase a	54
Figura 55: Analisi in frequenza della tensione	54
Figura 56: Effetto della modulazione simmetrica	54
Figura 57: Confronto fra le PWM del DUT e dell'EMU	55
Figura 58: Banco prova del motore reale	56
Figura 59: Test di accelerazione svolto sul motore reale, rappresentate in ordine i_a , i_q e la velocità.....	56
Figura 60: Test di accelerazione svolto sull'emulatore, rappresentate in ordine i_a , i_q e la velocità.....	56
Figura 61: Test dell'inversione di velocità sul motore reale, in ordine sono mostrate i_a , i_q e la velocità.....	57
Figura 62: Test dell'inversione di velocità sull'emulatore, in ordine sono mostrate i_a , i_q e la velocità.....	57
Figura 63: Test di accelerazione svolto sul motore reale, rappresentate in ordine i_a , i_q e la velocità.....	57
Figura 64: Test di accelerazione svolto sull'emulatore, rappresentate in ordine i_a , i_q e la velocità.....	57
Figura 65: Presa di carico emulata, in ordine sono mostrate le correnti i_a e i_b , la velocità e la coppia	58
Figura 66: Perdita di carico emulata, in ordine sono mostrate le correnti i_a e i_b , la velocità e la coppia	58

INTRODUZIONE

I sistemi Power Hardware-In-the-Loop (PHIL) rappresentano una frontiera avanzata nella simulazione e nel test di componenti e sistemi elettrici ed elettronici. Fornendo un ponte tra la simulazione basata su modelli virtuali e il test fisico in laboratorio, i sistemi PHIL consentono di sperimentare su componenti reali condizioni operative canoniche o di guasto in un ambiente controllato e sicuro, dove scenari complessi possono essere replicati e analizzati in dettaglio.

Questi sistemi offrono un metodo unico per valutare l'interazione tra hardware fisico e modelli di sistema complessi, permettendo ai progettisti e agli ingegneri di identificare, analizzare e risolvere potenziali problemi prima dell'implementazione su larga scala. Le applicazioni PHIL possono essere usate per il test di inverter per impianti fotovoltaici, il controllo di dispositivi di accumulo energetico, la verifica di sistemi di protezione della rete elettrica, fino allo sviluppo di azionamenti per veicoli elettrici. La versatilità di queste architetture le rende uno strumento prezioso per l'industria e la ricerca. Infatti, offrono numerosi benefici, tra cui la riduzione dei costi e dei tempi di sviluppo, la capacità di testare in sicurezza condizioni operative estreme o pericolose, e la possibilità di iterare rapidamente i cicli di design basandosi su feedback immediato. Tuttavia, esistono anche sfide significative, come la necessità di garantire un'alta fedeltà della simulazione, la gestione della latenza nel sistema di loop, e l'integrazione di componenti fisici e modelli virtuali in un sistema coeso.

Questo elaborato verte sulla realizzazione di un'architettura PHIL che emuli un motore sincrono al fine di testare un inverter trifase. La metodologia utilizzata è la seguente:

- Modellazione tempo continuo dell'ambiente operativo
- Passaggio al modello tempo discreto
- Realizzazione dell'hardware per l'isolamento dei segnali fra l'emulatore e il controller e per la misura delle tensioni e correnti
- Implementazione del modello su scheda di sviluppo
- Test del funzionamento

CAPITOLO 1

POWER HARDWARE IN THE LOOP

Il funzionamento dell'architettura PHIL si basa su un ciclo di feedback in cui i segnali elettrici vengono scambiati tra il sistema fisico sotto test (*device under test*, DUT) e il modello simulato. Questo processo inizia con la modellazione di un dispositivo di cui si vuole emulare il comportamento. Il modello simulato viene poi convertito in segnali elettrici reali. Il passaggio dai segnali simulati, generati da una unità di controllo, ai segnali elettrici di potenza è mediato da due componenti chiave: l'interfaccia di potenza e il dispositivo di accoppiamento.

Il primo converte i segnali elettronici, derivanti dal dispositivo di controllo a bassa potenza, in segnali di potenza adatti a essere applicati ai dispositivi fisici sotto test.

L'accoppiatore è essenziale per garantire che i flussi di potenza siano trasmessi efficacemente tra l'ambiente simulato e il DUT. Ha quindi il compito di adattare dinamicamente i segnali di potenza dell'emulatore alle condizioni operative reali del DUT e deve assicurare la stabilità del ciclo di feedback.

La risposta del DUT viene misurata e riportata al modello simulato per aggiornare la simulazione in corso, completando così il ciclo di feedback.

Attraverso questi meccanismi, i sistemi PHIL consentono un'integrazione quasi trasparente tra il mondo fisico e quello simulato.

1.1 Hardware in the loop

Le architetture PHIL hanno origine dal concetto di hardware in the loop (HIL). Comunemente con il termine HIL si indica il sistema di loop fisico fra il dispositivo in prova e il simulatore in tempo reale. Questi si scambiano segnali di I/O analogici e digitali per ricreare le condizioni operative dell'apparecchio sotto test.

Il cuore del sistema è la modellazione dell'ambiente operativo, che deve essere il più accurata possibile per garantire una fedele emulazione del comportamento reale. Il modello deve essere in grado di rispondere agli input dal sistema di controllo hardware e aggiornare lo stato della simulazione abbastanza rapidamente da mantenere il passo con il tempo reale. In altre parole, le variabili del modello devono evolvere abbastanza velocemente da apparire reali al sistema testato. Si necessita, per il dispositivo che opera in tempo reale, di un tempo di esecuzione dei calcoli molto inferiore a quello della controparte sotto test. Le apparecchiature solitamente utilizzate per questi scopi sono le FPGA, dall'inglese *Field Programmable Gate Array*, oppure i DSP ad alte prestazioni, *Digital Signal Processor*. La principale differenza fra i due sta nell'architettura interna. Il primo è un dispositivo basato su una matrice di blocchi logici configurabili, collegati tramite una rete di interconnessione programmabile. Questi dispositivi consentono agli sviluppatori di configurare o programmare le funzionalità dell'hardware dopo la fabbricazione, il che li rende estremamente flessibili e adatti a una vasta gamma di applicazioni. Un DSP è un tipo di microprocessore specializzato nell'elaborare segnali digitali in modo efficiente e veloce. Questi dispositivi sono progettati per eseguire una vasta gamma di operazioni matematiche ad alta velocità, la loro architettura interna è però fissata dal costruttore. Dunque, le FPGA sono utili nel caso in cui vi sono molte operazioni parallelizzabili e semplici da eseguire, mentre i DSP sono preferiti laddove i calcoli sono pochi e complessi. Vi sono anche soluzioni ibride, come i System on a Programmable Chip (SoPC), in cui sullo stesso semiconduttore sono presenti un DSP e una FPGA.

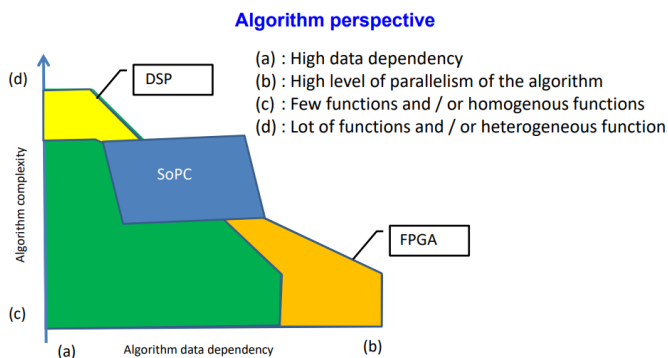


Figura 1: Panoramica generale dell'utilizzo delle FPGA, DSP o SoPC.



Figura 2: dSpace HIL test system

1.2 Amplificatore di potenza e accoppiatore

Per testare un dispositivo di elettronica di potenza, è necessario un apparecchio che trasformi i segnali provenienti dal modello matematico in segnali di potenza. Riprodurre un segnale elettrico variabile nel tempo come quello in uscita dal modello matematico non è semplice, ma può essere facilmente ottenuto come la media di un'onda quadra in un determinato periodo. Ciò è svolto da un convertitore statico di potenza DC-AC (inverter). Quest'ultimo trasforma una tensione continua (DC), che funge da alimentazione primaria per il processo di conversione, in una tensione alternata (AC).

Il cuore dell'inverter è rappresentato dallo stadio di commutazione, composto da transistor o altri dispositivi (come IGBT o MOSFET), che hanno il compito di aprire e chiudere i circuiti in modo rapido e controllato. La configurazione di questi più utilizzata è quella del ponte ad H trifase, ma esistono molte altre strutture per soddisfare specifiche esigenze progettuali.

Nella maggior parte dei casi, i commutatori sono pilotati da segnali generati da una logica di modulazione a larghezza di impulso (PWM). Questo approccio consente di variare la durata degli impulsi di tensione nel tempo, mimando così la forma e le caratteristiche volute dal modello. Dovendo mantenere il passo con il tempo reale, la finestra di tempo su cui mediare gli impulsi deve essere più piccola rispetto a quella usata dal DUT. Commutatori di nuova generazione, sempre più veloci nell'aprire e chiudere il canale, rendono possibili frequenze di commutazione elevate adatte allo scopo. Tuttavia, il segnale generato dall'operazione di commutazione contiene spesso un numero significativo di armoniche, dovute alla natura non perfettamente sinusoidale dell'onda prodotta.

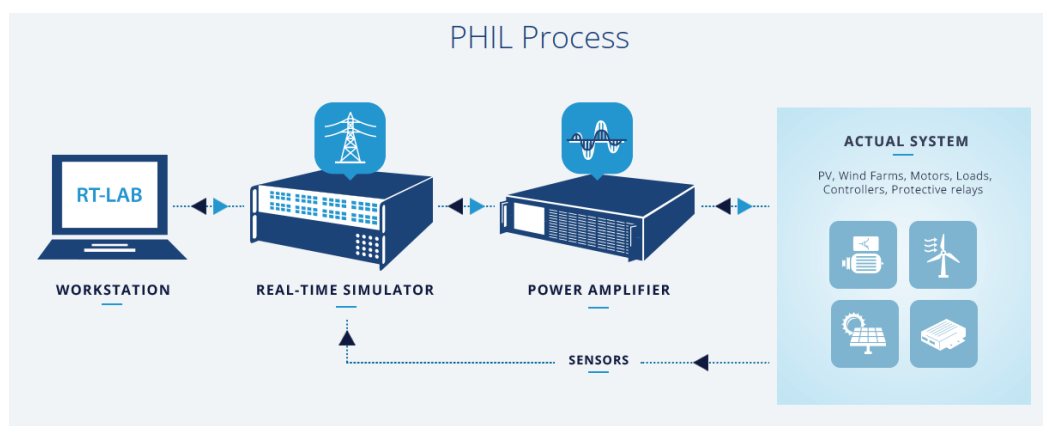


Figura 3: Architettura generale PHIL

La corrente in uscita da un normale inverter a tensione impressa solitamente dipende dal carico applicato. Nel caso di una configurazione PHIL, se il DUT fosse direttamente connesso all'emulatore, non si avrebbe nessun tipo di controllo sulle correnti che si scambiano. È necessario infatti un accoppiatore. Quest'ultimo è solitamente una resistenza in serie ad un induttore. Ciò fa sì che si possa determinare il flusso di corrente sapendo le tensioni applicate. Inoltre, l'induttore svolge anche un'azione filtrante per le correnti che altrimenti sarebbero più distorte.

1.3 Configurazione dell'emulatore

Analizzando il modello circuitale di un motore trifase, ogni fase è descritta dalla serie di un'induttanza, una resistenza e un generatore di tensione che rappresenta le forze elettromotrici interne. L'emulazione del motore avrà la stessa forma, l'accoppiatore sarà un'induttanza in serie ad una resistenza e l'emulatore la FEM del motore. Bisogna ricordare che l'avvolgimento del motore è chiuso in cortocircuito, ciò significa che la somma delle tre correnti circolanti nelle fasi è sempre uguale a zero. In un inverter non è possibile formare fisicamente un collegamento delle fasi a stella, ma tramite il controllo si può ottenere un cortocircuito virtuale imponendo il vincolo di somma nulla delle correnti.

Il controllo è operato da un processore montato su una scheda di sviluppo della Texas Instruments, la Delfino F28379D.

La scelta di un DSP al posto di una FPGA è motivata dalla facilità di implementazione della prima. Per semplicità, lo stesso processore controlla sia il DUT che l'emulatore, ma, per non falsare la simulazione, i due sistemi sono indipendenti.

Gli inverter sono moduli di espansione della scheda di sviluppo, denominati BOOSTXL-DRV8305EVM, e sono uguali. La configurazione è quella del ponte H trifase con commutatori in tecnologia Si.

I due convertitori devono essere totalmente isolati l'uno dall'altro per evitare il ricircolo di correnti dovute ad una differenza di tensione che si vedrebbe fra i due poli negativi dei bus DC in ingresso ad essi. Studiando la loro configurazione interna, si nota che i

due moduli non sono full isolated, ovvero non vi è una completa isolazione tra i segnali a bassa tensione provenienti dal controllo e quelli di potenza. Il ground del livello segnale è direttamente connesso al polo negativo dell'alimentazione. Connettendo entrambe le espansioni sulla stessa scheda, i ground del livello di potenza sarebbero collegati fra loro. È necessario realizzare un'apparecchiatura che provveda all'isolamento dei segnali. Su di essa sono presenti dei sensori ad effetto Hall per la misura delle correnti, seguiti da filtri analogici per ripulire il segnale dalle armoniche indesiderate. Inoltre, vi sono dei sensori di tensione per misurare le tensioni stellate e quella in ingresso all'emulatore, anch'esse filtrate analogicamente.

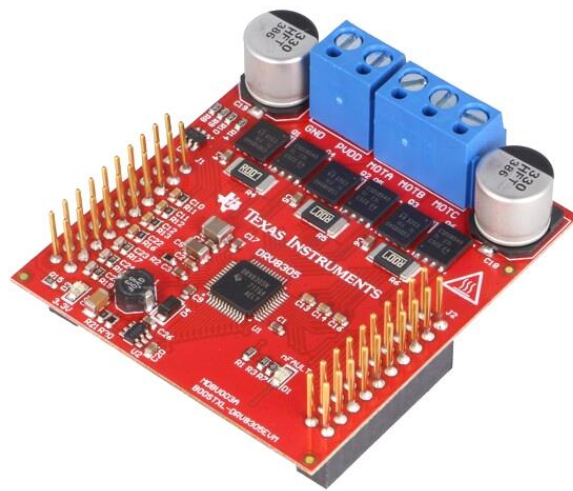


Figura 4: Modulo di espansione BOOSTXL-DRV8305EVM

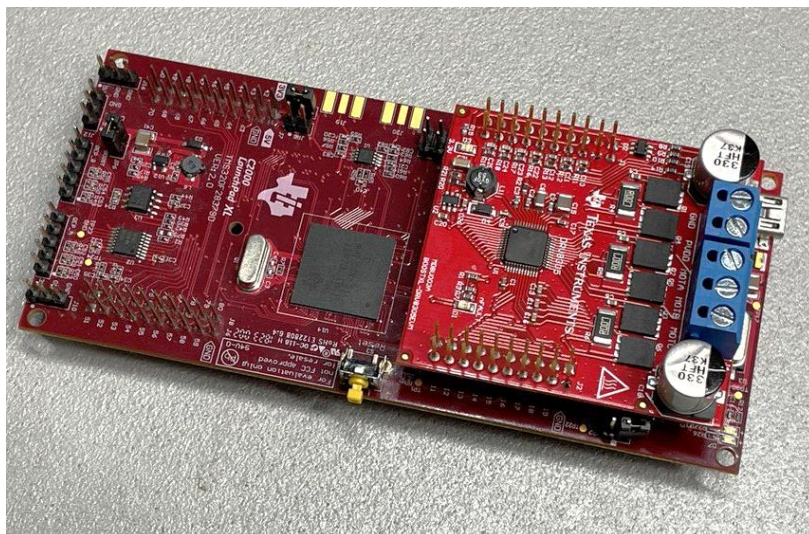


Figura 5: Scheda di sviluppo Delfino F28379D con il modulo della figura precedente

CAPITOLO 2

MODELLAZIONE DELL'EMULATORE

In questo capitolo si andrà ad analizzare il modello matematico implementato nell'emulatore.

I modelli sono realizzati nell'ambiente di simulazione Simulink e la loro struttura di base di è composta di due parti:

- la prima consistente in un azionamento con motore fisico per avere un riferimento di come dovrebbe funzionare il sistema reale
- la seconda con motore simulato da un convertitore elettrico

2.1 Specifiche del motore emulato

Il motore, sincrono isotropo, emulato è il TEM BTSS 1524.



Figura 6: Motore BTSS 1524 preso come riferimento

DATI DEL MOTORE	SIMBOLO	VALORE	UNITÀ DI MISURA
COPPIA ROTORE BLOCCATO ΔT 100°C IN ARIA	C_m	0.6	Nm
COPPIA DI PICCO	$C_{m,max}$	2.25	Nm
VELOCITA' NOMINALE	N_n	1500	rpm
VELOCITA' MAX	$N_{n,max}$	5000	rpm
POTENZA NOMINALE IN ARIA ΔT 100°C	P_n	95	W
CORRENTE A VELOCITA' NOM. ΔT 100°C IN ARIA	I_n	2.85	Arms
CORRENTE DI PICCO	I_{max}	10.8	Arms
MOMENTO D' INERZIA ROTORICO	J	0.380	$kg \cdot m^2 \cdot 10^{-4}$
F.E.M CONCATENATA	K_e	0.123	V·s
COSTANTE DI COPPIA	K_T	0.21	Nm/A
RESISTENZA CONCATENATA	R_w	0.74	Ω
INDUTTANZA CONCATENATA	L_w	1.4	mH
FREQUENZA	f	100	Hz
RENDIMENTO A POTENZA NOM. ΔT 100°C	η	93	%
N° POLI	p	8	

2.2 Modellazione matematica del motore sincrono

Il motore utilizzato è di tipo sincrono isotropo, a magneti superficiali. Come ampiamente noto, è costituito da uno statore cilindrico nel quale si dispone un avvolgimento trifase simmetrico. Questo ha il compito di generare una distribuzione,

sinusoidale, di forza magnetomotrice al traferro. Mentre il rotore è dotato di magneti permanenti disposti sulla superficie del nucleo laminato.

Per descrivere il suo funzionamento si può partire dalle equazioni generali di bilancio delle tensioni V_a, V_b, V_c , che con la convenzione di segno degli utilizzatori risultano:

$$\begin{aligned} V_a &= Ri_a + \frac{d\phi_a}{dt} & \phi_a &= L_s i_a + \Phi_e \cos(\theta) \\ V_b &= Ri_b + \frac{d\phi_b}{dt} & \phi_b &= L_s i_b + \Phi_e \cos\left(\theta - \frac{2}{3}\pi\right) \\ V_c &= Ri_c + \frac{d\phi_c}{dt} & \phi_c &= L_s i_c + \Phi_e \cos\left(\theta - \frac{4}{3}\pi\right) \end{aligned}$$

(1)

Dove i_a, i_b, i_c sono le correnti che percorrono le tre fasi, ϕ_a, ϕ_b, ϕ_c sono i flussi magnetici concatenati con ciascuna fase ed R è la resistenza di fase, che si suppone uguale per le tre fasi. Nelle equazioni dei flussi compare L_s , un coefficiente di autoinduttanza apparente che coincide con l'induttanza sincrona. Φ_e è il valore di picco del flusso concatenato con un avvolgimento statorico, dovuto al campo di eccitazione prodotto dai magneti del rotore. È poi moltiplicato per il coseno dell'angolo del flusso rispetto alla posizione delle tre fasi statoriche.

Per semplificare l'utilizzo di queste equazioni nei sistemi di controllo, si passa, con l'utilizzo dei vettori di spazio, ad un modello bifase (trasformazione di Clark). Successivamente, si pone il modello bifase in un sistema di riferimento rotante, di assi d-q, solidale con il rotore e con asse d orientato secondo la direzione del campo di eccitazione. Tale sistema risulta essere orientato di un angolo θ rispetto a quello statorico; dunque, basterà applicare una trasformazione di rotazione (trasformata di Park) al modello bifase.

