



UNIVERSITÀ DEGLI STUDI DI PADOVA

FACOLTÀ DI INGEGNERIA

CORSO DI LAUREA IN INGEGNERIA MECCATRONICA

TESI DI LAUREA TRIENNALE

VALIDAZIONE SPERIMENTALE
DELL'ALGORITMO DI MISURA
TRAMITE FPGA DI UNA TENSIONE PWM

Relatore: Ch.mo Prof. ZIGLIOTTO MAURO

Correlatore: Ch.mo Ing. PERETTI LUCA

Laureando: BERDIN PAOLO

Matricola 582453-IMC

ANNO ACCADEMICO 2009-2010

Sommario

Si vuole progettare un azionamento per un motore asincrono mediante l'utilizzo di un algoritmo sensor-less per la stima della posizione al fine di effettuare il controllo di coppia. In particolare andremo ad analizzare e sviluppare la parte relativa alla stima delle tensioni del motore in real-time; tratteremo quindi il condizionamento del segnale, la sua elaborazione ed il collaudo funzionale del sistema.

Verranno introdotte le modalità con cui si sono svolti i test funzionali, lo sviluppo del software per l'FPGA per la stima delle tensioni del motore al fine di renderlo il più ottimizzato possibile in termini di efficienza e risorse impiegate.

Indice

Sommario	iii
Indice	v
1 Introduzione	1
1.1 Necessità di stima delle tensioni	1
1.2 Schema generale del sistema	2
1.3 Non idealità dell'inverter	3
2 Calcolo tensioni concatenate del motore	5
2.1 Idea di integrazione	5
2.2 Introduzione hardware	7
2.2.1 Scheda analogica di acquisizione	7
2.2.2 Stadio di trasformazione da Single Ended Mode a Differential Mode	7
2.2.3 Stadio di condizionamento del segnale di sincronismo	8
2.2.4 Introduzione scheda di sviluppo FPGA	10
2.2.5 Scheda di acquisizione analogica digitale	12
2.2.6 Il convertitore AD	12
2.2.7 Il convertitore DA	14
3 Setup sperimentale	17
3.1 Setup per le misure	17
3.2 Schema funzionale	18
3.3 Collegamento esterno	19
3.4 Collegamento interno	19
3.4.1 Scheda analogica	19
3.4.2 Scheda di interfaccia	20
3.4.3 Scheda FPGA	20
4 Citazione Software	21
4.1 Introduzione	21
4.2 Algoritmo di calcolo della media mobile	21
4.2.1 Estrazione del valore medio	21
4.2.2 Descrizione generale dell'algoritmo	21
5 ANALISI TEORICA DEL SISTEMA DI ACQUISIZIONE DELLE TENSIONI	25
5.1 Sistema di acquisizione a 20Mhz	25
5.1.1 Attenuazione	25
5.1.2 Sensibilità	26
5.2 Sistema di acquisizione a 1,28Mhz	27
5.2.1 Attenuazione	28
5.2.2 Sensibilità	28
6 RISULTATI SPERIMENTALI	31
6.1 Misure in continua	31
6.1.1 Sistema con campionamento a 20Mhz	31
6.1.2 Sistema con campionamento a 1,28Mhz	32
6.1.3 Confronto tra i due sistemi	32

6.2 Misurazioni sperimentali delle tensioni in inverter trifase	33
Conclusioni	37
Ringraziamenti	39
Bibliografia	41

Introduzione

1.1 Necessità di stima delle tensioni

L'azionamento con motore sincrono per attuare il controllo di corrente (coppia) necessita della conoscenza della posizione assoluta del rotore ovvero dell'angolo ϑ_{me} .

Tale angolo costituisce la posizione dell'asse diretto del sistema di riferimento nel quale sono definite le correnti da imporre allo statore per avere la desiderata coppia. Questa è una caratteristica di tali azionamenti e consente di ottenere prestazioni dinamiche altrimenti non possibili, grazie all'accurato controllo vettoriale di corrente che si può realizzare. Al tempo stesso però la necessità di un trasduttore di posizione assoluta (anche quando non vi è alcun controllo di posizione e neanche di velocità) costituisce uno svantaggio in termini di costo, ingombro, affidabilità; la sua eliminazione ricorrendo all'utilizzo di una posizione stimata invece che misurata (azionamenti sensorless) ha assunto un interesse pratico crescente e stimola lo studio di soluzioni innovative ed avanzate. Tale stima viene effettuata da un algoritmo che necessita della misura, in tempo reale, delle tensioni che alimentano il motore.

Le soluzioni per la stima della posizione sono molteplici e differiscono a seconda delle condizioni di lavoro (tipologia di rotore isotropo o anisotropo, velocità nulla o non nulla, stima di velocità o posizione).

Esempi possono essere lo stimatore MRAS (Model Reference Adaptive Systems) o KALMAN.

Il primo permette di stimare la posizione rotorica calcolando una stessa variabile x del motore (vettore spaziale di flusso, f.e.m, variabile reale) a partire da grandezze misurabili ai terminali elettrici (tensioni) usando due modelli di calcolo differenti. Una differenza fra i due risultati calcolati evidenzia un erroneo valore di posizione (*posizione stimata*). Un successivo meccanismo di adattamento (*Adaptation mechanism*) può correggere la posizione stimata fino a far coincidere i due risultati di calcolo, al che la posizione stimata coincide con quella reale.

Il secondo è una procedura ricorsiva a tempo discreto per la stima ottima dello stato di sistemi lineari o non lineari (*Kalman esteso*) assumendo che il sistema sia affetto da un rumore aleatorio di sistema, da un rumore sulle grandezze misurate e, conseguentemente, da un rumore sullo stato stimato (*rumori non correlati*) [1]

Quanto detto vuole mettere in luce il fatto che tali algoritmi necessitano di ricevere in ingresso delle grandezze misurate direttamente quali ad esempio, nel nostro caso, correnti e tensioni. Si può ben capire quindi che eventuali diversità tra la grandezza ricevuta dall'algoritmo e quella reale del carico comporta errori di stima e quindi errori nel fornire coppia.

Pertanto l'assenza di controllo delle grandezze di ingresso in un algoritmo sensorless comporta perdite prestazionali.

1.2 Schema generale del sistema

Al fine di rendere più agevole la comprensione dell'elaborato, viene fornito lo schema a blocchi dell'intero sistema, il quale mette in luce il principio del suo funzionamento.