Con queste trasformazioni, le equazioni del motore in forma vettoriale risultano essere:

$$\bar{V} = R\bar{i} + \frac{d\bar{\Phi}}{dt} + j\omega\bar{\Phi} \quad \bar{\Phi} = L_s\bar{i} + \Phi_e$$

(2)

in cui compare la velocità del rotore ω in angoli elettrici, ovvero $\omega = p\omega_m$ con p numero di poli del motore.

Sostituendo nella prima la seconda ed esplicitandole nelle componenti d-q, diventano:

$$V_d = Ri_d + L_s \frac{di_d}{dt} - \omega L_s i_q$$

$$V_q = Ri_q + L_s \frac{di_q}{dt} + \omega L_s i_d + \omega \phi_e$$

(3)

Tramite un bilancio di potenze, è possibile ricavare la coppia meccanica erogata all'albero

$$C_m = \frac{3}{2} p i_q \phi_e$$

(4)

Osservando l'equazione del motore, è facilmente intuibile come ottenere uno schema circuitale. Utilizzando la convenzione del generatore, ogni fase può essere modellata come la serie di una resistenza, un'induttanza e un generatore di tensione, che rappresenta le forze elettromotrici interne. Considerando che gli avvolgimenti del motore sono collegati a stella il circuito risulta essere quello in *figura 7*.

Spesso le specifiche del motore fornite dal costruttore non comprendono tutti i dati necessari per la modellazione dello stesso, quelli mancanti possono essere ricavati.

L'induttanza sincrona è ricavata dall'induttanza concatenata riportata nelle specifiche, che è misurata ai capi di due fasi. Immaginando di alimentare, con un generatore di tensione, solo due delle tre fasi del motore, il flusso creato sarà espresso dalle *equazioni (1)*. Prendiamo, ad esempio, la fase *a* e la *b* e applichiamo una tensione ai loro capi come appena descritto, la corrente che circolerà nella fase *a* sarà uguale e opposta a quella nella fase *b* ($i_a = -i_b$). Il flusso, senza la presenza del rotore, sarà espresso da

$$\phi_a - \phi_b = L_s i_a - L_s i_b = 2L_s i_a = L_w i_a$$

(5)

Dunque, l'induttanza sincrona presente nelle equazioni del motore risulta essere

$$L_s = \frac{L_w}{2}$$

(6)

Analogo ragionamento può essere fatto per la resistenza concatenata

$$R_w = 2R$$

(7)

Il flusso concatenato ϕ_e è ricavato dalla costante di tensione. Tale costante è definita come il rapporto fra la coppia e la corrente di asse q

$$K_e = \frac{C_m}{i_q} = \frac{3}{2} p \phi_e$$

(8)

Manovrando l'equazione e considerando la corrente efficace, è possibile ottenere il flusso concatenato

$$\phi_e = \frac{2}{3p} \frac{K_e}{\sqrt{2}} = 0.0247 \text{ Wb}$$

(9)

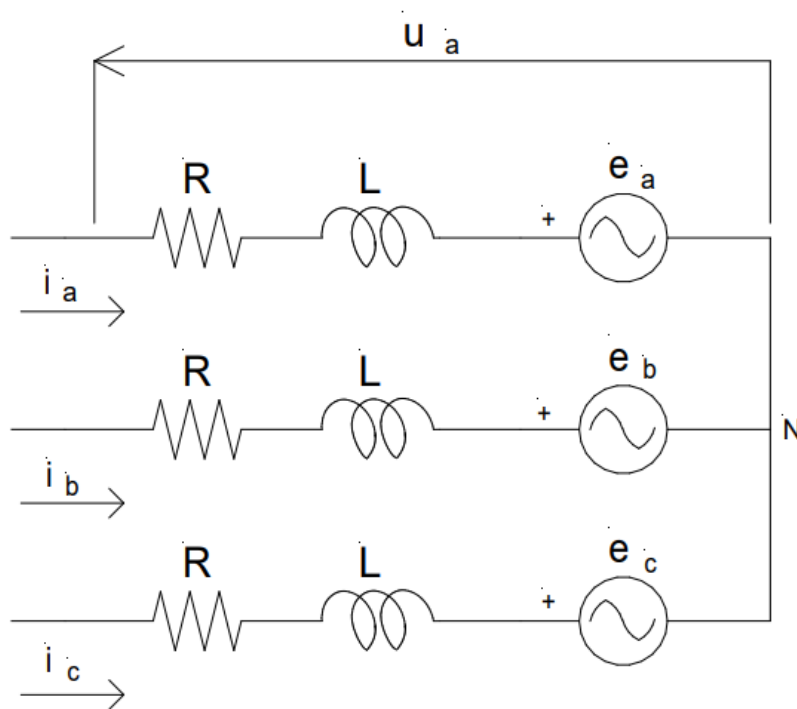


Figura 7: Circuito equivalente del motore

2.3 Modello tempo continuo motore sincrono trifase

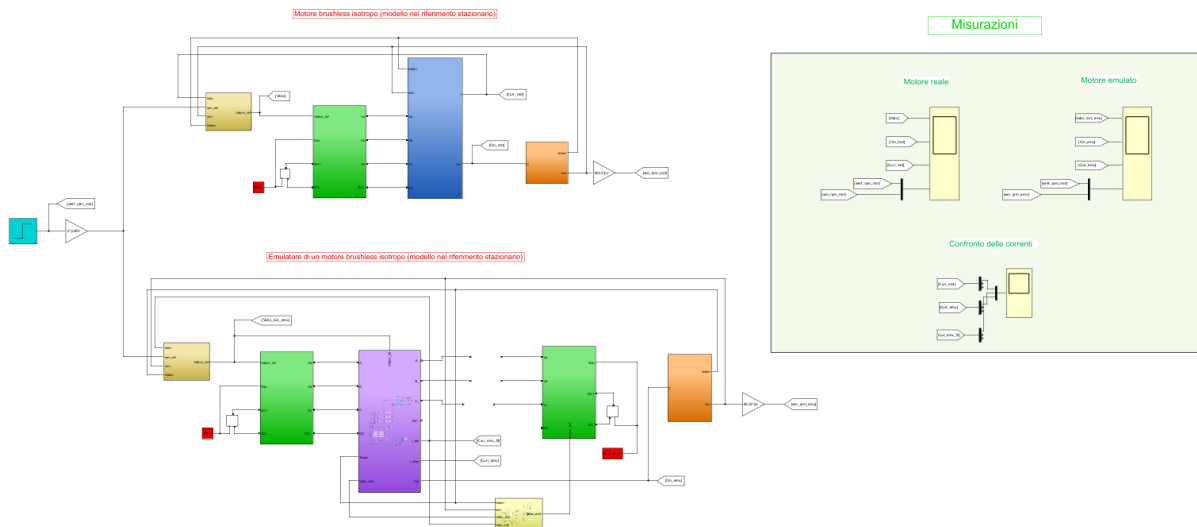


Figura 8: visione d'insieme del modello tempo continuo

Il primo modello del sistema è stato realizzato con un approccio tempo continuo, che semplifica l'implementazione non dovendo tenere in conto dei ritardi e dei tempi di esecuzione

2.3.1 Motore reale

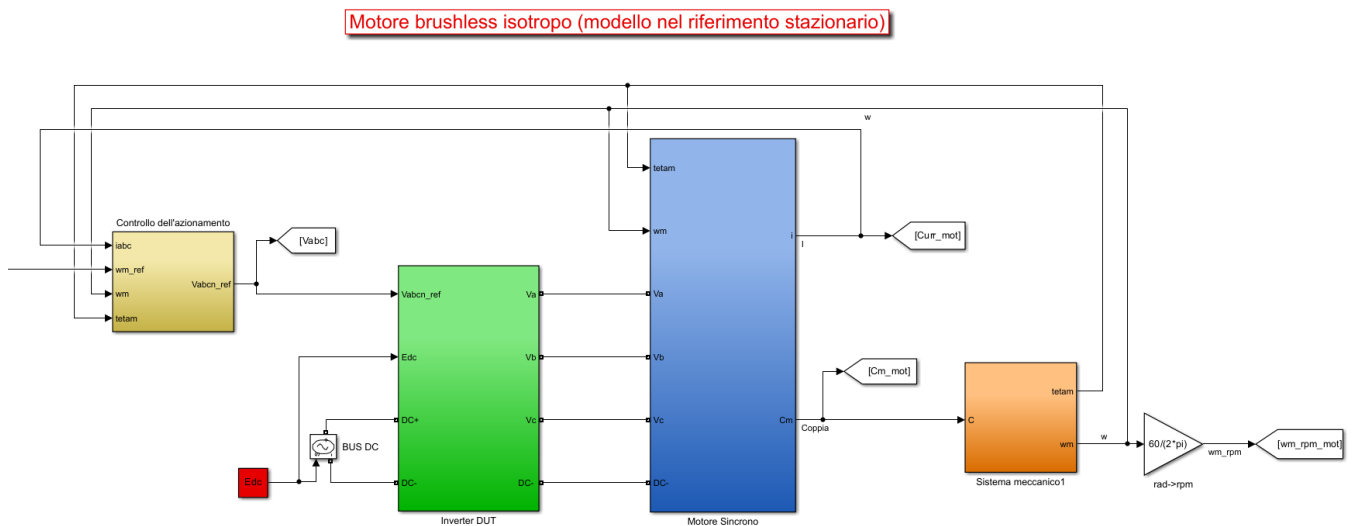


Figura 9: Modello tempo continuo del motore reale

Lo schema è composto da un motore sincrono alimentato da un inverter e con un carico inerziale dipendente dalla coppia.

Nell'ambiente di simulazione sono presenti due tipi di segnali, i segnali di controllo a bassa tensione e i segnali elettrici di potenza.

Il primo blocco verde mostrato nella *figura 9* è l'inverter di cui testare il funzionamento, che è alimentato da una tensione $E_{DC} = 30\text{ V}$. Questo è un chopper quattro quadranti i cui interruptori sono pilotati da una tensione di riferimento proveniente dalla logica di controllo. La tecnica di modulazione utilizzata è denominata PWM unipolare a sette intervalli simmetrica.

Nella modulazione PWM unipolare, gli interruptori del convertitore sono controllati in modo tale da generare tensioni di uscita con polarità positiva o nulla rispetto al terminale di riferimento comune, a differenza della modulazione bipolare dove le tensioni possono essere sia positive che negative.

La versione a sette intervalli di questa tecnica si riferisce alla sequenza di commutazione degli interruptori, che prevede sette configurazioni. Imponendo i vincoli di avere una sola commutazione di ramo per ogni cambio di configurazione e che la sequenza di commutazione deve essere speculare rispetto alla mezzeria, una delle possibili soluzioni è quella della rappresentata in *figura 10*. La simmetria è data dal fatto che i tempi in cui sono applicate le due configurazioni che producono in uscita il vettore nullo sono uguali.

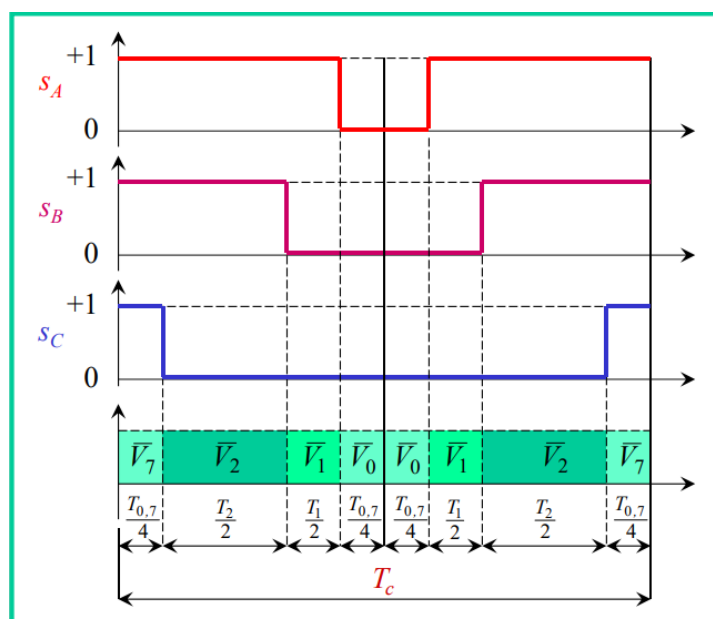


Figura 10: Stati della modulazione PWM unipolare a sette intervalli

Per ottenere questo risultato si deve spostare le modulanti in modo che risultino centrate rispetto all'intervallo [0, 1]. Le modulanti ottenute avranno quindi le seguenti equazioni:

$$\begin{cases} m_A = m_0 + \frac{V_{AN,ref}}{V_{dc}} \\ m_B = m_0 + \frac{V_{BN,ref}}{V_{dc}} \\ m_C = m_0 + \frac{V_{CN,ref}}{V_{dc}} \end{cases} \quad (10)$$

$$m_0 = \frac{1}{2} \left[1 - \min \left(\frac{V_{AN,ref}}{V_{dc}}, \frac{V_{BN,ref}}{V_{dc}}, \frac{V_{CN,ref}}{V_{dc}} \right) - \max \left(\frac{V_{AN,ref}}{V_{dc}}, \frac{V_{BN,ref}}{V_{dc}}, \frac{V_{CN,ref}}{V_{dc}} \right) \right] \quad (11)$$

Dove V_{dc} è la tensione sul bus DC, $V_{AN,BN,CN,ref}$ le tensioni di riferimento derivanti dal controllo e m_0 il termine che trasla le modulanti.

Questo metodo riduce la distorsione armonica totale nell'uscita e minimizza le perdite di commutazione migliorando l'efficienza complessiva del sistema.

Le modulanti sono confrontate con una portante triangolare di frequenza 5 kHz. Il risultato del confronto determina lo stato degli interruttori. In particolare, quando la portante è minore del valore modulante, lo stato dell'interruttore superiore è alto e quello inferiore basso, mentre nella condizione di portante maggiore della modulante, avviene il contrario. In questo caso non si sono tenuti in considerazione i tempi morti che permettono l'apertura e chiusura dei canali, necessari nell'implementazione reale.

Il motore modellato è quello descritto nel paragrafo precedente. Il blocco del motore è composto da una stella trifase con neutro flottante, proprio come gli avvolgimenti di una macchina elettrica. Su due delle tre fasi insistono due generatori di corrente e quella che circolerà nella terza fase è ottenuta dal vincolo di somma nulla.

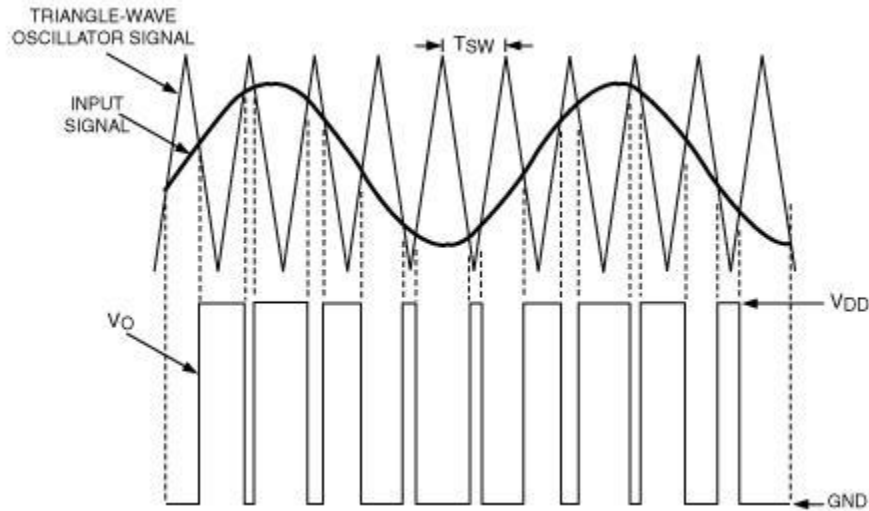


Figura 11: Modulazione PWM

La corrente che deve essere prodotta dai due generatori è ricavata dalle equazioni che descrivono il comportamento del motore.

$$I_d = \frac{V_d + \omega_e L_s I_q}{s L_s + R_s}$$

$$I_q = \frac{V_q - \omega_e L_s I_d - \omega_e \phi_e}{s L_s + R_s}$$

(12)

L_s ed R_s rappresentano l'induttanza e la resistenza dell'avvolgimento statorico, mentre ϕ_e e ω_e il flusso creato dai magneti rotorici e la velocità elettrica del rotore.

Le equazioni (12) sono funzione delle tensioni imposte dal convertitore, misurate e trasformate nel sistema di assi d-q. Il segnale logico di tali correnti è inviato al controllo del motore.

Inoltre, dalla corrente di asse q, è ricavata la coppia generata dal motore tramite l'equazione (4).

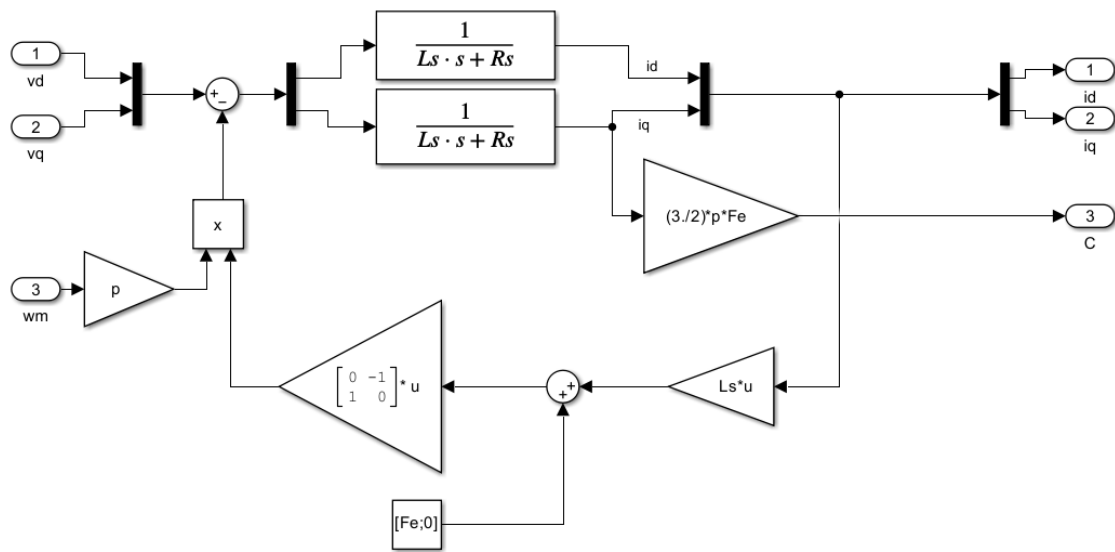


Figura 12: Modello matematico del motore

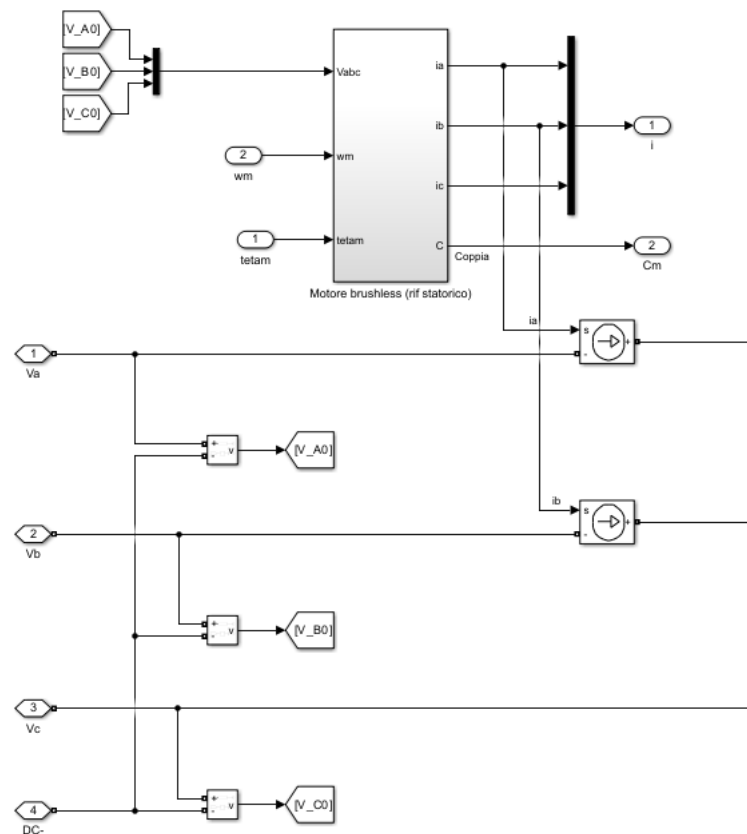


Figura 13: Modello circuitale implementato del motore

L'ultimo blocco è relativo al carico del motore. Presa in ingresso la coppia generata dal motore, la funzione di trasferimento che descrive il carico è

$$G_{load} = \frac{\omega_m}{C_m} = \frac{1}{sJ + b + K_C} \quad (13)$$

Il termine $J = J_{mot} + J_{load}$ è il momento d'inerzia totale del carico, comprensivo di un termine che considera l'inerzia del rotore a vuoto e di uno relativo al carico, supposto essere $J_{load} = 3 * J_{mot}$. Il coefficiente di attrito, b , è impostato a 10^{-3} . La costante di coppia resistente, K_C , è stata dimensionata per avere, in condizioni di regime del motore, una velocità pari a quella nominale.

$$K_C = \frac{C_m}{\omega_{nom}} - b \quad (14)$$

La *funzione* (13) restituisce la velocità meccanica del rotore; da cui, tramite integrazione della stessa, si ottiene la posizione angolare del rotore θ_m .

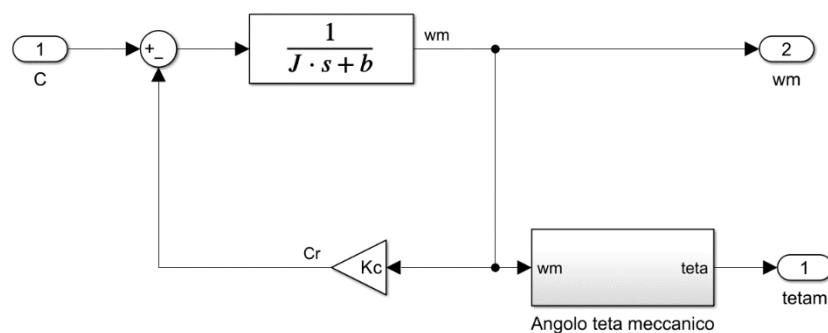


Figura 14: Modellazione del carico del motore

Il controller riceve in entrata un riferimento di velocità da inseguire, la velocità attuale del motore, l'angolo di riferimento e le correnti misurate. L'errore, fra la velocità di riferimento e quella attuale del motore, è dato in ingresso ad un regolatore proporzionale integrale (PI) che a sua volta produce un riferimento di corrente. Quest'ultimo è confrontato con le correnti misurate e l'errore è regolato da un secondo PI che restituirà i riferimenti di tensione da applicare al motore tramite l'inverter.

Non essendo necessario provare condizioni di sovraccarico, il motore opera in orientamento di campo. Ciò significa che non viene mai portata la macchina in condizione di sovraccarico. In particolare, tale tecnica impone l'utilizzo del vettore di corrente con modulo minore, che nel caso di un motore sincrono isotropo è sempre un vettore sull'asse q.

La scelta dei parametri dei regolatori è di fondamentale importanza per ottenere un controllo efficace. Un regolatore proporzionale integrale, mostrato in *figura 16*, è un sistema di controllo in retroazione che ha la funzione di portare l'errore in ingresso a zero, generando un opportuno segnale di riferimento per il sistema da controllare. L'equazione del regolatore è la seguente

$$u(t) = K_p \text{err}(t) + \frac{K_i}{s} \text{err}(t) \quad (15)$$

Dove K_p e K_i sono i parametri del regolare e s l'operatore della trasformata di Laplace. Il primo svolge un'azione proporzionale sull'errore e quindi regola istante per istante. Il secondo un'azione integrale che tiene in conto degli istanti passati. Solitamente si prende in considerazione la funzione di trasferimento e quindi possiamo riscrivere l'equazione (15) come

$$\frac{u(t)}{\text{err}(t)} = K_p + \frac{K_i}{s} \quad (16)$$

Raccogliendo K_p e definendo $\tau = \frac{K_p}{K_i}$ otteniamo

$$R_{PI} = K_p \left(1 + \frac{1}{\tau s} \right) = \frac{K_i}{\tau} \left(\frac{1 + s \tau}{s} \right) \quad (17)$$

Si può notare che τ è la costante di tempo dello zero della funzione di trasferimento. La scelta di τ tendenzialmente ricade sul valore che porta alla compensazione del polo più vicino all'origine nella funzione di trasferimento del sistema da controllare. Essendo due i parametri del regolatore, guardando la (17), bisogna fissare anche K_i . La scelta della costante integrativa è fatta in modo da garantire la stabilità del sistema. Il giusto compromesso fra la prontezza con cui risponde il sistema e l'assenza di oscillazioni a

regime, o sovraelongazioni in fase transitoria, si ottiene solitamente scegliendo il margine di fase della funzione nell'intorno dei 75° . Graficando la funzione di trasferimento del sistema sui diagrammi di Bode è possibile visualizzare molte di queste informazioni.

La funzione di risposta armonica di un sistema può essere graficata con i diagrammi di Bode, diagramma delle ampiezze e diagramma delle fasi. Sul primo viene rappresentato il modulo in funzione della pulsazione, che solitamente sono espressi in scala logaritmica. Invece, sul secondo è graficata la fase in funzione della pulsazione, la prima in scala lineare e la seconda in scala logaritmica. Sono definiti anche i margini di stabilità, quello di ampiezza (M_A) e quello di fase (M_F) mostrati in *figura 15*. Il margine di ampiezza è definito come l'inverso del modulo del guadagno di anello alla pulsazione corrispondente alla fase $-\pi$. Il margine di fase è l'angolo che occorre sottrarre alla fase (normalmente negativa) per ottenere il valore $-\pi$, misurato alla pulsazione in cui il guadagno di anello è unitario (detta pulsazione attraversamento). In altre parole, di quanti gradi la fase deve essere diminuita per raggiungere -180° .

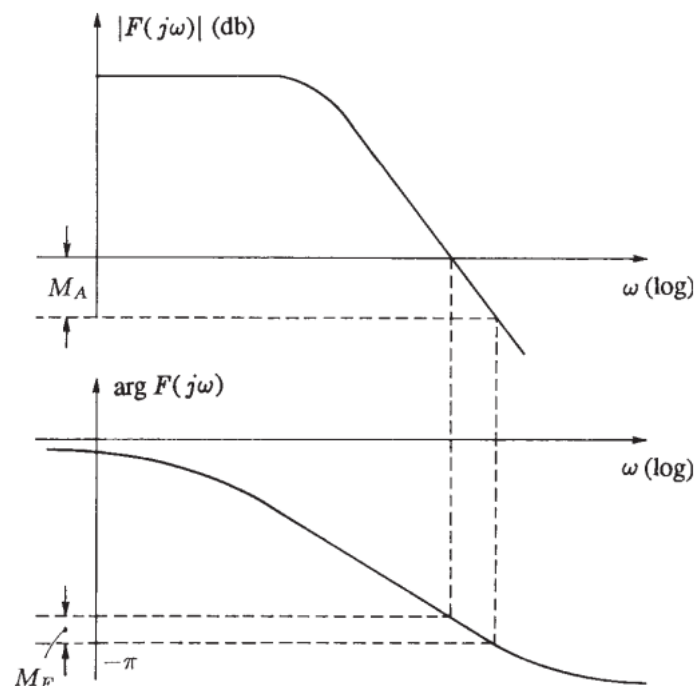


Figura 15: Margine di ampiezza e di fase nei diagrammi di Bode

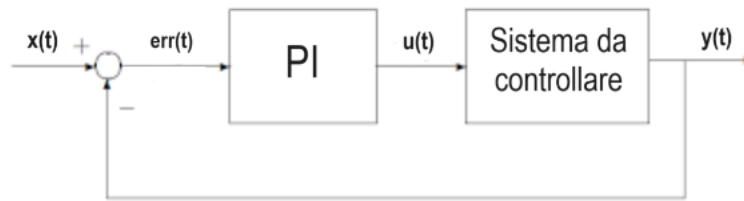


Figura 16: Schema di principio di un regolatore PI

Nel caso in questione il sistema da controllare è riassumibile dalla *figura 17*, anche se non esplicitati ci sono da tenere in considerazione anche i filtri, relativi alle misure di tensione e corrente, che introducono ritardi. Il convertitore ha un polo con costante di tempo uguale al periodo di commutazione, quindi molto lontano dall'origine. Lo stesso si può dire per il filtro di corrente. Mentre la funzione di trasferimento del motore, omettendo la FEM, può essere scritta come

$$G_{mot} = \frac{I(t)}{V(t)} = \frac{1}{R + sL_s} = \frac{1}{R(1 + s\tau_{mot})}$$

(18)

Dove è stato definito $\tau_{mot} = \frac{L_s}{R}$ come costante di tempo del motore.

Per il PI dell'anello di corrente è stata scelta una costante di tempo di tempo uguale a quella del motore e un $K_{i,I} = 412$. Tale valore è stato ottenuto con un tool di MatLab, chiamato Control System Designer, che permette di visualizzare i diagrammi di Bode del sistema implementato e calcolare il valore più opportuno per il K_i del PI.

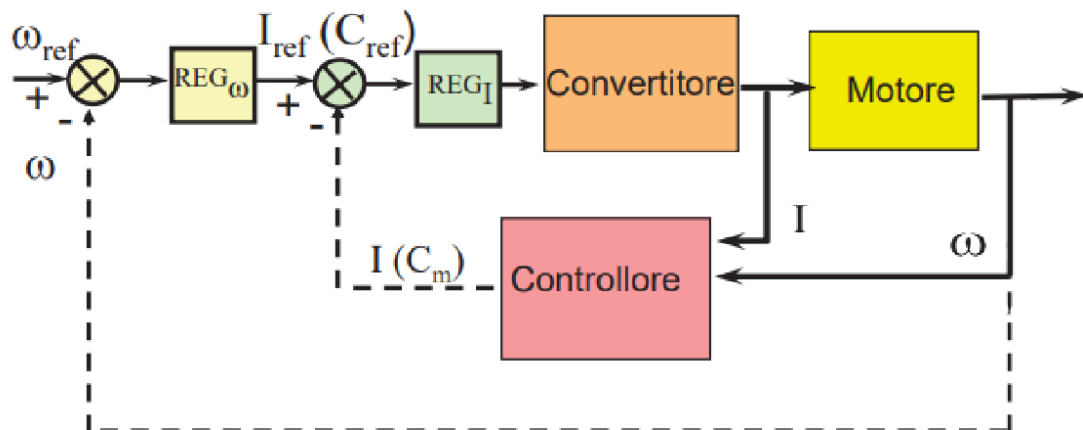


Figura 17: Schema di controllo in velocità di un azionamento

È presente anche una compensazione dei termini relativi al flusso. Osservando le equazioni (3) del motore si può notare che $-\omega_e L_s I_q$, per la prima, e $\omega_e L_s I_d + \omega_e \phi_e$, per la seconda, hanno la dimensione di una tensione e possono essere compensate in avanti.

[illegible]

26

2.3.2 Emulatore

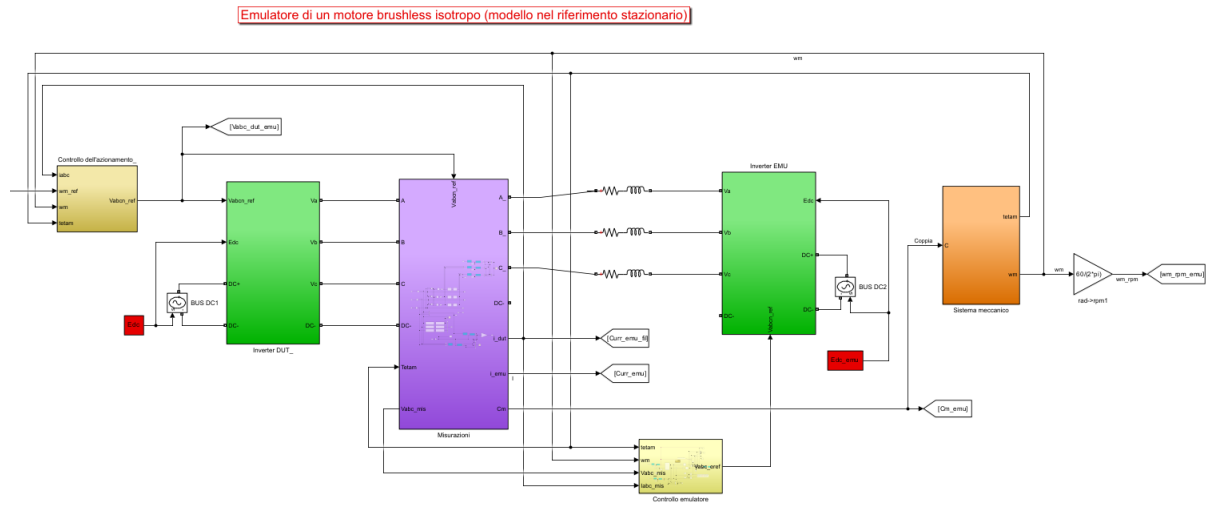


Figura 19: Modello tempo continuo dell'emulatore

Lo schema con l'emulatore (EMU) è molto simile al precedente. L'unico blocco che cambia è quello del motore che è stato sostituito da un convertitore.

Come si può notare dalla corrispondenza dei colori dei blocchi nella *figura 19*, per l'inverter EMU, è stato utilizzato lo stesso modello del DUT. Questo perché per la realizzazione fisica sono previsti due inverter uguali. Ciò che cambia è il controllo, la frequenza di commutazione, impostata a 25 kHz sulla portante, e l'alimentazione $E_{DC,EMU} = 2E_{DC}$. L'EMU esige una tensione maggiore del DUT dal momento che deve imporre una caduta di tensione sull'accoppiatore atta a ricreare le correnti presenti in un motore.

Il controllo, a partire dalle tensioni e correnti misurate, deve restituire un riferimento di tensione per l'EMU. In questo caso si necessita di filtrare le misure, altrimenti entrando in equazioni di controllo produrrebbero riferimenti distorti. Nel caso precedente, le correnti che il controllo del DUT utilizzava erano direttamente prese dal modello matematico del motore e quindi non affette da disturbi causati dalla modulazione.

Il filtraggio, sia sulle tensioni che sulle correnti, è compiuto da due filtri analogici del primo ordine uguali in cascata, risultando complessivamente un filtro del secondo ordine. Le costanti di tempo utilizzate per i filtri del primo ordine sono $10^{-4}s$, per quello di corrente, e $10^{-3}s$, per quello di tensione.

Le tensioni misurate sono usate dal modello matematico del motore, il cui blocco contenente le equazioni del motore è uguale a quello visto precedentemente. Le correnti circolanti nel motore che sono state calcolate, vengono confrontate con le correnti misurate e l'errore è regolato da un PI.

Il sistema retroazionato in cui lavora il regolatore è composto da un convertitore, da filtri analogici e dalla serie di un'induttanza ed una resistenza, che devono simulare un motore. Quindi, come fatto per il controllo del motore, è stato scelto di compensare il polo della funzione di trasferimento del ramo R-L con l'azione del PI, $\tau_{EMU} = \frac{L_{EMU}}{R_{EMU}}$.

Per la scelta del K_i è necessario considerare che il regolatore non deve essere influenzato dalla frequenza di commutazione del DUT, ovvero non deve cercare di inseguire la tensione modulata. Come indicato precedentemente la frequenza di commutazione del DUT è 5 kHz. Allo stesso tempo deve essere in grado di rispondere alle variazioni di corrente del motore, la cui frequenza è circa 200 Hz. La frequenza di attraversamento, che è quella per cui il modulo della funzione è unitario (ovvero 0 dB), deve essere compresa fra questi due riferimenti. Considerando che la funzione è monotona decrescente, tutto ciò che è a frequenza maggiore di quella di incrocio sarà attenuato. In questo modo è possibile escludere dal controllo tutto quello che varia ad una frequenza maggiore. Tenuti in considerazione tutti questi aspetti, si è scelto un $K_{i,EMU} = 190$, con una frequenza di attraversamento di 350 Hz e margine di fase 75.5°.

Il riferimento di tensione ottenuto è utilizzato per manovrare l'emulatore. Siccome le correnti sono misurate come uscenti dal DUT, è necessario invertirle per controllare l'EMU nella maniera corretta. La logica di modulazione prevede per convenzione che la corrente è positiva quando è uscente dal convertitore.

Inizialmente sono state introdotte anche le compensazioni della FEM e delle tensioni, ma per la natura distorta dei segnali misurati, introducevano ulteriore disturbo sulle tensioni di riferimento e quindi non sono state utilizzate.

L'accoppiatore, composto da una resistenza ed un'induttanza per ogni fase, collega i livelli di potenza dei due convertitori. I parametri utilizzati nella simulazione sono stati $R_{EMU} = 0.2 R$ e $L_{EMU} = 1.5 L_S$.

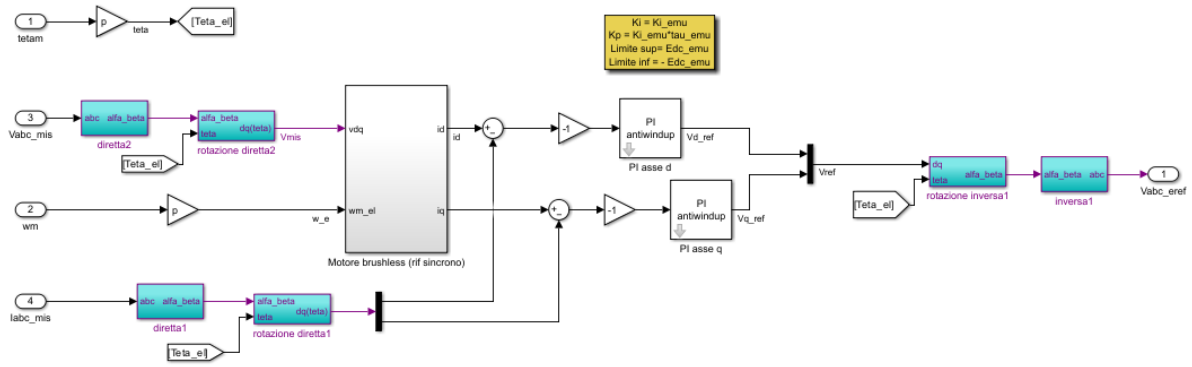


Figura 20: Controllo dell'emulatore

2.3.3 Risultati della simulazione

Le simulazioni si sono svolte considerando un riferimento di velocità del motore del 70% rispetto alla velocità nominale. Tale riferimento è un gradino, dato in ingresso contemporaneamente ai due schemi, ritardato di 0.01 s per permettere una migliore visualizzazione dei risultati. La finestra di tempo su cui sono state effettuate le prove è 1.5 s, in modo da permettere al motore di arrivare a regime.

Nelle figure sottostanti sono mostrati la terna di tensioni applicate al motore, la coppia generata, la terna di correnti circolanti e la velocità meccanica nelle simulazioni. Partendo dalle tensioni, si può notare che l'ampiezza delle tensioni applicate non supera mai la metà del bus DC per entrambe i convertitori; quindi, la scelta del bus DC è ritenuta corretta. Non appaiono oscillazioni di velocità, quindi il controllo risulta funzionare correttamente.