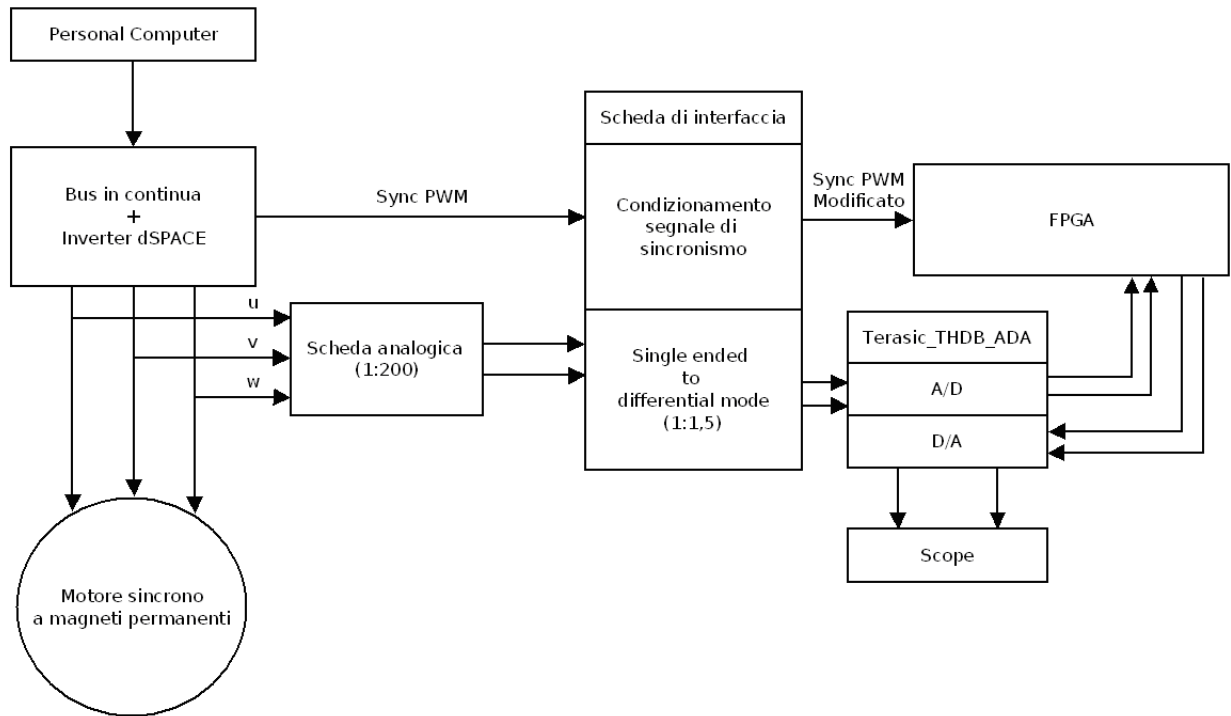


Fig. 1.1: Schema per la misura di tensione.

lo schema di figura Fig. 1.1 comprende i seguenti blocchi:

1. Computer per la gestione del software utilizzato sia per il controllo del motore, che per la misurazione delle tensioni concatenate fornite al motore stesso
2. Scheda dSPACE, per l'interfacciamento tra il computer e l'inverter di tensione
3. Bus in continua per la regolazione della tensione proveniente dalla rete
4. Inverter trifase PWM che fornisce le tre tensioni di fase al motore
5. Scheda analogica, della quale è stato esclusivamente utilizzato lo stadio di attenuazione delle tensioni di ingresso.
6. Scheda di interfaccia tra la scheda dSPACE e la scheda FPGA e tra la scheda analogica e la scheda di conversione Analogico/Digitale (A/D)
7. Scheda di conversione A/D Terasic_THDB_ADA
8. FPGA Altera Cyclone_III Starter Board
9. Motore sincrono a magneti permanenti

Il software di gestione dell'inverter, fornisce i parametri per gestire il bus in continua e il controllo del motore. Il bus in continua fornisce la tensione di ingresso all'inverter, il quale alimenta le fasi del motore sincrono a magneti permanenti. Le tre tensioni di fase vengono prelevate dall'inverter, e acquisite in ingresso dalla scheda analogica, la quale in uscita fornisce le due tensioni concatenate scalate (rispetto alle tensioni di fase di ingresso) di un fattore 1:200. Le due tensioni concatenate single ended entrano nella scheda di interfaccia, la quale fornisce in uscita le due tensioni concatenate in modo differenziale scalate di un fattore 1:1,5. La scheda di interfaccia inoltre riceve dalla dSPACE il segnale di sincronismo della PWM e lo modifica in termini di duty-cycle poichè questo risulta essere troppo piccolo (ordine ns). Le tensioni concatenate differenziali entrano nella scheda Terasic_THDB_ADA, e vengono convertite in segnali digitali. La scheda FPGA riceve in ingresso, le due concatenate digitalizzate e il segnale di sincronismo PWM opportunamente modificato dalla scheda di interfaccia. Il software caricato nell'FPGA si occupa di ricavare dai segnali di ingresso le corrispondenti fondamentali, ovvero i valori medi dei segnali di ingresso. Le uscite dell'FPGA sono collegate alla scheda Terasic_THDB_ADA che fornirà all'oscilloscopio la conversione analogica dei segnali elaborati dall'FPGA stessa; in modo da ottenere in uscita un segnale proporzionale alla media sul periodo di modulazione PWM della tensione concatenata del motore.

1.3 Non idealità dell'inverter

Dallo schema di figura Fig. 1.1 si può notare che le tensioni stimate vengono calcolate a partire da quelle di fase prelevate direttamente dal carico. Questo permette, successivamente, al controllo di effettuare l'eventuale compensazione dei disturbi introdotti dall'inverter: cadute di tensione sui diodi necessari, data la natura induttiva del carico, per fornire una via di richiusura delle correnti ad ogni apertura degli interruttori comandati. I tempi con cui un interruttore di potenza reagisce ai comandi di apertura e chiusura non sono mai perfettamente uguali; per essere certi di comandare la chiusura di un interruttore quando l'altro (dello stesso ramo) abbia effettivamente cessato di condurre si frappone tra i comandi ai due switch un tempo di pausa (tempo morto o dead time). Questo ritardo nella conduzione di uno degli interruttori provoca una distorsione nelle tensioni di uscita rispetto ai segnali di riferimento e risulta opportuno, specialmente negli azionamenti più sofisticati, implementare opportuni algoritmi di compensazione. [2]

Di fatto avremmo potuto prelevare le tensioni in uscita alla DSPACE ma così facendo non saremmo stati in grado di tenere in considerazione l'eventuale presenza di disturbi, introdotti dall'inverter, sulle tensioni applicate al motore e di conseguenza effettuare la compensazione. Tuttavia non possiamo effettuare una compensazione totale in quanto l'invertitore risulta essere troppo non lineare. Ci limitiamo pertanto a compensare il più possibile.

Oltre ad eventuali problemi di stabilità, la conoscenza della distorsione legata ai tempi morti, e la sua corretta compensazione, è importante in tutti gli algoritmi sensorless. Questi algoritmi non necessitano della misura della coppia all'albero motore, la quale viene stimata attraverso modelli matematici del motore controllato. Il corretto funzionamento di tali modelli si basa sulla conoscenza della tensione applicata ai capi del motore; se essa è diversa dalla tensione di riferimento richiesta dagli algoritmi di controllo, la stima della coppia ne verrà influenzata, specialmente a basse velocità, in quanto si è visto che a basse velocità (e quindi basse tensioni) le non linearità risultano più penalizzanti. Il problema della distorsione legata ai tempi morti può essere comunque superato mediante un accurato studio teorico ed un'adeguata strategia di compensazione, che permetta di

considerare la tensione di riferimento generata dagli algoritmi di controllo come la tensione effettivamente generata sul carico. Ciò permette il quasi completo ripristino della linearità del sistema di controllo con benefici per la stabilità dello stesso. Tuttavia la stima delle tensioni tramite compensazione delle non linearità dell'inverter presentano numerosi difetti (non considerazione delle possibili variazioni dei tempi morti (t_d), errori di compensazione qualora la corrente cambi segno all'interno del periodo di commutazione) e si giungerà alla conclusione che per ottenere le massime prestazioni dall'algoritmo sensorless, si preferisce misurare direttamente le tensioni applicate al motore (e non stimarle).

Il fine del progetto sarà pertanto quello di misurare le tensioni sinusoidali reali fornite dall'inverter al motore, per poterle utilizzare, in una fase più avanzata, all'interno di un algoritmo sensorless per la stima di coppia del motore stesso. Per ottenere una stima accettabile si deve far sì che il progetto in tutte le sue fasi sia dotato di una notevole precisione, soprattutto nelle parti prive di eventuali controlli in retroazione capaci di annullare eventuali errori. Quindi, considerata la scarsa bontà della stima delle tensioni tramite tecniche di compensazione delle non linearità dell'inverter, si è scelto di misurare direttamente le tensioni in uscita dall'inverter, attraverso un sistema di acquisizione dati, basato sull'utilizzo di una scheda FPGA.

Calcolo tensioni concatenate del motore

2.1 Idea di integrazione

L'idea di integrazione nasce allo scopo di rendere disponibile al sistema di controllo dell'alimentazione, ad ogni periodo PWM, le tensioni reali concatenate che arrivano dal motore; in particolare la loro componente fondamentale poichè essendo prelevate all'uscita dell'inverter trifase risultano essere onde quadra modulate, PWM, tra un'alimentazione positiva ed una negativa.

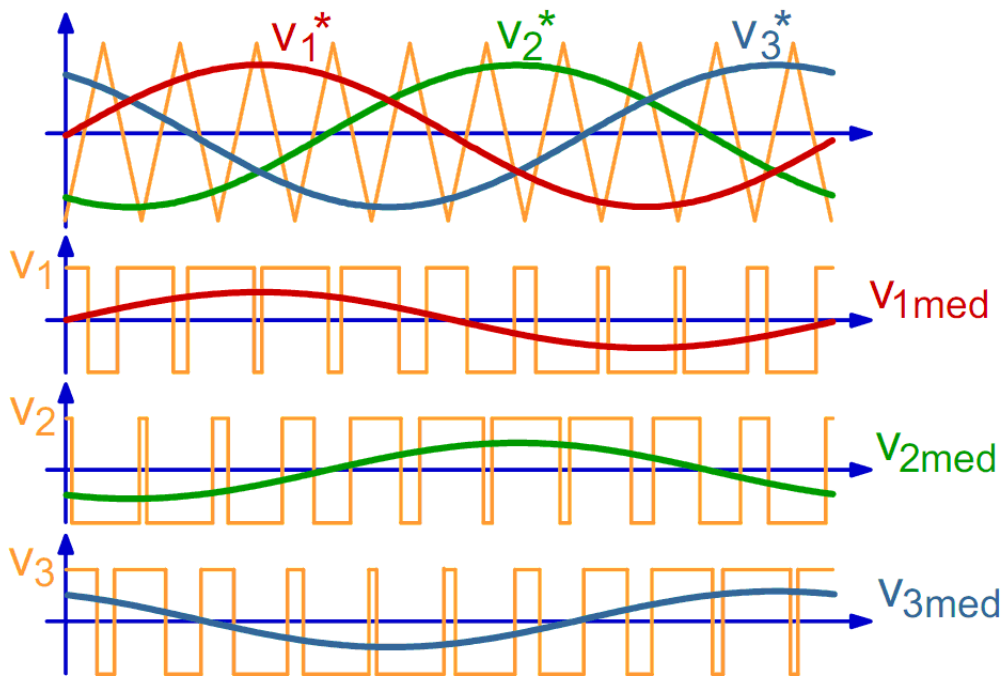


Fig. 2.1: Generazione di tre tensioni sinusoidali simmetriche con portante triangolare comune.

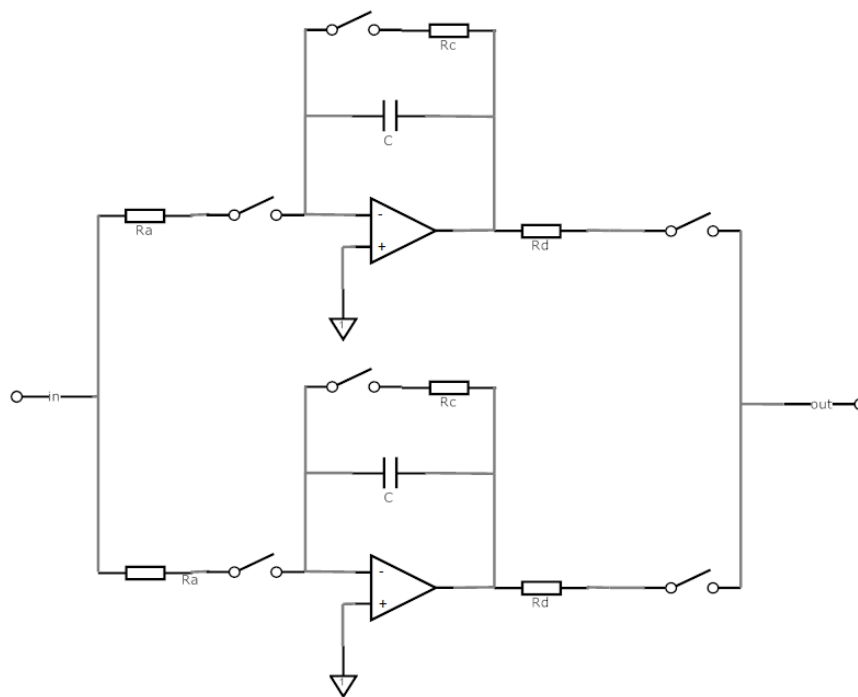


Fig. 2.2: Configurazione dell'integratore comandato.