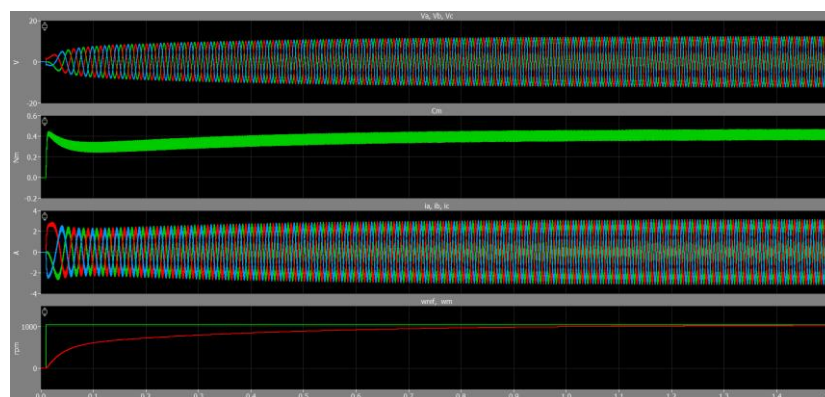


Figura 21: Risultati della simulazione del motore reale. In ordine sono mostrate tensioni, coppia, correnti e velocità

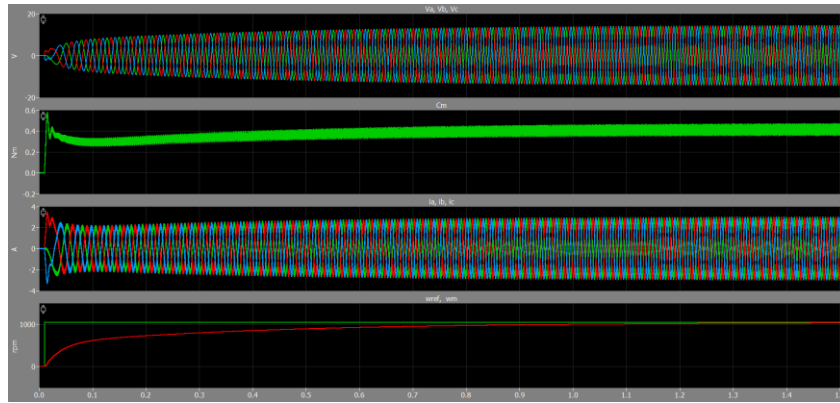


Figura 22: Risultati della simulazione del motore emulato. In ordine sono mostrate tensioni, coppia, correnti e velocità

È di particolare interesse il confronto fra le correnti circolanti nel motore reale e quelle circolanti nell'emulatore. Nella *figura 23* è graficata, in rosso, la corrente circolante nell'emulatore che risulta approssimare, con un ripple contenuto, l'andamento della corrente reale. Inoltre, nella *figura 24* è possibile osservare uno zoom su queste correnti in cui, sulla traccia rossa, si nota l'effetto della commutazione del DUT e dell'EMU. Il ripple di corrente più ampio è dato dall'inverter sotto test che commuta ad una frequenza minore, mentre il ripple di ampiezza minore è dovuto alla commutazione a frequenza maggiore dell'emulatore.

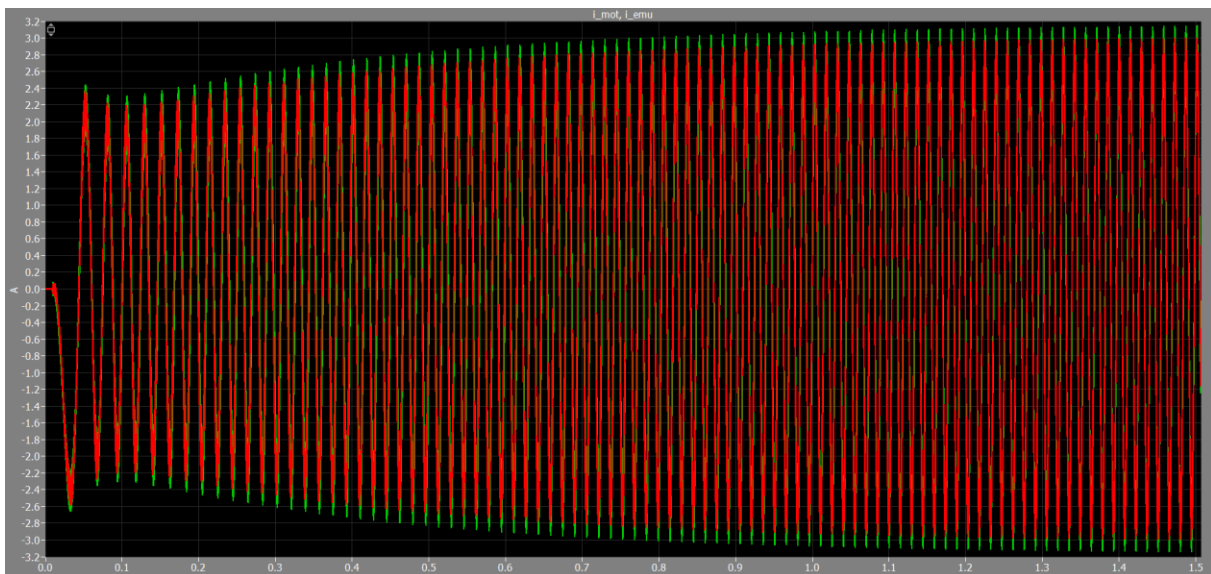


Figura 23 : Confronto fra le correnti circolanti nel modello del motore reale (in verde) e di quello emulato (in rosso)

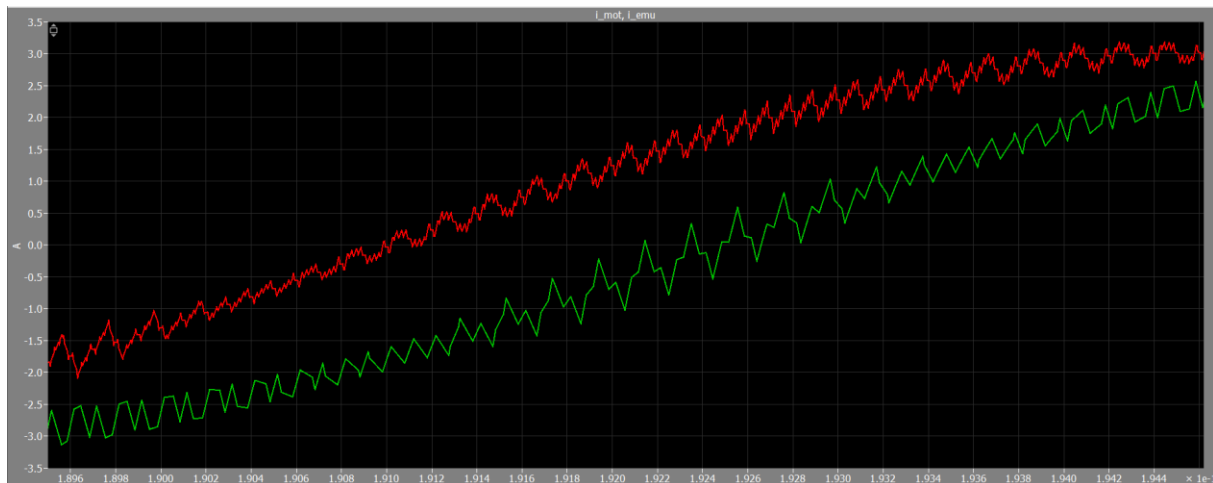


Figura 24: Zoom sulle correnti presentate nella figura precedente

2.4 Modello tempo discreto del motore sincrono

Per passare da una simulazione tempo continuo ad una tempo discreto è necessaria una conversione. Nel modo dei segnali tempo continuo, tutte le variabili sono definite per ogni istante temporale senza discontinuità. In un sistema tempo discreto i segnali elaborati hanno valori definiti solo in specifici istanti di tempo, separati da intervalli costanti. Questo significa che il sistema opera su segnali che cambiano solo a intervalli di tempo discreti, piuttosto che variare continuamente. I sistemi tempo discreto sono ampiamente utilizzati in ambiti come l'elaborazione digitale dei segnali, dove i segnali continui vengono campionati a intervalli regolari per la loro elaborazione digitale. L'intervallo con cui sono campionati i segnali definisce il periodo di discretizzazione.

I principali metodi di discretizzazione sono tre: Forward Euler, Backward Euler e Tustin. I primi due approssimano il segnale nell'intervallo di tempo della discretizzazione come differenze finite in avanti o all'indietro, mentre il metodo di Tustin lo approssima con un metodo trapezoidale. Quest'ultima, essendo più accurata e non avendo problemi di instabilità, è stata scelta per la discretizzazione del modello

2.4.1 Discretizzazione di Tustin

La discretizzazione di Tustin, nota anche come metodo di trasformazione bilineare, è un procedimento utile alla conversione di sistemi dinamici continui in sistemi dinamici discreti. Questo metodo è particolarmente utile nell'ambito del controllo digitale, dove si desidera implementare algoritmi di controllo su hardware digitale.

Il concetto di base della discretizzazione di Tustin si basa sulla sostituzione della derivata temporale presente nelle equazioni differenziali (che descrivono i sistemi continui) con un'approssimazione che risulta valida in un contesto discreto. Questa approssimazione si ottiene mediante l'uso di differenze finite, che permettono di esprimere la derivata di una funzione in termini delle sue differenze successive a intervalli di tempo regolari.

La trasformazione bilineare mappa il piano s della trasformata di Laplace, utilizzata per analizzare i sistemi continui, nel piano z della trasformata Z , utilizzata per i sistemi discreti.

Ogni relazione ingresso uscita, nella forma a poli e zeri nel dominio del tempo, può essere scritta in funzione dell'integratore di Laplace. Quindi a partire da esso è possibile ricavare una relazione fra il sistema tempo continuo e quello tempo discreto. Nel dominio del tempo l'integratore della trasformata di Laplace è definito come

$$G(s) = \frac{Y(s)}{U(s)} = \frac{1}{s} \Rightarrow y(t) = \int_0^t u(\tau) d\tau \quad (19)$$

Volendo discretizzare la funzione, il sistema sarà campionato con tempo T diventando

$$y(kT) = \int_0^{kT} u(\tau) d\tau = y[k] \quad (20)$$

con k appartenente ai numeri naturali.

L'istante successivo a questo sarà

$$\begin{aligned}
 y[k+1] &= y((k+1)T) = \int_0^{(k+1)T} u(\tau) d\tau = \int_0^{kT} u(\tau) d\tau + \int_{kT}^{(k+1)T} u(\tau) d\tau = \\
 &= y[k] + \int_{kT}^{(k+1)T} u(\tau) d\tau
 \end{aligned}$$

(21)

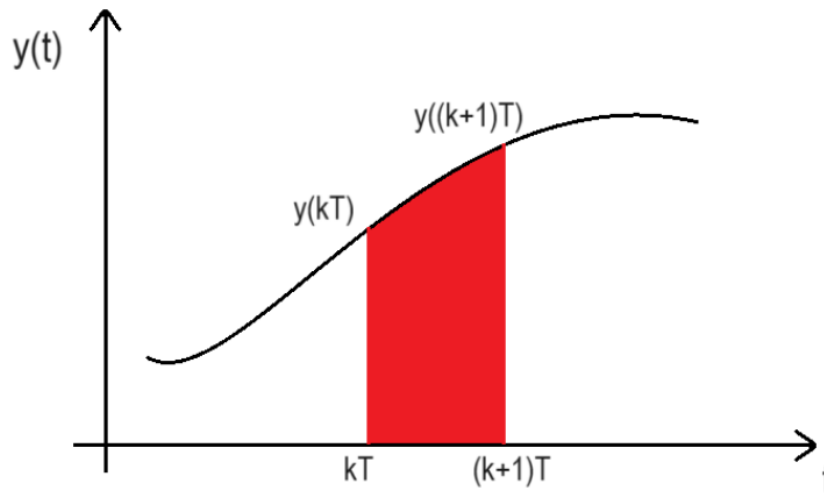


Figura 25: Area sottesa dalla funzione $y(t)$ nel periodo di campionamento

Il metodo di Tustin consiste nell'approssimare l'integrale con la media aritmetica dei campioni adiacenti alla finestra di campionamento.

$$y[k+1] = y[k] + \frac{u[k] + u[k+1]}{2} T$$

(22)

Applicando la trasformata Z all'equazione (22) si ottiene

$$Y(z) \cdot z = Y(z) + \frac{U(z) + U(z) \cdot z}{2} T$$

(23)

La cui funzione di trasferimento è

$$G(z) = \frac{Y(z)}{U(z)} = \frac{z+1}{z-1} \frac{T}{2}$$

(24)

$$\frac{1}{s} = \frac{z+1}{z-1} \frac{T}{2} \Rightarrow s = \frac{2}{T} \frac{z-1}{z+1}$$

2.4.2 Modello tempo discreto

Nella modellazione tempo discreta spesso conviene scrivere le funzioni di trasferimento direttamente in codice MatLab, come ad esempio per i regolatori PI. Inoltre, per evitare loop algebrici non processabili, si introducono nello schema dei ritardi di un periodo di discretizzazione, a fare questo sono i blocchi z^{-1} .



I blocchi gialli in *figura 26* sono quelli relativi al controllo dei due inverter. I calcoli all'interno di questi blocchi sono eseguiti ogni volta che si arriva il segnale di trigger. In particolare, tale segnale, è un'onda quadra con periodo uguale all'inverso delle frequenze di commutazione dei due convertitori e duty cycle del 5%. Su ogni fronte di salita del segnale vengono svolti i calcoli all'interno dei blocchi.

I regolatori PI sono stati discretizzati con una funzione in codice Matlab. L'equazione implementata è

$$G_{PI}(s) = K_p + \frac{1}{s} K_i \Rightarrow G_{PI}(z) = K_p + \frac{z + 1}{z - 1} \frac{T}{2} K_i \quad (25)$$

All'interno di questa funzione è presente anche un meccanismo di anti wind-up. Quando si utilizza un regolatore con azione integrale, è possibile che l'uscita del controllore raggiunga i limiti imposti dal dispositivo fisico. In presenza di saturazione,

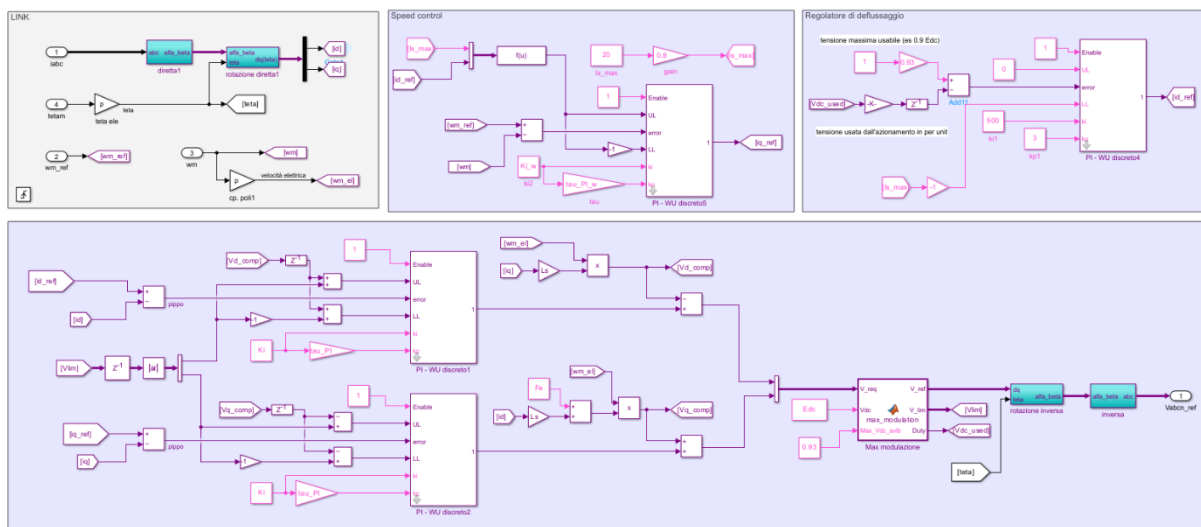


Figura 27: Schema di controllo del DUT, discreto

l'uscita dell'attuatore non cresce anche se l'errore di regolazione non è nullo. Conseguentemente, il termine integrale continua a crescere, ma tale incremento non produce alcun effetto sulla variabile di comando dell'impianto. Questa situazione, oltre a non far funzionare correttamente il regolatore, lo rende inattivo anche quando l'errore diminuisce o si inverte di segno; infatti, il sistema di regolazione può riattivarsi solo quando il segnale in uscita dal PI rientra nella zona di linearità della caratteristica dell'attuatore (scarica del termine integrale). Questo fenomeno si chiama comunemente carica integrale o integral wind-up.

Il controllo del DUT utilizza le stesse equazioni di controllo presentate nel modello tempo continuo con l'aggiunta della logica per il deflussaggio, ma i PI sono discreti. È presente anche la funzione “Max modulazione” che, nel caso in cui la tensione richiesta sia superiore a quella generabile, impone la massima come riferimento.

Il deflussaggio permette al motore di operare a velocità superiori a quella di base, ovvero introduce una corrente di asse d utile a creare un flusso che si oppone a quello dei magneti. Come visto precedentemente la coppia del motore è direttamente proporzionale alla corrente di asse q e quindi nel normale campo di utilizzo non è necessaria una corrente i_d . Nel momento in cui la velocità a cui si vuole far funzionare il motore è maggiore di quella base, la tensione necessaria, affinché circoli la corrente i_q che permette la generazione della coppia occorrente, potrebbe essere maggiore di quella generabile dal convertitore. Il riferimento della corrente i_d è individuato da un PI che regola in base all'errore sulla tensione presente sul bus DC del DUT. Se la $V_{DC,DUT}$ è minore della massima, ciò vuol dire che il carico del convertitore sta cercando di assorbire una corrente maggiore di quella massima.

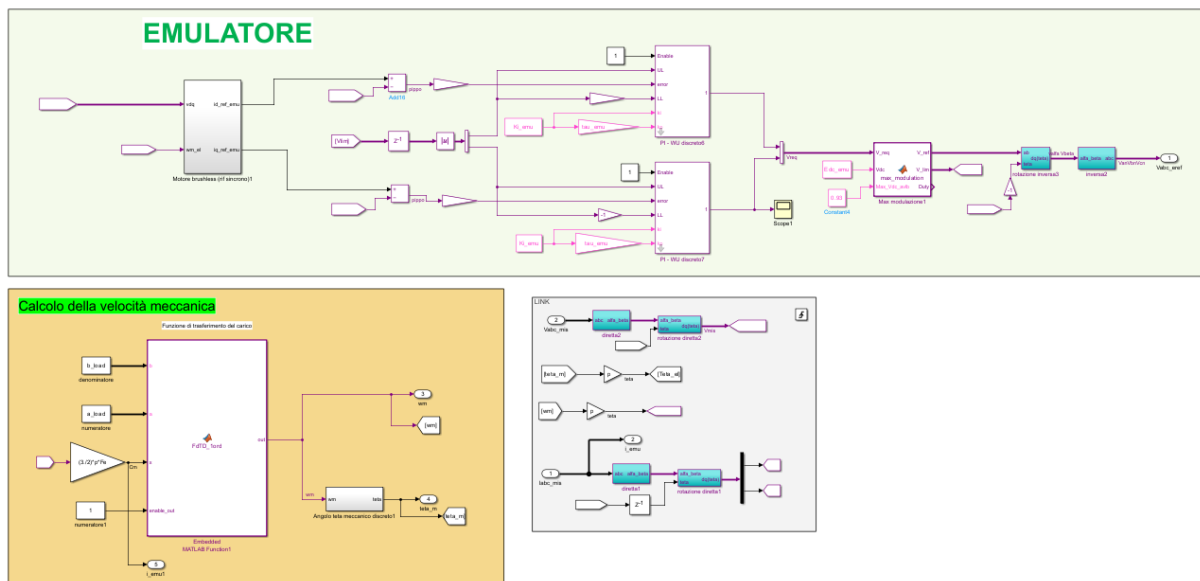


Figura 28: Schema di controllo dell'EMU e carico meccanico, discreto

Anche per lo schema dell'EMU sono stati discretizzati i PI e l'introdotta il blocco “Max modulazione”.

La funzione di trasferimento del motore (equazione (18)) è stata discretizzata come segue

$$G_{mot}(z) = \frac{z + 1}{\left(\frac{2L_s}{T} + R\right)z + \left(R - \frac{2L_s}{T}\right)}$$

A partire dalle *equazioni* (12) e introducendo la funzione appena ricavata, sono stati ottenuti i riferimenti di corrente di asse d e q implementati nel modello matematico.

All'interno di questo sistema sono stati fatti anche i calcoli relativi al carico meccanico, mostrati nell'area arancio della *figura 28*. L'*equazione* (13) che descrive il carico è stata discretizzata ottenendo

$$G_{load}(z) = \frac{z + 1}{\left(\frac{2J}{T} + b\right)z + \left(b - \frac{2J}{T}\right)}$$

Tale funzione restituisce la velocità meccanica che è successivamente integrata per ricavare l'angolo θ .

Questo modello è stato simulato con i parametri delle induttanze e resistenze che verranno utilizzate nella realizzazione pratica. L'induttanza scelta per l'accoppiamento ha un valore di $L_{emu} = 1800 \mu H$ e la resistenza serie è quella dell'avvolgimento dell'induttanza stessa, $R_{emu} = 42 m\Omega$. Sono stati anche ricalcolati i parametri del regolatore PI del controllo dell'emulatore ed è stato usato un $\tau_{emu} = \frac{L_{emu}}{R_{emu}}$ e un $K_i = 77$.

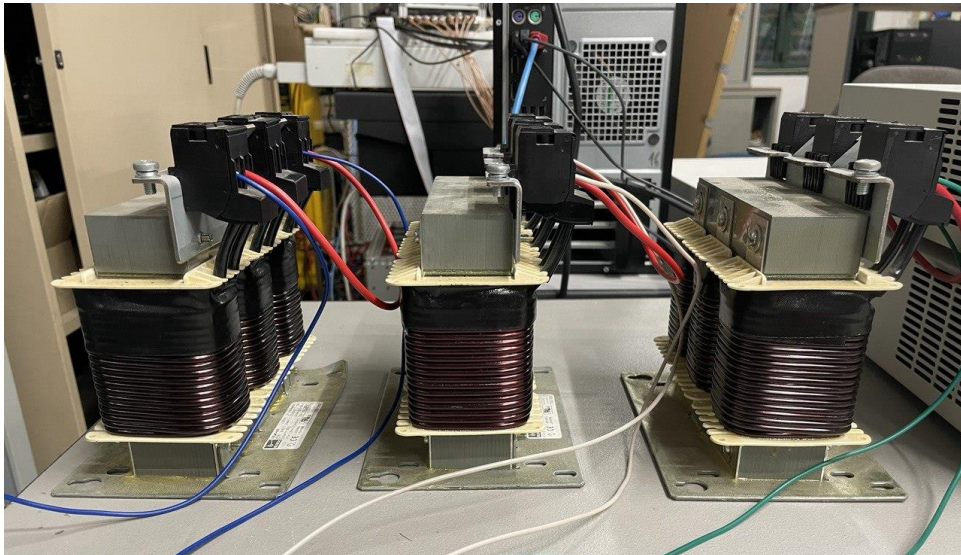


Figura 29: Induttori utilizzati per la realizzazione del dispositivo

2.4.3 Risultati della simulazione

La simulazione, anche in questo caso, è fatta su una finestra temporale di 1.5 s per permettere al motore di arrivare a regime.

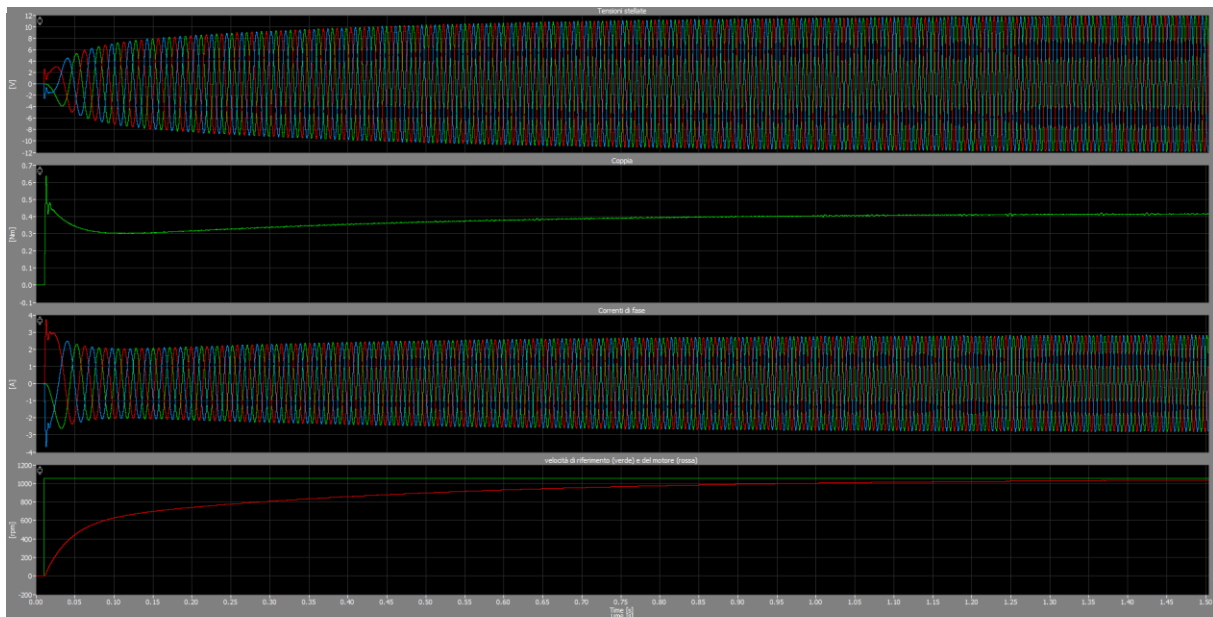


Figura 32: Risultati della simulazione del motore emulato, discreto

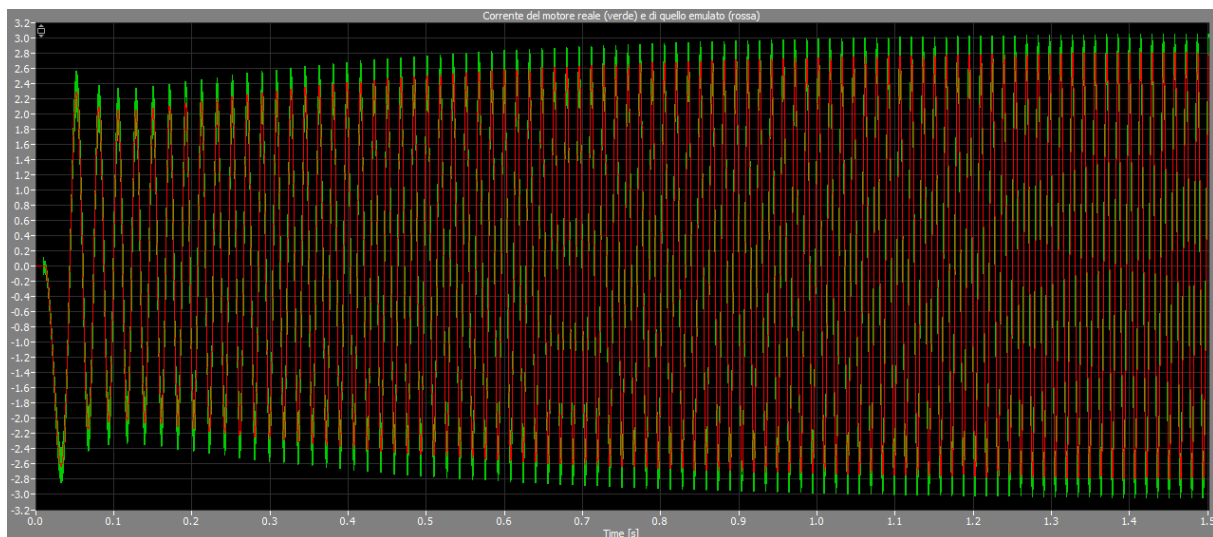


Figura 31: Confronto delle correnti del modello reale e quello emulato, discreto

Le forme d'onda del modello discreto risultano avere meno ripple in quanto tutte le variabili sono campionate e ciò comporta un'azione filtrante che permette di diminuire i disturbi in ingresso al sistema.

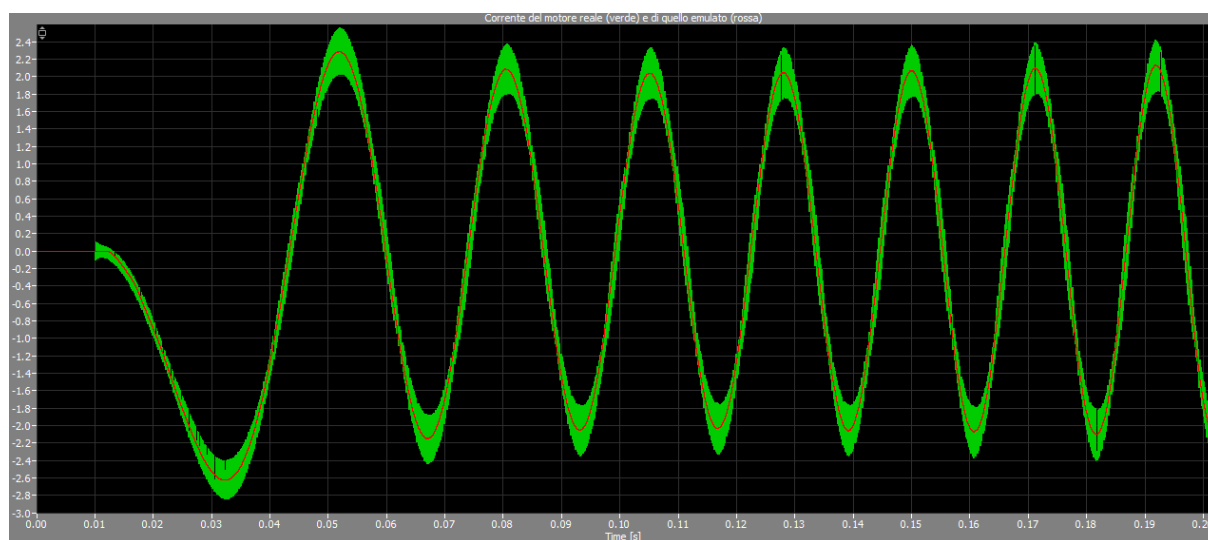


Figura 33: Zoom sulle correnti della figura precedente

CAPITOLO 3

DISPOSITIVO DI CONDIZIONAMENTO DEI SEGNALI

Per evitare che i poli negativi dei due convertitori entrino in contatto, con la conseguente circolazione di correnti indesiderate, è necessario realizzare un dispositivo che isoli i segnali. La configurazione del sistema prevede un unico DSP per entrambi gli inverter e, dato che la loro struttura interna impone il collegamento del ground di potenza a quello di segnale, solo uno dei due può essere montato sulla scheda di sviluppo Delfino F28379D. Si è scelto di mantenere il DUT sulla board e isolare i segnali di comunicazione dell'EMU.

Inoltre, per poter filtrare analogicamente le tensioni e le correnti, sono stati montati su questo dispositivo dei sensori ad effetto Hall per le correnti e dei sensori di tensione.

Lo stadio di alimentazione è composto da un buck che riceve un'alimentazione di 12 V e la abbassa a 5 V. Il riferimento creato è utilizzato per alimentare i sensori di tensione e un regolatore lineare che produce il secondo riferimento di 3.3 V, necessario a tutti gli altri componenti integrati.

Gli isolatori sono unidirezionali, i segnali PWM provenienti dal DSP e il segnale di enable sono suddivisi su due isolatori a quattro canali. Mentre il segnale di fault prodotto dall'inverter è isolato con un componente ad un solo canale. Questi sono alimentati da una tensione di 3.3 V su entrambi i fronti di isolamento. Dal lato controllore l'alimentazione proviene dalla scheda di isolamento, mentre per il lato EMU essa è fornita dall'inverter stesso.

La misura delle correnti è fatta su ogni fase nel tratto di collegamento fra il DUT e le induttanze di accoppiamento. I sensori scelti per le correnti sono gli Allegro ACS722.

Il segnale in uscita è filtrato analogicamente da un filtro attivo Sallen-Key passa basso. La configurazione circuitale è la seguente

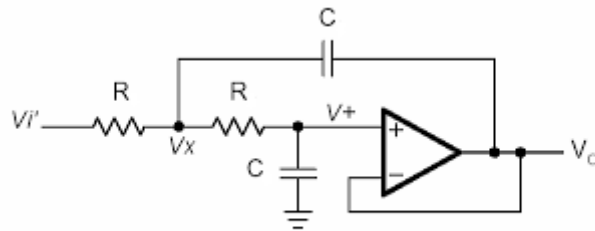


Figura 34: Configurazione filtro Sallen-Key passa basso

La sua funzione di trasferimento presenta due poli e un guadagno unitario. Per il dimensionamento si è considerato l'approccio a componenti uguali, ovvero le due resistenze e i due condensatori hanno il medesimo valore. In questo modo la frequenza di taglio del filtro è espressa dalla relazione $f_0 = \frac{1}{2\pi RC}$. Considerata la frequenza di taglio dimensionata nelle simulazioni, i componenti reali che riproducono meglio quella frequenza sono $R = 10\text{ k}\Omega$ e $C = 10\text{ nF}$, da cui si ottiene

$$f_{0,curr} = 1.592\text{ kHz}$$

Il circuito descritto, prima dell'implementazione, è stato simulato in ambiente PLECS e lo schema è mostrato in *figura 35*.

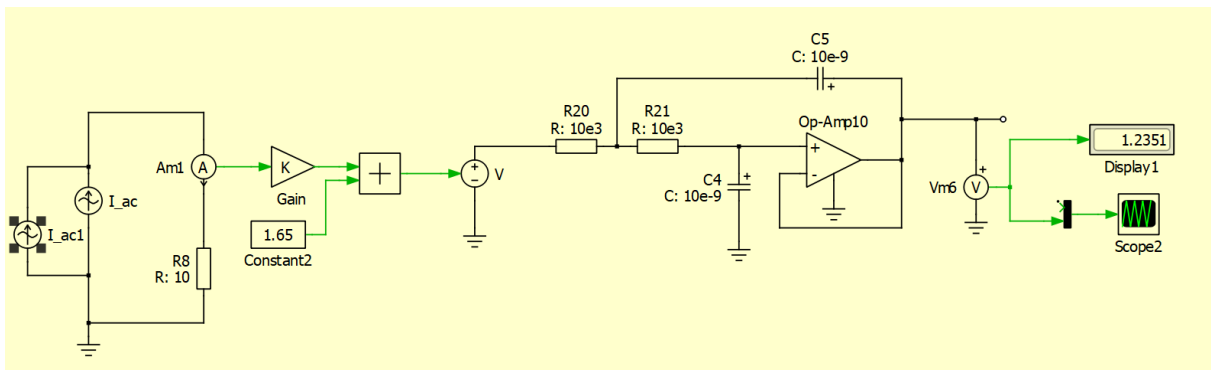


Figura 35: Circuito di misura delle correnti

Le tensioni stellite sono prelevate sul DUT e scalate tramite un partitore resistivo. I sensori scelti sono amplificatori isolati Broadcom ACPL-C79A, la cui tensione in ingresso ha un range di $\pm 300\text{ mV}$. Considerata una tensione massima sul bus DC degli inverter di 40 V , il partitore resistivo è ottenuto dalla seguente relazione

$$\frac{V_{fase}}{V_{sense}} = \frac{R_1 + R_2}{R_2}$$

Le resistenze scelte sono $R_1 = 100 \text{ k}\Omega$ e $R_2 = 680 \Omega$. Sul partitore è presente anche una capacità di 1 nF per aiutare a filtrare le tensioni.

L'ADC ha un range di acquisizione di $0\text{-}3.3 \text{ V}$ e per poter osservare tensioni positive e negative, al segnale in uscita dagli amplificatori, è stato imposto un offset di 1.5 V proveniente da un componente che crea una tensione stabilizzata.

Queste misure sono poi filtrate da uno stadio uguale a quello descritto per le correnti, ma con differente frequenza di taglio. Le resistenze scelte in questo caso hanno valore $R = 22 \text{ k}\Omega$ e le capacità $C = 10 \text{ nF}$.

Anche per le tensioni si è prima verificato il corretto funzionamento su PLECS.

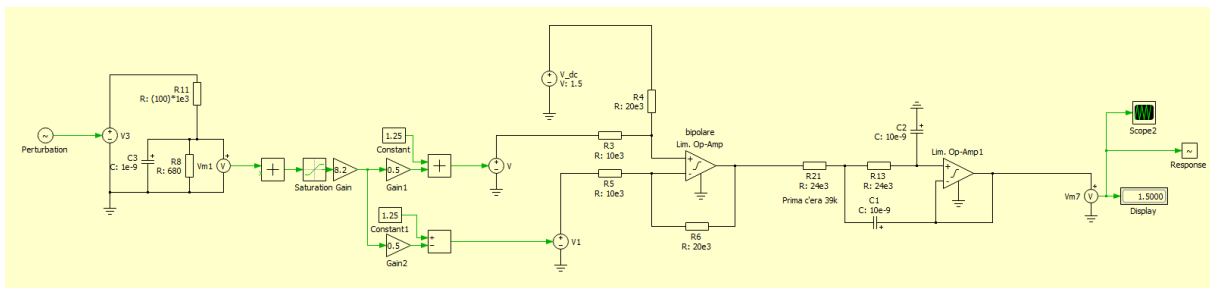


Figura 36: Circuito di misura delle tensioni stellate

Lo stesso circuito è stato realizzato per la misura del bus DC dell'emulatore, ad eccezione del fatto che non è necessario l'offset introdotto sulla misura, la tensione di alimentazione è sempre positiva.

Per favorire una migliore analisi del sistema, sulla scheda sviluppata, sono stati introdotti nove connettori che permettono di visualizzare i segnali misurati con l'utilizzo di un oscilloscopio. Questi comprendono: le tre misure di tensione, le tre misure di corrente, la tensione del bus DC dell'emulatore e l'uscita dei due canali del DAC presenti sul controller. Tutti i segnali sono bufferizzati con amplificatori operazionali in configurazione di inseguitore di tensione.

A seguito della realizzazione, è stata fatta un'analisi al piccolo segnale sui sensori di tensione per verificarne il corretto funzionamento. I risultati ottenuti confermano la frequenza di taglio del filtro, come mostrato in figura 37. Per le correnti non è possibile fare la stessa verifica a causa della diversa tecnologia utilizzata per la misura.