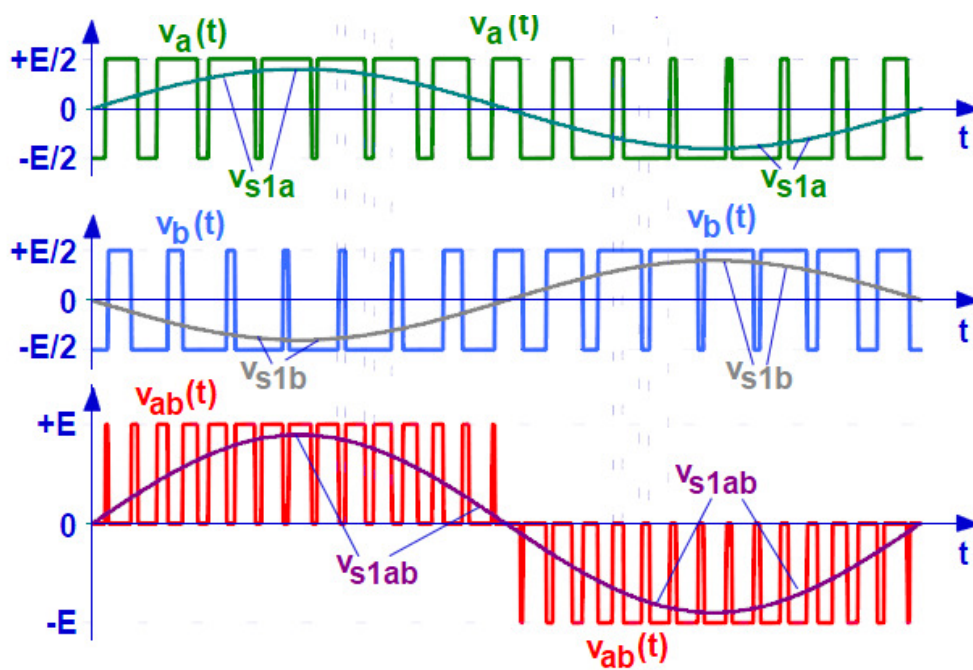


Fig. 2.3: Esempio di tensione concatenata generata dall'inverter e suo valore medio.

Lo stadio di integrazione Fig. 2.2 inizialmente è stato implementato a livello analogico mediante scheda analogica 4108, il quale permette di calcolare effettivamente le medie dei segnali che arrivano dallo stadio di riduzione. I segnali d'ingresso sono le tensioni concatenate del motore Fig. 2.3 (opportunamente ridotte per renderle compatibili con gli integratori utilizzati) ed eventuali segnali di sincronismo. Per ogni tensione concatenata è previsto un doppio canale di integrazione: la doppia integrazione è resa necessaria a causa della configurazione analogica degli integratori, i quali devono per forza avere il tempo di resettarsi all'interno del periodo PWM, altrimenti saturerebbero.

Ogni canale integra un semiperiodo della forma d'onda che arriva in ingresso: mentre un integratore lavora, l'altro ha il tempo di memorizzare il valore calcolato su una capacità di uscita e resettarsi. Lavorano pertanto in maniera speculare e alterna l'uno rispetto all'altro.

Tuttavia ricreiamo la soluzione analogica adottandone una digitale per una questione di precisione in quanto i valori delle componenti (resistenze, condensatori ...) devono essere uguali il più possibile nei 2 rami complementari al fine di avere un funzionamento ottimale. Viene pertanto realizzata tramite software Quartus II, la soluzione digitale che sarà successivamente implementata su scheda FPGA Altera Cyclone III. Quest'ultima verrà introdotta nel successivo capitolo.

2.2 Introduzione hardware

2.2.1 Scheda analogica di acquisizione

La tensione di fase dei motori di media/grossa portata possono arrivare a qualche centinaio di Volt. Devono pertanto essere opportunamente condizionate e attenuate utilizzando una scheda analogica per poter effettuare delle elaborazioni su di esse con i normali amplificatori operazionali della scheda. Il circuito accetta in ingresso due fasi del motore, e, attraverso una rete di resistenze e un amplificatore operazionale, ottiene in uscita la differenza delle tensioni di fase, ovvero la concatenata, scalata di un fattore 1:200 rispetto ai segnali di ingresso. Nella scheda analogica sono presenti due circuiti speculari. Ovvio ricordare che la simmetria tra i due rami del progetto è di fondamentale importanza ai fini di ottenere dei risultati confrontabili, e successivamente utilizzabili per la stima di coppia.

2.2.2 Stadio di trasformazione da Single Ended Mode a Differential Mode

La tensione concatenata single ended scalata di un fattore 1:200 deve essere poi digitalizzata attraverso il convertitore AD della scheda Terasic_THDB_ADA. Tale convertitore accetta segnali in ingresso di tipo differenziale; in particolare su ciascun canale l'AD accetta valori compresi tra $-0,5V$ e $+0,5V$. Pertanto si è tarato il sistema sul caso peggiore, ovvero immaginando di avere in ingresso una tensione pari a $600V_{pp}$. Quindi, considerando l'attenuazione di 1:200 introdotta dalla scheda analogica, ai fini di interfacciare la tensione concatenata scalata con il convertitore AD, viene introdotto uno stadio che attenui di un ulteriore fattore 1:1,5 tale valore di tensione, e che trasformi il segnale da single ended a differenziale. Uno schema a blocchi semplificato è riportato in Fig. 2.4.

Per permettere al convertitore AD di leggere correttamente le tensioni in ingresso, è stata progettata una scheda di condizionamento delle tensioni concatenate, uscenti dallo stadio analogico di riduzione, il layout della stessa è stato disegnato utilizzando il programma freeware Kicad. La parte della scheda di interfaccia relativa al condizionamento delle tensioni concatenate, è costituito da un amplificatore operazionale AD8138 utilizzato per

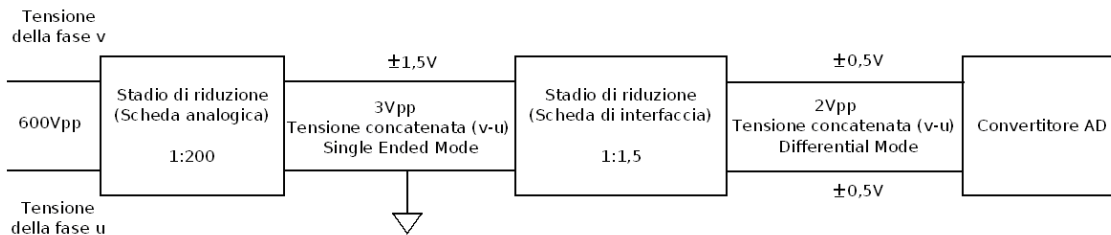


Fig. 2.4: Stadi di attenuazione

trasformare il segnale da single ended a differenziale, e scalarlo di un fattore 1:1,5 attraverso l'opportuno dimensionamento della rete di resistenze in modo da renderlo compatibile con gli ingressi dell'ADC della scheda Terasic_THDB_ADA.

2.2.3 Stadio di condizionamento del segnale di sincronismo

La scheda dSPACE fornisce in uscita il segnale di sincronismo della PWM schematizzato in Fig. 2.5. Tale segnale è caratterizzato da una frequenza di 10kHz (la frequenza del segnale è settabile a piacere), da un valore logico alto pari a 5V e un valore logico basso pari a 0V.