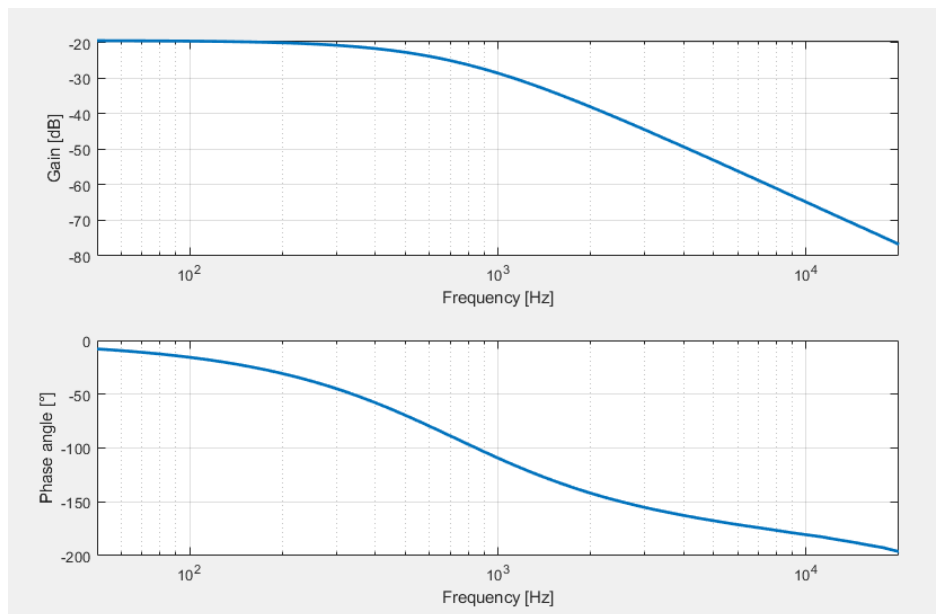


Figura 37: Risultati dell'analisi al piccolo segnale per lo stadio di misura delle tensioni

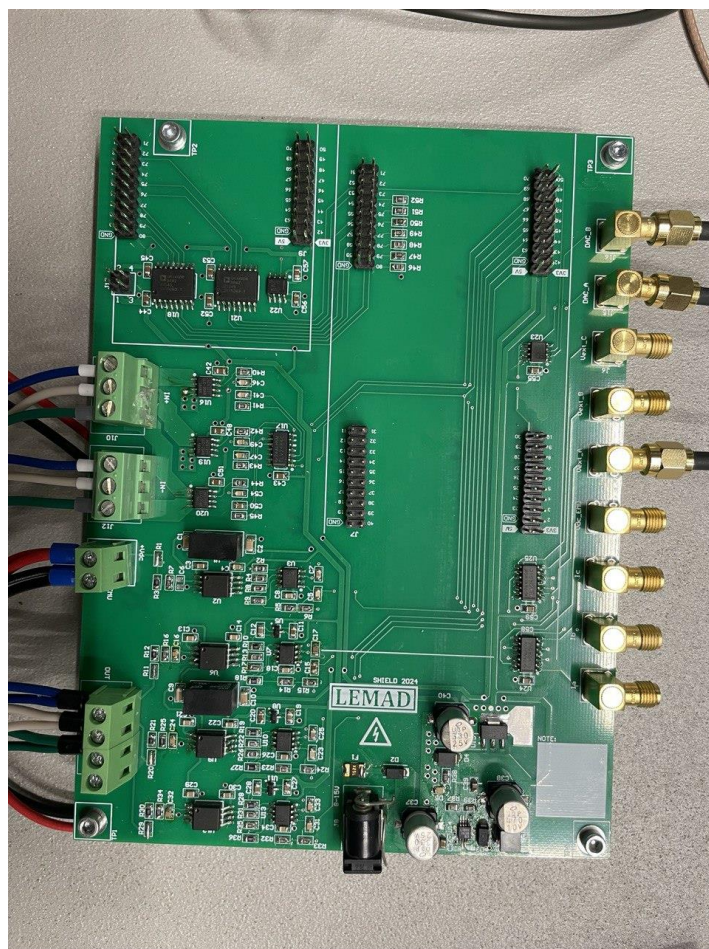


Figura 38: Scheda di condizionamento dei segnali realizzata

CAPITOLO 4

IMPLEMENTAZIONE DEL MODELLO

SU DSP

La scheda di sviluppo monta un chip con frequenza di clock di 200 MHz e capacità computazionale che è risultata sufficiente per lo scopo del progetto. Su di essa sono presenti comparatori con cui generare le PWM, quattro ADC a 12 bit e due DAC.

Sul DSP sono stati implementati i due blocchi di controllo analizzati nella modellazione discreta.

L'ambiente di sviluppo integrato, o IDE, utilizzato per caricare il codice sulla scheda di sviluppo è Simulink. Fra le tante sue funzioni, questo software è in grado di generare il codice sorgente a partire dallo schema a blocchi, caricarlo sulla scheda di sviluppo e fare da interfaccia per lo sviluppo o il debug.

La dashboard del sistema si presenta come in *figura 39*. Su di essa è presente l'interruttore di start che prepara il sistema al funzionamento e quello di enable che abilita i blocchi del controllo. Sui due display in alto sono visualizzati la modalità in cui sta funzionando il dispositivo e il tipo di errore, se presente. Il pulsante di reset permette di uscire dalla condizione di fault e, nel caso non siano più presenti errori, ritornare al funzionamento normale. Nella casella "SPEED REF" viene impostata la velocità di riferimento, che può essere cambiata anche mentre il dispositivo è in funzione. Vi sono poi una serie di display che mostrano le principali variabile del DUT, ma siccome sarebbe troppo complesso per il sistema aggiornare quei valori alla frequenza degli inverter, hanno un tempo di campionamento di 0.8 s. Nei menù a tendina è possibile settare quali parametri visualizzare sui canali del DAC.

Per quanto riguarda l'emulatore è possibile modificare il carico agendo su un parametro di scala dell'inerzia e sulla costante di coppia resistente.

I comandi di trigger vengono generati da interrupt. Per il blocco relativo all'emulatore è la fine della conversione analogico digitale a scaturire il segnale, ovvero quando le misure dell'ADC sono disponibili. Mentre per il DUT ciò che genera l'interrupt è l'inizio del periodo delle PWM. Inoltre, siccome l'EMU ha una frequenza molto più alta necessita di fare i calcoli in poco tempo. Per questo motivo le operazioni relative al DUT sono settate come interrompibili e terminabili non appena il processore è libero.



4.1 Controllo dell'emulatore

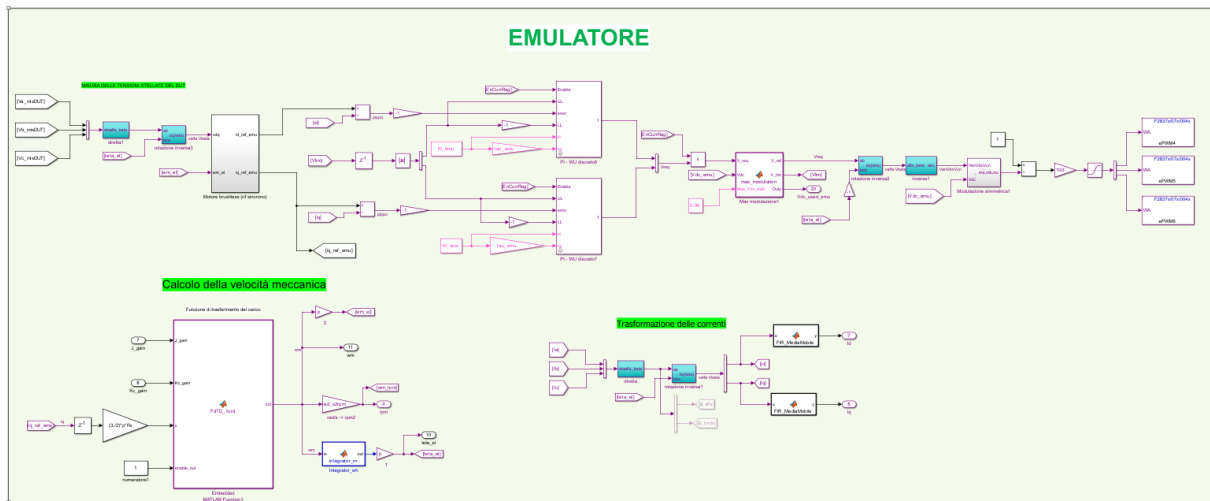


Figura 40: Schema di controllo dell'EMU implementato su DSP

Lo schema di base che controlla l'emulatore è lo stesso presentato precedentemente. Le tensioni in ingresso al modello matematico del motore sono prelevate dal blocco di controllo del DUT per un primo approccio all'implementazione fisica.

Prima che le correnti, misurate e trasformate, siano passate al blocco del DUT sono mediate da un filtro FIR a media mobile di ordine tre, la cui equazione è

$$y[k] = \frac{1}{N} \sum_{m=0}^{N-1} x[k-m]$$

Ciò è dovuto al fatto che in un periodo di commutazione del DUT ce ne sono cinque dell'EMU e quindi cinque campionamenti da parte del ADC. Se, nel momento in cui l'ADC sta elaborando la misura, il DUT accedesse al registro di memoria della conversione potrebbe prelevare un valore non corretto. Per evitare questo si passa al blocco più lento una media degli ultimi tre campioni.

I riferimenti di tensione sono dati in ingresso ai blocchi per la creazione dei segnali PWM. Il dispositivo permette di sincronizzare tutte le portanti rispetto alla prima (ePWM1) tramite un canale di sincronizzazione interno.

La strategia di modulazione utilizzata prevede un segnale alto quando la portante è maggiore della modulante. È necessario quindi usare il complemento a uno sugli indici di modulazioni calcolati per ottenere il risultato corretto.

La corrente i_{q,ref_emu} calcolata è usata per ottenere la coppia del motore, da cui è ricavata la velocità tramite la funzione di trasferimento del carico. La velocità è poi passata come riferimento al DUT.

4.1.1 Conversioni Analogico Digitali

All'interno del blocco triggerato dell'emulatore è presente la conversione analogico digitale. Disponendo di quattro ADC e tredici canali per ognuno (più due canali comuni), sono state campionate le misure delle stellate, delle correnti e delle tensioni sui due bus DC. A seguito del campionamento vi è la fase di quantizzazione in cui l'ADC assegna a ogni campione un valore discreto selezionato da un insieme finito di possibili valori. Questo processo introduce un errore di quantizzazione, poiché il valore assegnato può non corrispondere esattamente al valore reale del campione. La risoluzione dell'ADC, espressa in bit, determina il numero di livelli discreti possibili e, di conseguenza, l'accuratezza della conversione. Quelli presenti sono a 12 bit, ovvero 4096 livelli che garantiscono una sensibilità di $\frac{3.3V}{4096} = 0.806 \text{ mV}$. L'ultimo passaggio è la conversione e la taratura di tutta la catena di misura. Bisogna tenere anche in considerazione l'offset introdotto su alcune misure e rimuoverlo dal valore convertito.

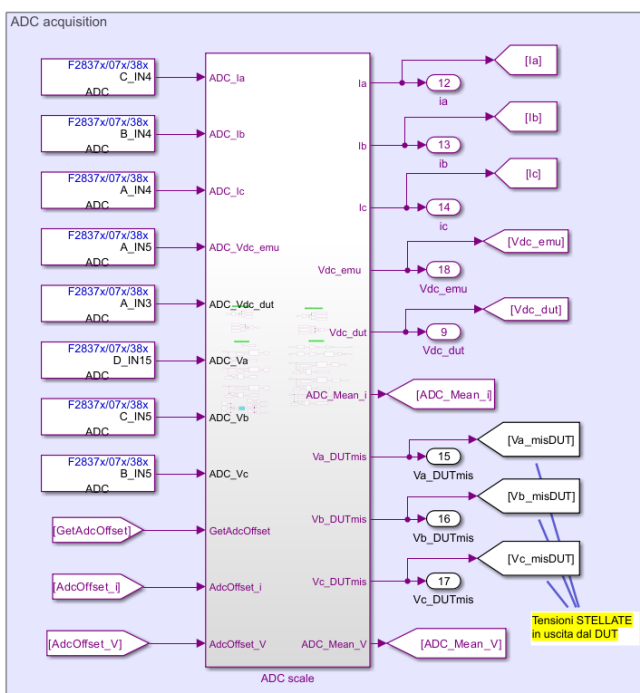


Figura 41: Blocco di acquisizione delle misure

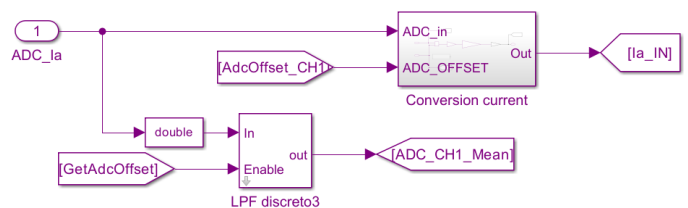


Figura 42: Conversione, offset e taratura delle misure

4.1.2 Macchina a stati

Una macchina a stati è un modello concettuale utilizzato per progettare sistemi di controllo che evolvono attraverso una serie di stati in risposta a degli input esterni. Questo approccio consente di descrivere il comportamento di un sistema in termini di transizioni tra stati specifici, ognuno dei quali rappresenta una particolare configurazione del sistema. In altre parole, è un metodo strutturato per gestire la logica complessa e le decisioni basate sugli eventi.

Il flow chart implementato è mostrato in *figura 43* sulla destra. Il punto di ingresso è il pallino blu, nel primo stato si rileva l'offset presente sui canali e si passa al secondo stato quando è passato un secondo dall'accensione. Questo è lo stato di attesa, tutti i riferimenti sono posti a zero, i controlli disattivati e gli offset imposti sulle uscite. Quando sulla dashboard si preme il bottone start, il controllo si mette in uno stato in cui è ancora tutto disabilitato ma può partire in qualsiasi momento. Se dalla dashboard si attiva il controllo, i regolatori vengono attivati e l'azionamento è messo in funzione. Se, in un qualsiasi momento, viene rilevato un fault le correnti sono poste a zero e il driver esterno spento. Il sistema, se non ci sono più le condizioni di fault, esce dalla modalità di protezione quando viene premuto il tasto “RESET FAULT” sulla dashboard.

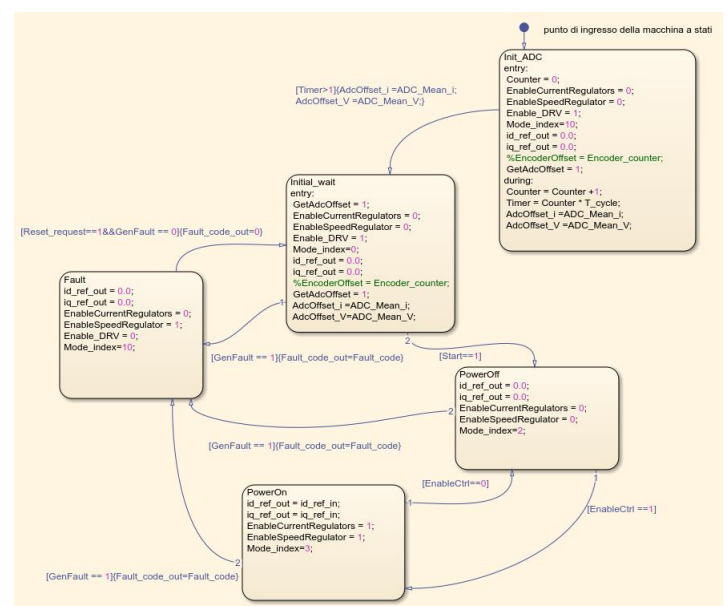
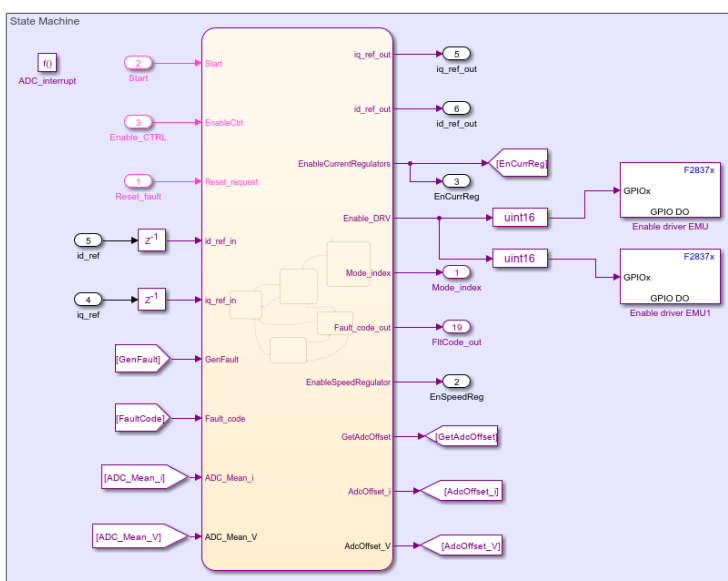


Figura 43: Macchina a stati

4.1.3 Protezioni

Un sistema di protezioni è necessario per evitare di danneggiare il sistema. Le condizioni di fault sono determinate da correnti circolanti nelle fasi, tensioni in ingresso agli inverter o la velocità calcolata che superano un valore preimpostato. Se il driver dell'inverter rileva un'anomalia pone in uscita sul pin di fault un livello basso, il controllo rileva ciò e porta il sistema in protezione.

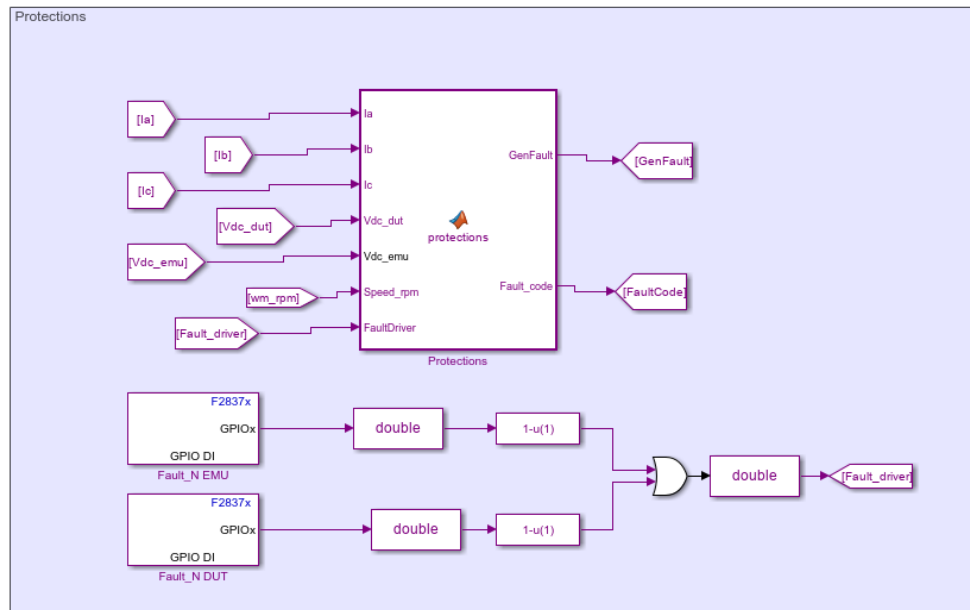


Figura 44: Blocco relativo alle protezioni

4.2 Controllo dell'inverter sotto test

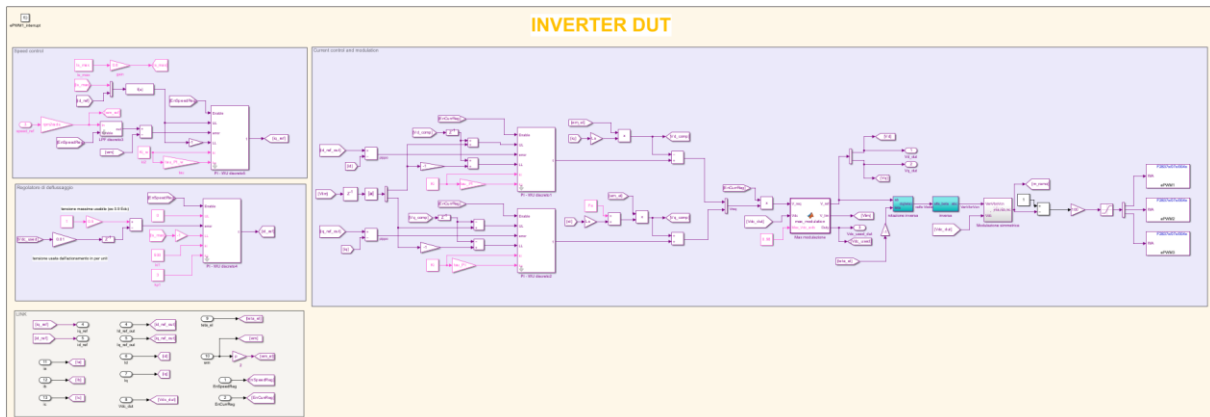


Figura 45: Schema di controllo del DUT

Come per il controllo dell'emulatore, non c'è nessun cambiamento nel modello rispetto a quello già visto, vengono solo generate le PWM a partire dai riferimenti creati.

4.2.1 Conversione Digitale Analogico

Il blocco della conversione non è altro che un demultiplexer con cui selezionare quale misura imporre sul pin relativo al DAC considerato. La selezione è fatta sulla dashboard e i segnali sono visibili attraverso due dei connettori della shield progettata nel capitolo 3.

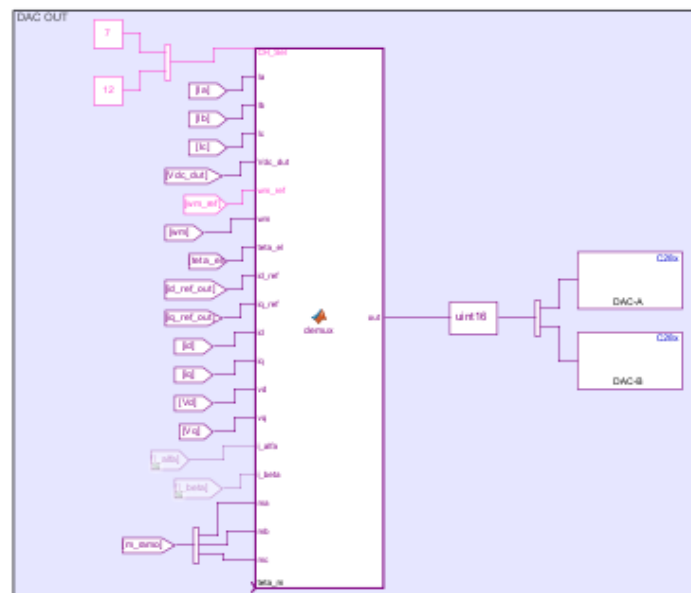


Figura 46: Blocco relativo al DAC

CAPITOLO 5

RISULTATI SPERIMENTALI

Il setup sperimentale realizzato è mostrato in *figura 42*, dove sulla scheda di condizionamento dei segnali sono presenti il DUT montato sulla scheda Delfino F28379D e l'EMU a sinistra. Due alimentatori da banco, producono la tensione in ingresso all'emulatore e all'inverter sotto test, rispettivamente di 35 V e 25 V.

In questo capitolo vengono commentate le forme d'onda ottenute dalle prove sperimentali sul dispositivo. Le prove sono state fatte ad una velocità impostata di 1000 rpm e disattivando il controllo di deflussaggio, ovvero ponendo i_d sempre a zero.

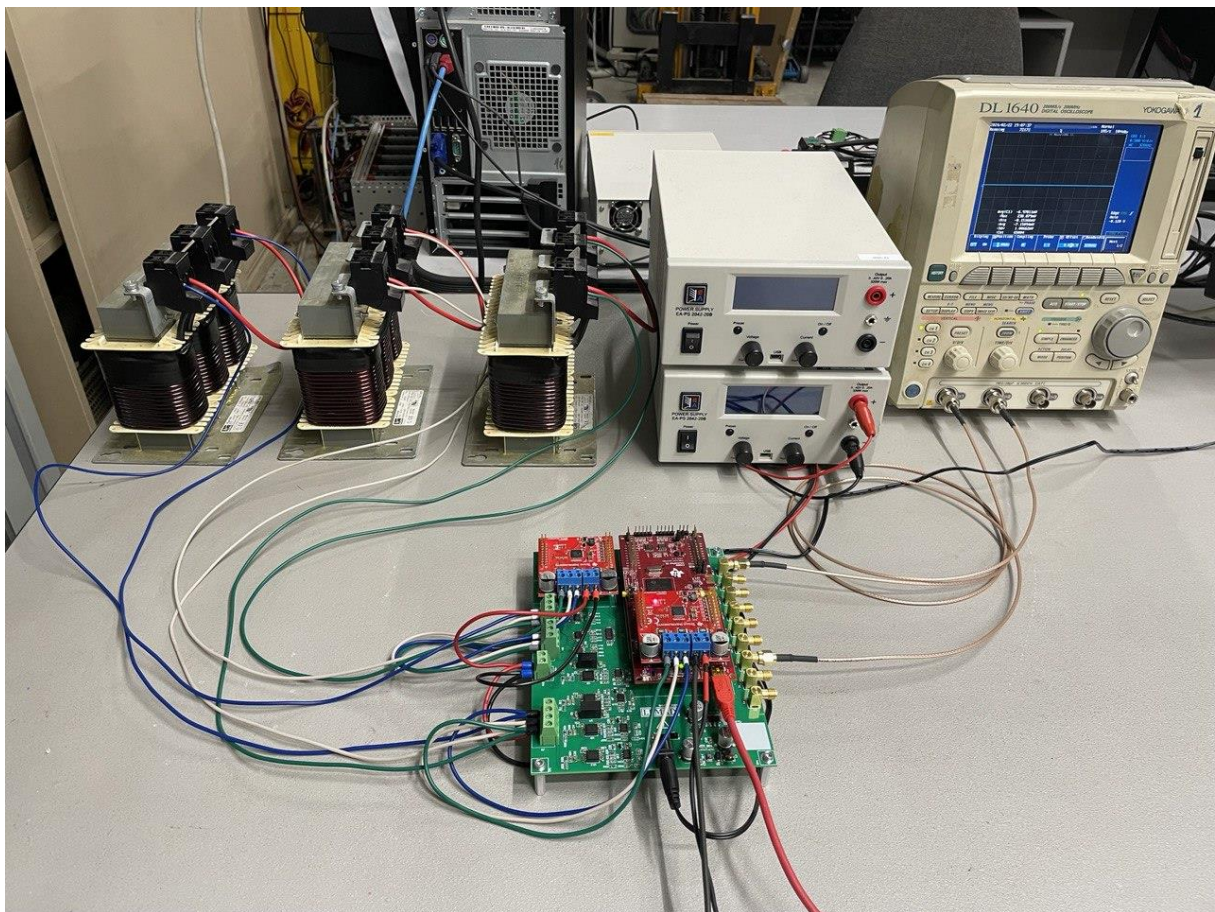


Figura 47: Setup dell'emulatore

La corrente misurata, con una sonda dell'oscilloscopio, della fase a è mostrata in giallo nella figura sottostante. Essa è confrontata con la forma d'onda in uscita dal DAC, queste due misure hanno lo stesso guadagno e sono sovrapponibili. Mentre nel riquadro sottostante è visualizzato il segnale misurato dai sensori sulla scheda.

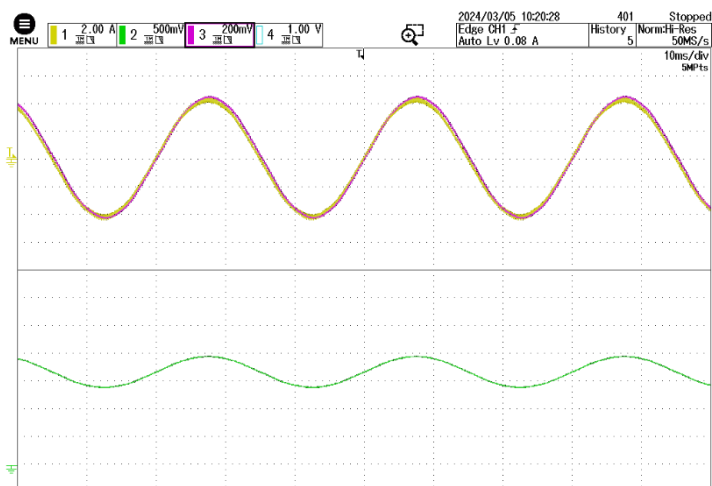


Figura 49: Corrente della fase a

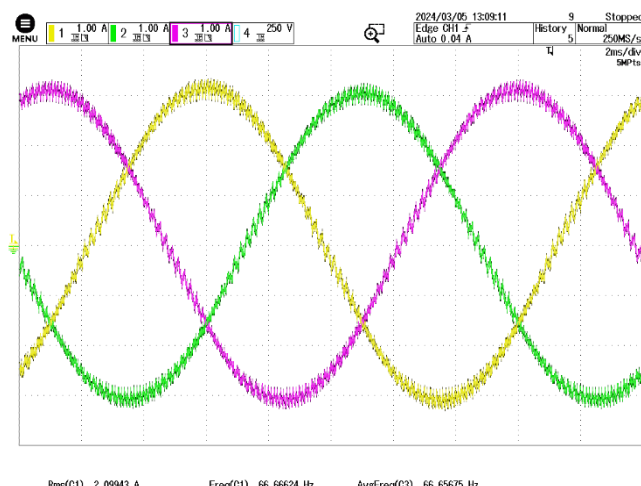


Figura 48: Correnti nelle tre fasi

Inoltre, facendo uno zoom sulle tracce della sonda e del DAC, si nota il ripple presente sulla corrente e l'effetto della quantizzazione per il secondo.

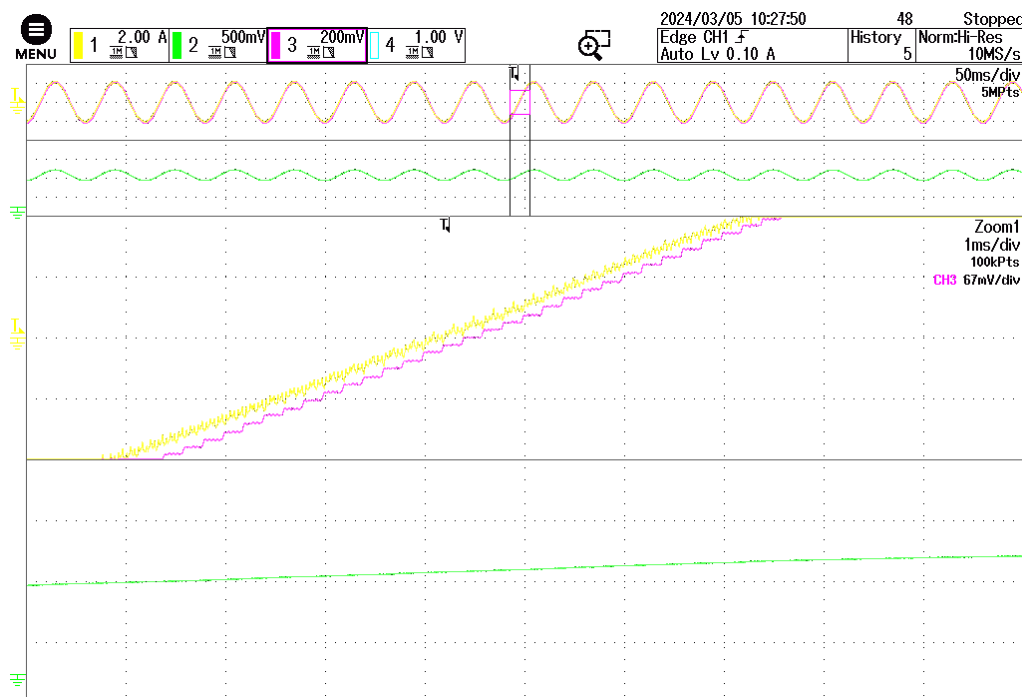


Figura 50: Zoom sull'effetto di quantizzazione

Riguardo al ripple di corrente, operando un'analisi in frequenza del segnale proveniente dalla sonda, si notano picchi delle componenti armoniche in corrispondenza della frequenza di commutazione del DUT (5 kHz) e delle sue multiple, fra cui quella dell'EMU (25 kHz).

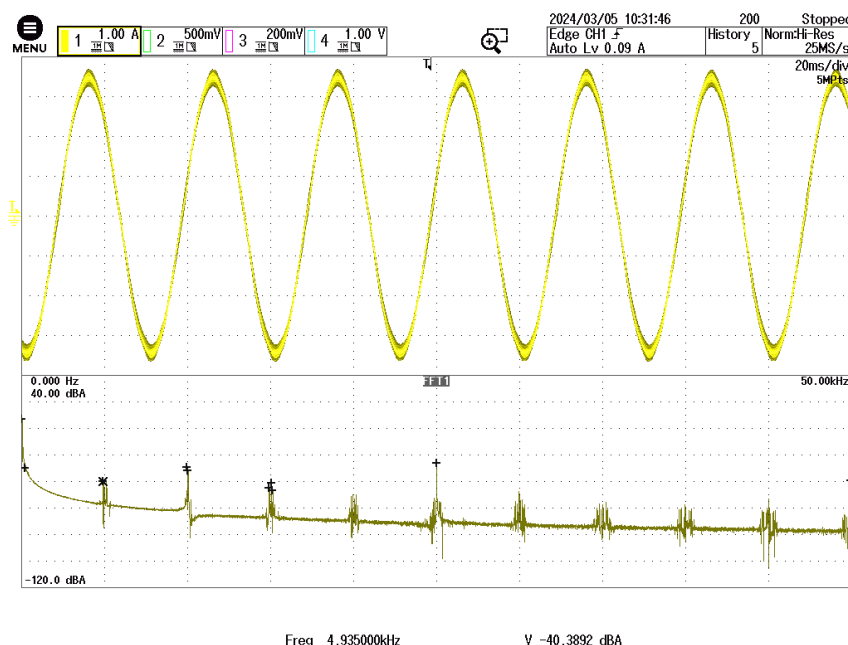


Figura 51: Analisi in frequenza sulla forma d'onda della corrente

Filtrando tale corrente con una frequenza di taglio di 8 kHz, rimane solo il ripple causato dal DUT, mostrato in figura 52. Confrontando questa immagine con quella dell'onda filtrata a 20 MHz, in modo tale da includere il ripple dell'EMU, si nota che l'emulatore produce un ripple di frequenza cinque volte maggiore.

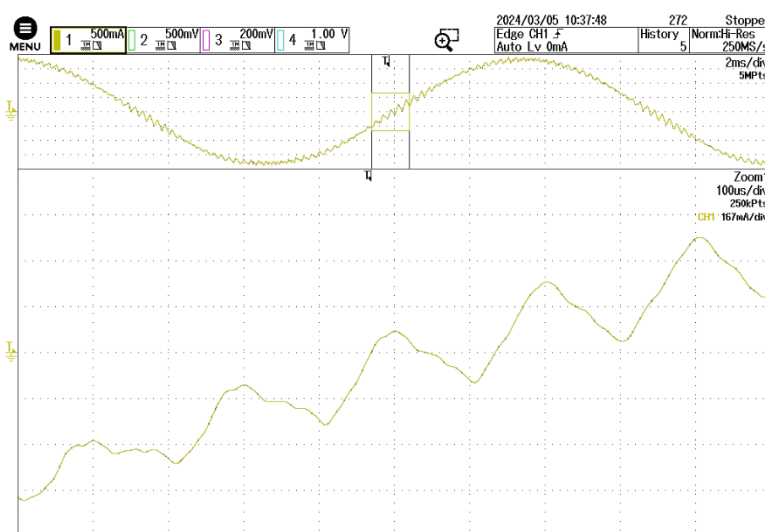


Figura 52: Ripple del DUT, corrente i_a filtrata a 8 kHz

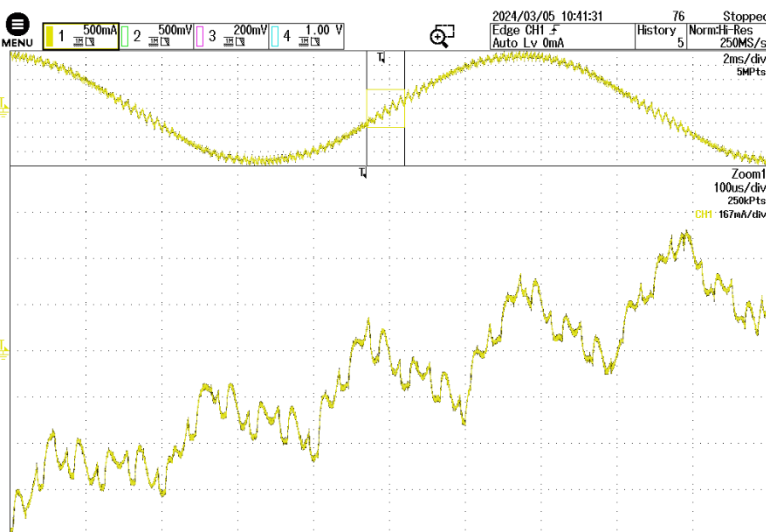


Figura 53: Ripple dell'EMU, corrente i_a filtrata a 20 MHz

Passando all'analisi delle tensioni, nella *figura 54* è visualizzata in viola la tensione stellata V_a , misurata con l'oscilloscopio sui terminali dell'emulatore. In relazione ad essa è posta la corrente della fase a , da cui si vede che il dispositivo funziona a $\cos \phi$ unitario. La traccia verde è il risultato del filtraggio analogico che comporta un inevitabile ritardo sul segnale.

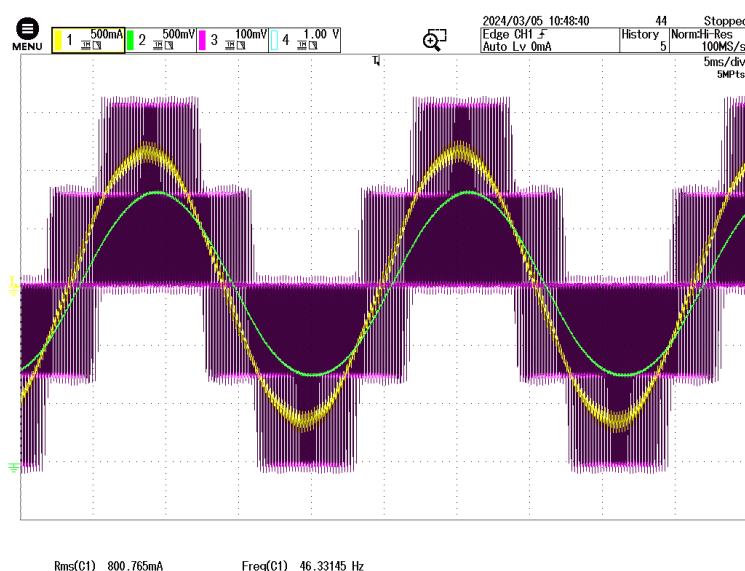


Figura 54: Tensione della fase a

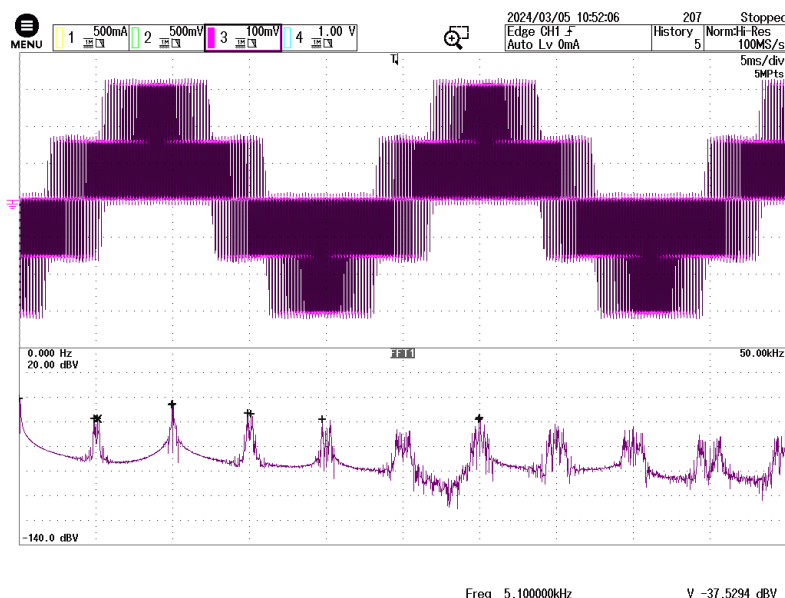


Figura 55: Analisi in frequenza della tensione

Come per la corrente, l'analisi in frequenza evidenzia che le armoniche più rilevanti sono quelle multiple della frequenza di commutazione.