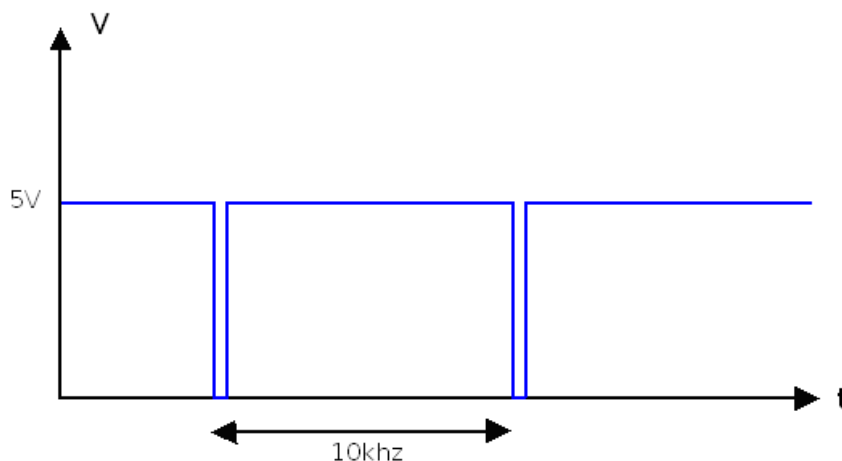


Fig. 2.5: Sincronismo PWM fornito dalla scheda dSPACE (prima del condizionamento realizzato dalla scheda di interfaccia)

La scheda FPGA interpreta correttamente segnali in ingresso con valore logico alto compreso tra +1,7V e +4,1V e con valore logico basso compreso tra -0,5V e +0,7V. È necessario pertanto condizionare il segnale di sincronismo per renderlo compatibile con l'ingresso dell'FPGA. Tale stadio provvederà pertanto a modificare l'ampiezza del segnale in ingresso ed il suo duty-cycle (50%). Si desidera quindi ottenere un segnale simile a quello riportato in Fig. 2.6. Dove sono stati tratteggiati i limiti di tensione per i valori logici alto (in verde) e basso (in rosso) che sarà poi utilizzato dalla scheda FPGA come segnale di sincronismo.

Si riporta in Fig. 2.9 il risultato, ottenuto con Matlab, del processamento di un milione di campioni, estratti dalle misurazioni effettuate con l'oscilloscopio.

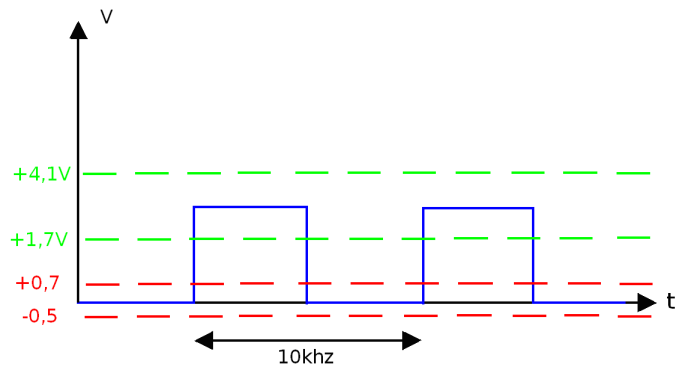


Fig. 2.6: Sincronismo PWM entrante nella scheda FPGA (dopo il condizionamento realizzato dalla scheda di interfaccia)

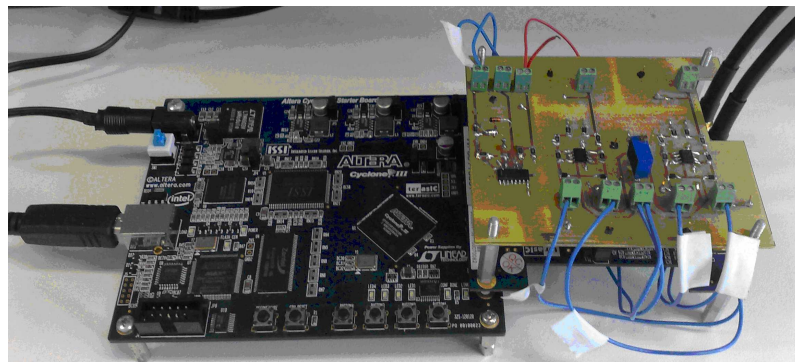


Fig. 2.7: Scheda di interfaccia (collegata al sistema FPGA + ADA Converter)

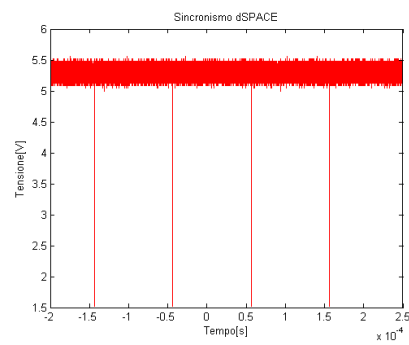


Fig. 2.8: Sincronismo dSPACE

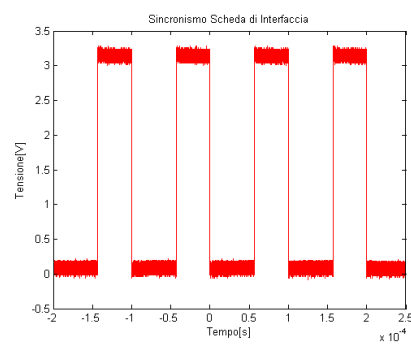


Fig. 2.9: Sincronismo scheda di interfaccia

2.2.4 Introduzione scheda di sviluppo FPGA

Una FPGA (Field Programmable Gate Array) è un insieme di porte logiche programmabili. È dotata di grande velocità di elaborazione, e viene utilizzata per realizzare funzioni logiche, macchine sequenziali sincrone e asincrone, reti combinatorie. È inoltre adatta al controllo di led e display. Per lo sviluppo del progetto è stata utilizzata l'FPGA_Cyclone_III prodotta dalla Altera, e montata dalla stessa azienda sulla scheda di test Cyclone_III_FPGA_Starter_Board. L'FPGA_Cyclone_III_Starter_Board è programmabile attraverso l'ambiente di sviluppo Quartus II, rilasciato dall'Altera. Tale ambiente facilita la programmazione della scheda, fornendo agli utenti molti blocchi preprogrammati, facenti numerosissime funzioni. L'utente può comunque realizzare nuovi blocchi attraverso il linguaggio di programmazione Verilog VHDL. La scheda FPGA può espandere le sue funzionalità attraverso il collegamento ad altre schede, sfruttando il connettore HSMC (High Speed Mezzanine Card). La scheda FPGA si collega alla scheda di acquisizione analogico/digitale attraverso il medesimo connettore. In Fig. 2.10 è riportata l'FPGA_Cyclone_III_Starter_Board con le sue caratteristiche hardware principali.

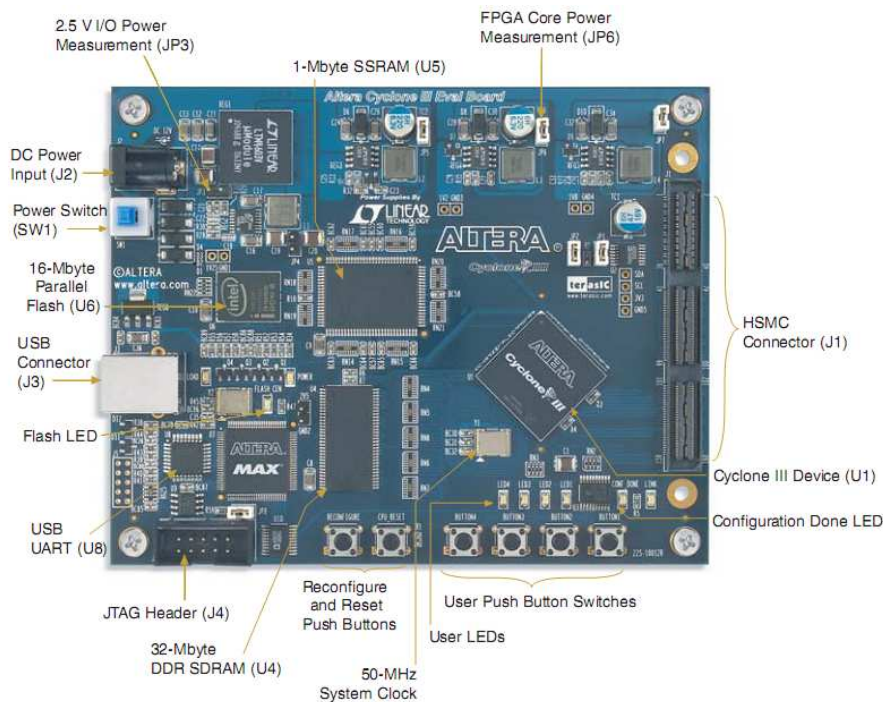


Fig. 2.10: FPGA_Cyclone_III_Starter_Board

La scheda FPGA è caratterizzata dai seguenti elementi:

- FPGA Altera Cyclone III EP3C25, contenuta in un package avente 324 pin, è composta da 25000 elementi logici programmabili, 0,6Mbit di blocchi di memoria e 16 blocchi moltiplicatori.
- Connettore HSMC (High Speed Mezzanine Card) per l'interfacciamento con schede aggiuntive. Attraverso questo connettore vengono inoltre forniti due segnali rispettivamente a 3,3V e 12V, e i relativi collegamenti di massa, necessari ad alimentare le schede di espansione connesse. Il connettore HSMC in totale presenta 84 pin di I/O bidirezionali, ad eccezione dei pin riservati alla trasmissione dei segnali di clock. I pin dell'HSMC sono caratterizzati da un livello logico alto pari a 2,5V. L'FPGA riconosce un livello logico alto all'interno di un range di tensione compreso tra 1,7V

e 4,1V, viceversa riconosce un livello logico basso all'interno di un range di tensione compreso tra -0,5V e 0,7V.

- Connettore USB per l'interfacciamento con il PC.
- Memoria DDR SDRAM da 32Mbyte.
- Memoria Flash da 16Mbyte per la configurazione dell'FPGA e il salvataggio dei dati allo spegnimento.
- Memoria SSRAM ad alta velocità da 1 Mbyte.
- 4 pulsanti programmabili, che, se premuti, forniscono un valore logico alto, finché non vengono rilasciati. 2 pulsanti non programmabili: il pulsante System-reset e il pulsante User-reset. Rispettivamente il primo serve per la riconfigurazione dell'FPGA attraverso la memoria flash; il secondo serve per resettare i progetti caricati nel dispositivo.
- 4 led programmabili. 3 led non programmabili: Power Led, Configuration Led, Flash Signal Led. Rispettivamente l'accensione del primo indica che la scheda è alimentata, l'illuminazione del secondo indica una programmazione dell'FPGA avvenuta con successo, l'attivazione del terzo indica un accesso alla memoria Flash.
- Circuito per la generazione del clock interno alla scheda, che fornisce un segnale a 50Mhz. Tutti gli altri segnali di clock necessari vengono ricavati da esso attraverso dei PLL (anelli ad aggancio di fase) Tali dispositivi forniscono i clock alla memoria flash, alla memoria SSRAM, al connettore HSMC (e di conseguenza alle schede collegate) E' inoltre possibile programmare a piacere i PLL in modo da fornire, a seconda delle esigenze, delle particolari frequenze di clock ai pin di I/O

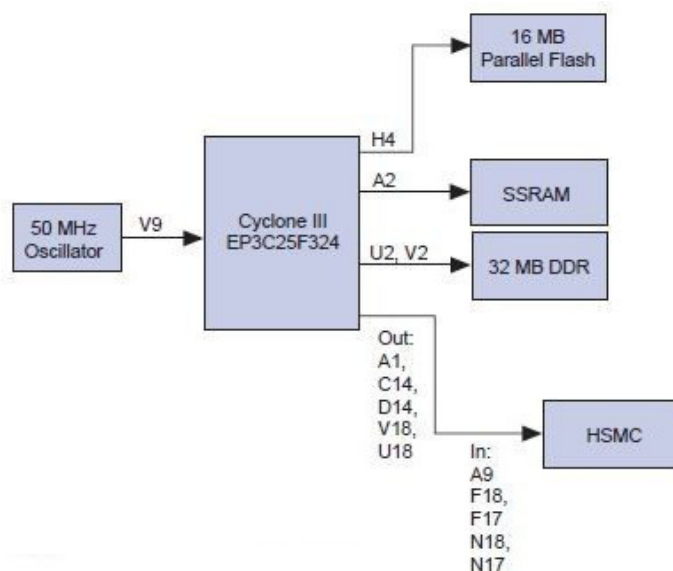


Fig. 2.11: Schema a blocchi del circuito di clock della scheda FPGA_Cyclone_III_Starter_Board

La figura Fig. 2.10 è stata estrapolata dalla [3], alla quale si rimanda per ulteriori approfondimenti.

2.2.5 Scheda di acquisizione analogica digitale

La scheda di acquisizione analogica digitale Terasic_THDB_ADA, viene utilizzata nel progetto, a monte dell'FPGA per trasformare in digitale i segnali analogici provenienti dalla scheda di interfaccia, e a valle dell'FPGA per trasformare il risultato dell'elaborazione dell'FPGA da digitale ad analogico, col fine di poterlo visualizzare correttamente sull'oscilloscopio. Per poter realizzare quanto sopra descritto la scheda Terasic_THDB_ADA è dotata di due convertitori AD e due convertitori DA, le cui caratteristiche saranno approfondite nelle sottosezioni 2.3.2, 2.3.3. Per i 2 ingressi e le 2 uscite ci sono 4 connettori BNC. Tali connettori sono una famiglia di connettori unipolari a baionetta usati per l'intestazione di cavi coassiali. In Fig. 2.12 si riporta una foto della scheda di acquisizione analogica digitale. La scheda Terasic_THDB_ADA si collega alla scheda FPGA_Cyclone_III_Starter_Board tramite il connettore HSMC apposito. Si riporta in Fig. 2.13 una foto delle due schede collegate tra loro.