Facendo uno zoom sulla tensione misurata dall'oscilloscopio, si può notare la forma d'onda imposta dalla modulazione simmetrica.



Figura 56: Effetto della modulazione simmetrica

Nella condizione di motore fermo le tensioni imposte dal controllo sono nulle in entrambi i convertitori, che corrisponde all'imposizione di un duty cycle del 50% sulle PWM. Da ciò si può apprezzare la frequenza di commutazione dei due dispositivi e la relazione fra i loro periodi. Inoltre, si vedono anche i diversi livelli di tensione di alimentazione degli inverter.

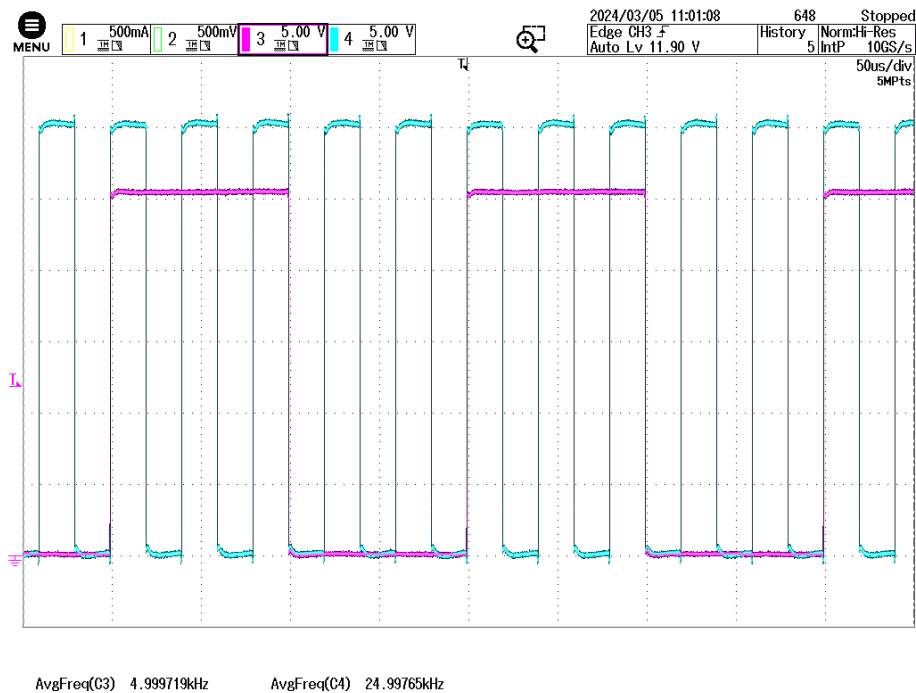


Figura 57: Confronto fra le PWM del DUT e dell'EMU

Le prove che seguono fanno riferimento ai transitori elettrici o elettromeccanici del motore emulato ed è quindi opportuno un confronto con il comportamento del motore reale su banco prova. Per rendere il test attendibile, il motore reale è controllato da un inverter uguale al DUT, con lo stesso schema di controllo implementato in questo elaborato. Le condizioni in cui vengono svolti i test sono identificate dalla velocità (1000rpm) e dalla corrente di asse q. È stata presa la i_q come riferimento siccome risulta essere direttamente proporzionale alla coppia del motore.

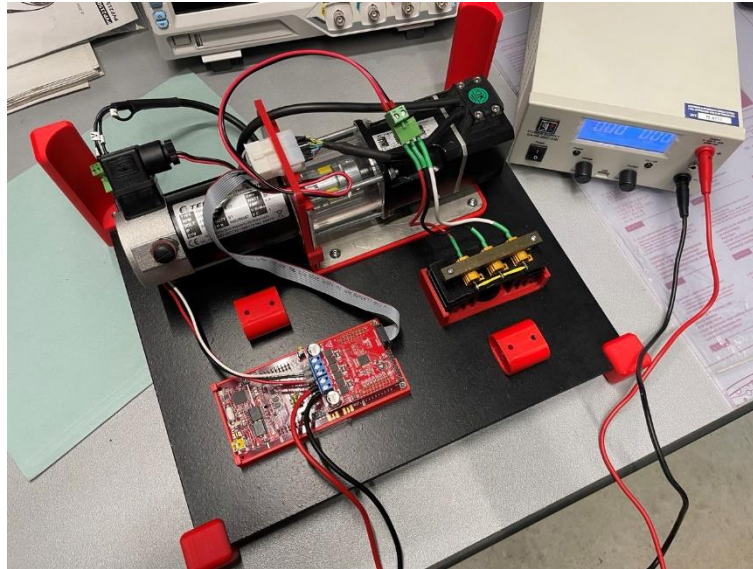


Figura 58: Banco prova del motore reale

Il primo test è svolto applicando un riferimento di velocità a gradino, poi filtrato (con costante di tempo 100 ms) per non portare il riferimento di corrente in saturazione. Dalla prova sul motore reale si registra un assorbimento di corrente a regime di circa $i_q = 3.1$ A. Quindi con l'emulatore si è cercato di ottenere la stessa corrente modificando la costante di coppia resistente dalla dashboard. Il parametro che è stato considerato è $K_c = 1.25$. Nelle immagini sono mostrate le correnti i_a in giallo, corrente i_q in verde e la velocità in viola. Dal confronto visivo della i_q si nota che per il motore reale è presente un picco di corrente iniziale, dovuto probabilmente a impuntamenti del motore che non sono modellati nell'emulatore e quindi non presenti. L'accelerazione è circa uguale considerando che la velocità impiega 250 ms per arrivare a regime in entrambi i casi.

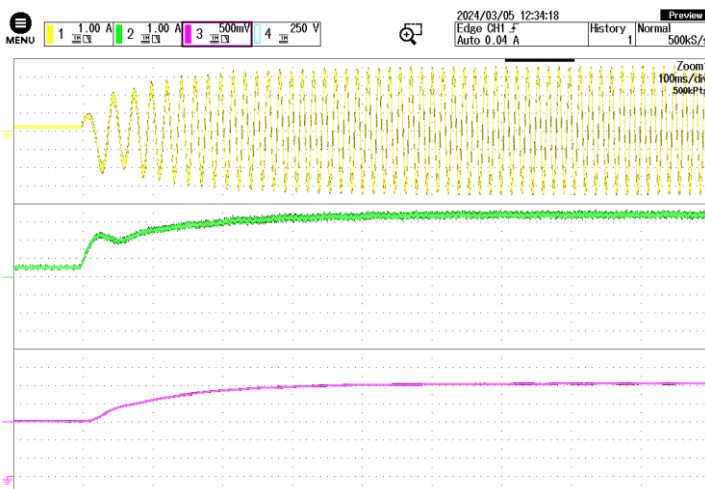


Figura 59: Test di accelerazione svolto sul motore reale, rappresentate in ordine i_a , i_q e la velocità

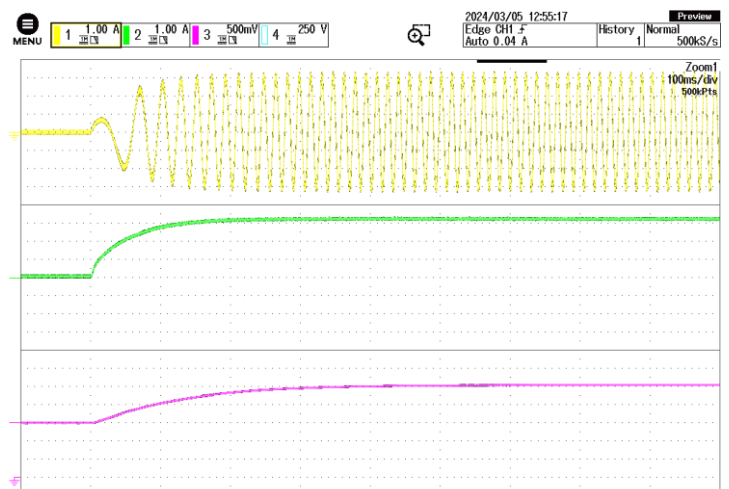


Figura 60: Test di accelerazione svolto sull'emulatore, rappresentate in ordine i_a , i_q e la velocità

Il test successivo è l'inversione di velocità, da 1000 rpm a -1000 rpm e le condizioni in cui è svolto sono le stesse del precedente. Come nel caso precedente si notano aspetti del motore non modellati nella simulazione, ma che nel caso reale possono essere rilevanti.

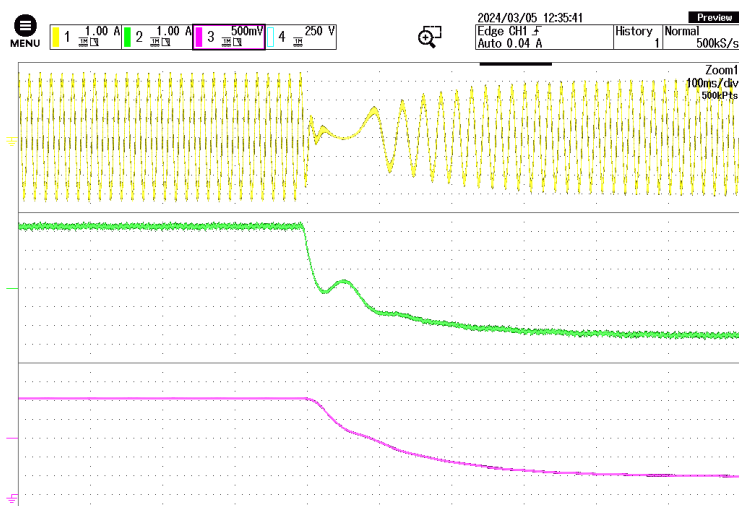


Figura 61: Test dell'inversione di velocità sul motore reale, in ordine sono mostrate i_a , i_q e la velocità

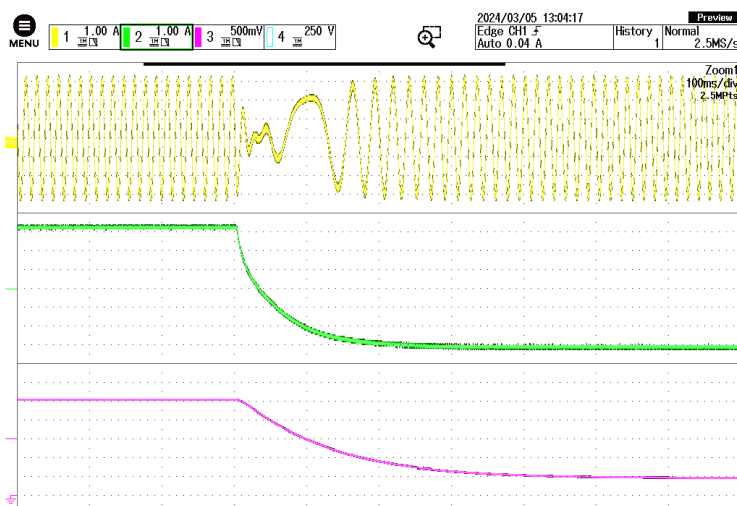


Figura 62: Test dell'inversione di velocità sull'emulatore, in ordine sono mostrate i_a , i_q e la velocità

Riducendo la costante di tempo del filtro sulla velocità a 10 ms, il riferimento di velocità è una rampa con pendenza maggiore. Inevitabilmente la corrente i_q arriva al valore massimo durante il transitorio e si nota un'accelerazione maggiore in entrambi i casi.

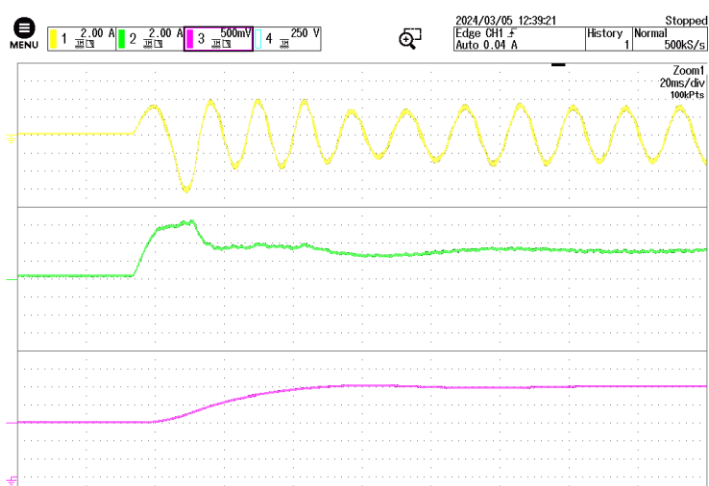


Figura 63: Test di accelerazione svolto sul motore reale, rappresentate in ordine i_a , i_q e la velocità

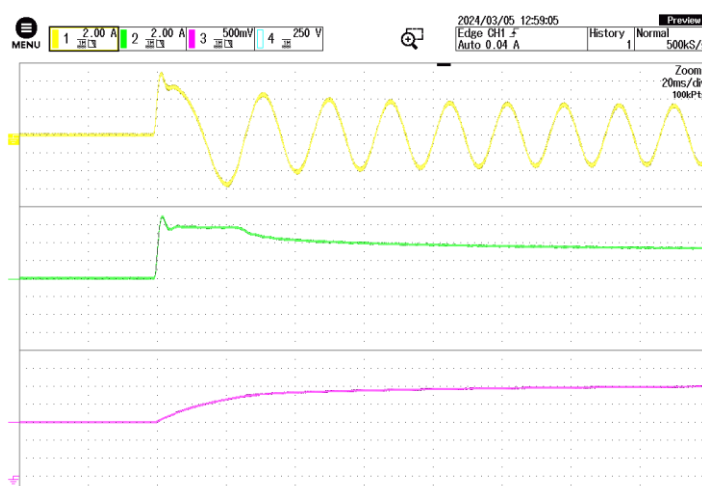


Figura 64: Test di accelerazione svolto sull'emulatore, rappresentate in ordine i_a , i_q e la velocità

Gli ultimi test fatti riguardano la simulazione del carico.

Il test è stato svolto a partire da una condizione iniziale di funzionamento a vuoto ($K_c=0$) alla velocità di 1000 rpm. Per simulare la presa di carico, durante il funzionamento, è stato imposto $K_c = 1.25$. Come mostrato nella *figura 65* dalla traccia azzurra, il carico aumenta e di conseguenza aumentano anche le correnti i_a e i_b raffigurate nel primo riquadro. La velocità, in viola, inizialmente diminuisce e dopo il transitorio torna al valore di regime. La prova opposta, la perdita di carico, presenta un comportamento duale.

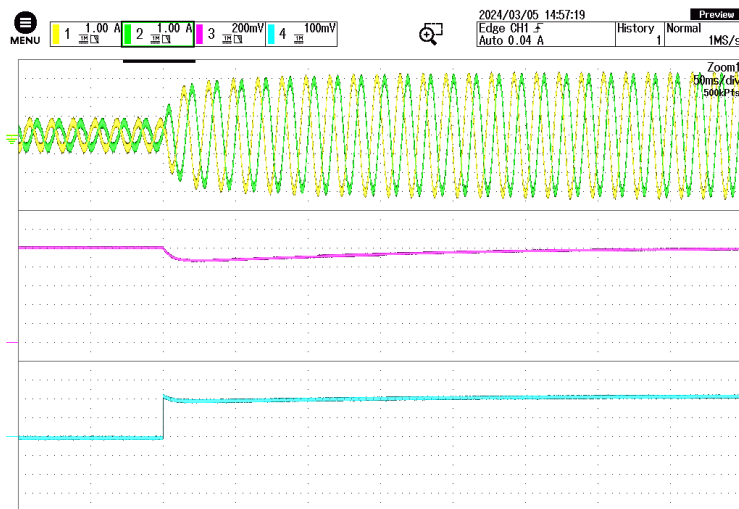


Figura 65: Presa di carico emulata, in ordine sono mostrate le correnti i_a e i_b , la velocità e la coppia

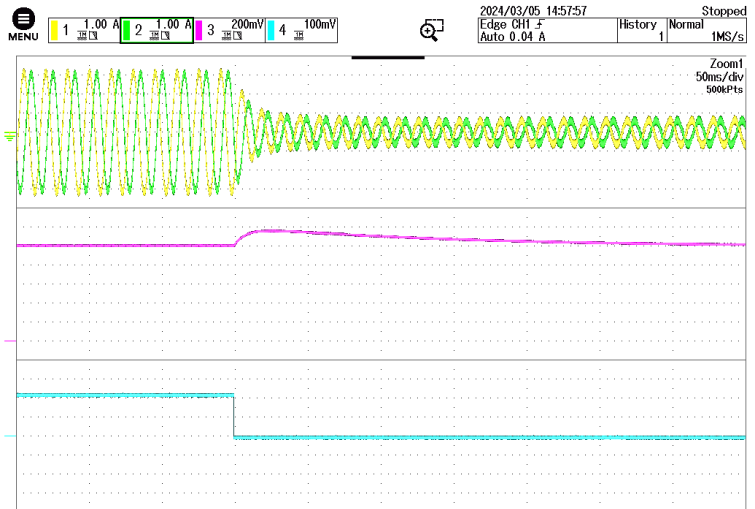


Figura 66: Perdita di carico emulata, in ordine sono mostrate le correnti i_a e i_b , la velocità e la coppia

CONCLUSIONI

Questo progetto di tesi verte sulla realizzazione di un'architettura di Power Hardware In the Loop per l'emulazione di un motore. A partire dalla modellazione del motore sincrono BTSS 1524, sono stati costruiti i modelli tempo continuo, e successivamente tempo discreto, del sistema in questione. È stato necessario il design di una scheda di condizionamento dei segnali per l'isolamento fra gli inverter, le misure utili e il loro filtraggio. Il modello creato è stato implementato su un DSP ad alte prestazioni e l'architettura testata in laboratorio. I risultati ottenuti, in confronto al motore reale, evidenziano un comportamento pressoché identico fra i due. Nel caso reale sono però presenti fattori non modellati che influiscono sul comportamento, come ad esempio gli impuntamenti di coppia.

Gli sviluppi futuri del progetto sono l'introduzione della tensione misurata, al posto del riferimento ottenuto dal controllo del DUT, nel controllo dell'emulatore. Successivamente il modello potrebbe essere complicato introducendo il deflussaggio del motore per testare condizioni operative limite della macchina.

Bibliografia

- [1]. Monti, S. D'Arco and A. Deshmukh, "A new architecture for low cost Power Hardware in the Loop testing of power electronics equipments," *2008 IEEE International Symposium on Industrial Electronics*, Cambridge, UK, 2008, pp. 2183-2188, doi: 10.1109/ISIE.2008.4677306.
- [2]. Bouscayrol, "Different types of Hardware-In-the-Loop simulation for electric drives," *2008 IEEE International Symposium on Industrial Electronics*, Cambridge, UK, 2008, pp. 2146-2151, doi: 10.1109/ISIE.2008.4677304.
- [3]. Y. Boukadida, F. Marignetti, G. M. Casolino, A. Masmoudi, A. Andreoli and M. Albanesi, "Emulation and Testing for Automotive Propulsion Drive Using Two Cascaded Inverters," in *IEEE Transactions on Industry Applications*, vol. 56, no. 2, pp. 1766-1783, March-April 2020, doi: 10.1109/TIA.2019.2959756.
- [4]. C. Nemec, O. Lehmann, M. Heintze and J. Roth-Stielow, "Optimal inductor setup for a power-hardware-in-the-loop machine emulator," *2013 IEEE 10th International Conference on Power Electronics and Drive Systems (PEDS)*, Kitakyushu, Japan, 2013, pp. 364-369, doi: 10.1109/PEDS.2013.6527045.
- [5]. Slide del corso di "Conversione statica dell'energia elettrica", prof A. Tani, AA 21/22.
- [6]. Slide del corso "Azionamenti elettrici per applicazioni industriali ed eoliche", prof. L.Zarri, AA 22/23.
- [7]. Slide del corso "Modellistica circuitale per l'enertronica", prof. M.Ricco, AA 22/23