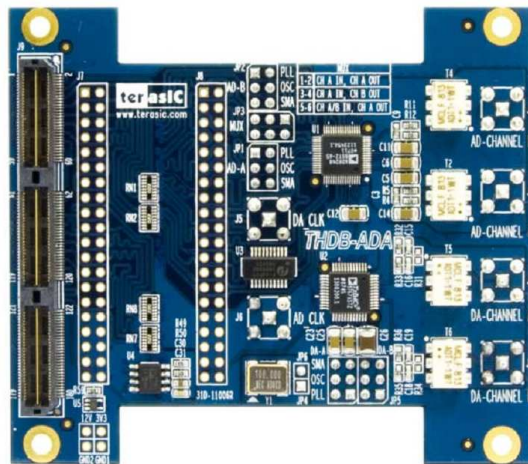


Fig. 2.12: Terasic_THDB_ADA

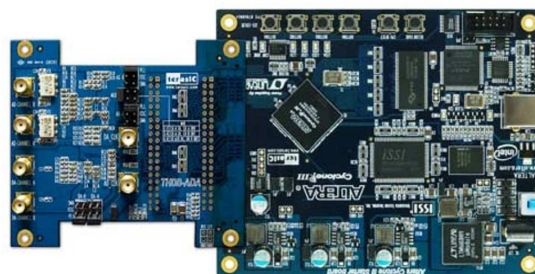


Fig. 2.13: Collegamento tra Terasic_THDB_ADA e FPGA_Cyclone_III_Starter_Board

2.2.6 Il convertitore AD

Il segnale opportunamente scalato e reso di tipo differenziale, può essere correttamente letto dal convertitore AD della scheda Terasic_THDB_ADA. Il convertitore AD trasforma

la tensione in ingresso in un segnale digitale, che viene poi letto dal programma precaricato nella scheda FPGA.

Il convertitore AD9248 è dotato delle seguenti caratteristiche principali: [4]

1. due AD con profondità di 14bits
2. due ingressi analogici, con span di ingresso compreso tra 1Vpp e 2Vpp
3. due uscite digitali offset binary (o complemento a due)
4. velocità limite di campionamento pari a 65MSPS (Mega Sample Per Second)
5. alimentato a 3V (tensione supportata da 2,7V a 3,6V)

In Fig. 2.14 si riporta il diagramma funzionale del convertitore AD9248 compreso di pin-out.

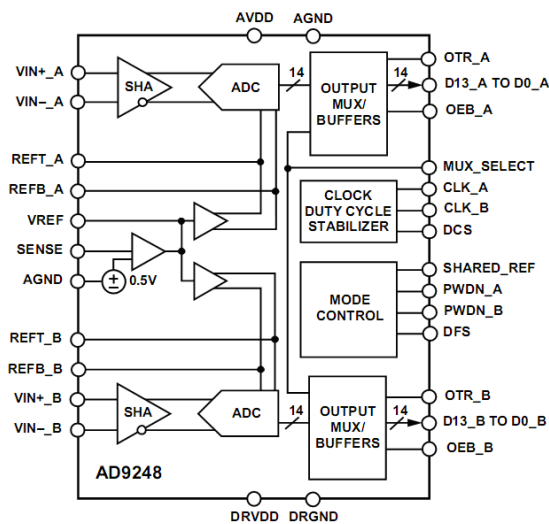


Fig. 2.14: AD9248, diagramma funzionale

L'AD9248 fornisce i dati digitali in uscita dopo un ritardo di propagazione (t_{PD} = time propagation delay) dopo il fronte di salita del segnale di clock. Tale situazione è illustrata graficamente in Fig. 2.15.

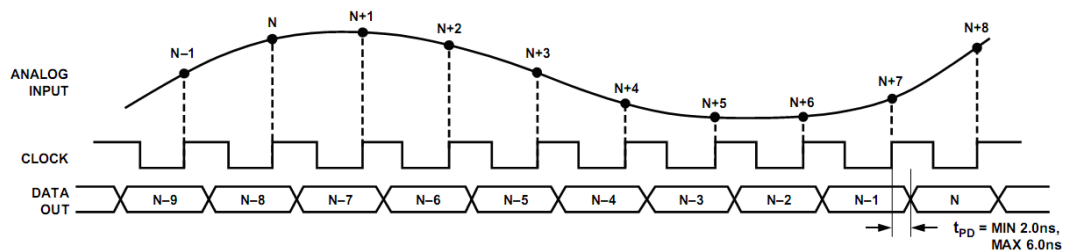


Fig. 2.15: Ritardo di propagazione dei dati di uscita

Quindi il programma caricato nell'FPGA fornirà la stessa frequenza di clock ai convertitori AD e agli accumulatori. In più gli accumulatori dovranno iniziare a processare i dati

in ingresso con un ritardo, impostato sul caso peggiore, pari a 6ns; in modo da rimanere sincronizzati con i dati in uscita dai due ADCs.

2.2.7 Il convertitore DA

I dati finali, opportunamente elaborati dal programma caricato nell'FPGA, devono poter essere visualizzati sull'oscilloscopio per opportune misurazioni e verifiche sperimentali. Vengono pertanto convertiti in analogico sfruttando la parte della scheda Terasic_THDB_ADA dedicata a tale scopo; in particolare il convertitore AD9767. [5]

Il convertitore AD9767 è dotato delle seguenti caratteristiche principali:

1. due DAC con profondità di 14 bits
2. velocità di campionamento fino a 125MSPS
3. capacità di operare a 5V o a 3,3V

In figura Fig. 2.16 si riporta il diagramma funzionale del convertitore AD9767 compreso di pin-out.

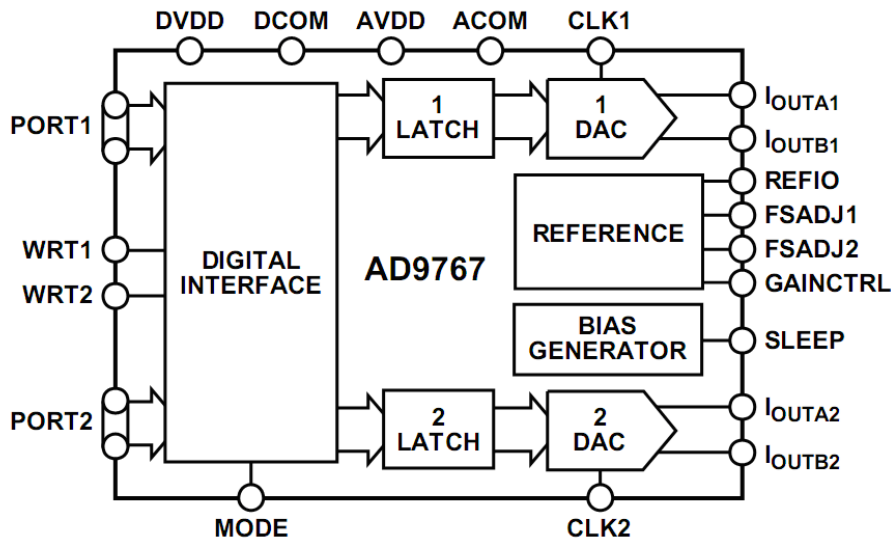


Fig. 2.16: AD9767, diagramma funzionale

Si mostra in Fig. 2.17 lo schema a blocchi di configurazione per l'utilizzo del riferimento interno di tensione.

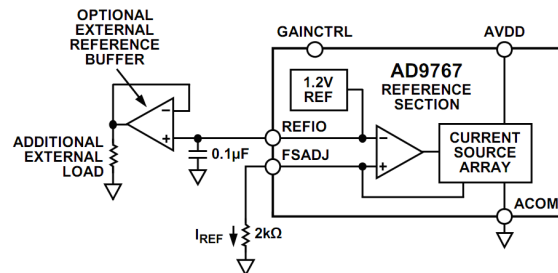


Fig. 2.17: Configurazione del riferimento interno di tensione

Entrambi i DACs dell'AD9797 contengono un amplificatore di controllo utilizzato per regolare il fondo scala della corrente di uscita (I_{OUTFS}). L'amplificatore di controllo è

configurato come un convertitore Tensione-Corrente come mostrato in Fig. 2.17, in modo che la sua corrente di uscita (I_{REF}) sia determinata dal rapporto tra la tensione di riferimento V_{REFIO} e la resistenza esterna R_{SET} come evidenziato nella 2.2.1.

$$I_{REF} = \frac{V_{REFIO}}{R_{SET}} \quad (2.2.1)$$

La corrente di fondo I_{OUTFS} , è data dalla 2.2.2.

$$I_{OUTFS} = 32 \cdot I_{REF} \quad (2.2.2)$$

Ne segue, essendo $V_{REFIO}=1,2V$ e $R_{SET}=2k\Omega$:

$$I_{REF} = \frac{V_{REFIO}}{R_{SET}} = \frac{1,2V}{2k\Omega} = 600\mu A \quad (2.2.3)$$

$$I_{OUTFS} = 32 \cdot I_{REF} = 32 \cdot 600\mu A = 19,2mA \quad (2.2.4)$$

Si chiarisce di seguito la logica utilizzata dai due DACs per il calcolo delle correnti di uscita. Entrambi i DACs nell'AD9767 forniscono correnti di uscita complementari, I_{OUTA} e I_{OUTB} . I_{OUTA} fornisce una corrente vicina al valore di fondo scala (I_{OUTFS}) quando tutti i bits del segnale digitale in ingresso sono pari a uno, come evidenziato nella 2.2.5.

$$DAC_CODE = 2^{14} - 1 = 16383 = 11111111111111 \quad (2.2.5)$$

Mentre I_{OUTB} , l'uscita complementare, fornisce corrente nulla. Le correnti I_{OUTA} e I_{OUTB} sono in funzione del codice digitale in ingresso e della corrente di fondo scala I_{OUTFS} , e possono essere espresse dalle equazioni 2.2.6 e 2.2.7.

$$I_{OUTA} = \frac{DAC_CODE}{16384} I_{OUTFS} \quad (2.2.6)$$

$$I_{OUTB} = \frac{16383 - DAC_CODE}{16384} I_{OUTFS} \quad (2.2.7)$$

Le due correnti di uscita tipicamente comandano una resistenza di carico.

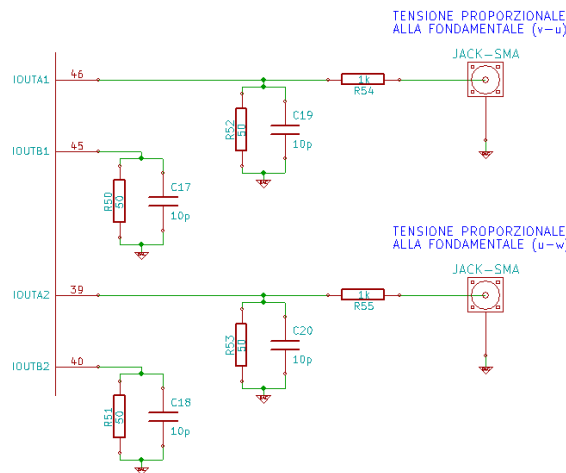


Fig. 2.18: Uscite analogiche

Le tensioni single-ended di uscita su I_{OUTA} e I_{OUTB} sono semplicemente date dalle 2.2.8 e 2.2.9.

$$V_{OUTA} = I_{OUTA} \cdot R_{LOAD} \quad (2.2.8)$$

$$V_{OUTB} = I_{OUTB} \cdot R_{LOAD} \quad (2.2.9)$$

In Fig. 2.19 viene illustrato uno schema rappresentante l'uscita di uno dei due convertitori DA (per l'altro le considerazioni sono duali). La configurazione adottata in laboratorio permette di fornire in uscita un range unipolare compreso tra 0V e 960mV. Questo perché, come spiegato in questa sottosezione, la corrente di uscita (I_{OUTA}), con valore di fondo scala pari a $I_{OUTFS} = 19,2\text{mA}$, scorre attraverso una resistenza da 50Ω . Il collegamento viene fatto in serie ad una resistenza da $1k\Omega$ (vedi Fig. 2.19). Tale resistenza non influenza la misura, in quanto l'oscilloscopio ha un'alta impedenza interna (pari a circa $1M\Omega$) e quindi il ramo composto dalla serie delle due resistenze ($1M\Omega + 1k\Omega$) non assorbe corrente, pertanto la resistenza da $1k\Omega$ non perturba la misurazione.

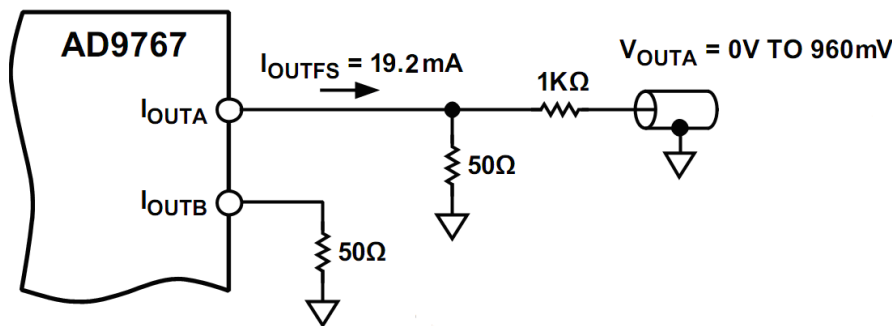


Fig. 2.19: Tensione analogica d'uscita

Pertanto tensioni nulle in ingresso al sistema corrisponderanno ad una tensione di uscita pari a 480mV. Tensioni positive in ingresso al sistema corrisponderanno a tensioni di uscita comprese tra 480mV e 960mV. Tensioni negative in ingresso al sistema corrisponderanno a tensioni d'uscita comprese tra 0V e 480mV.

Il convertitore DA processa ingressi digitali a 14bit, e la sua uscita ha un valore di fondo scala pari a 960mV; quindi la sensibilità, espressa in μV , del valore di uscita è data dalla 2.2.10.

$$Sensibilità_d'Uscita = \frac{960mV}{16384} = 58,594\mu V \quad (2.2.10)$$

Setup sperimentale

3.1 Setup per le misure

E' stato previsto di predisporre entrambe le schede di acquisizione, analogica e digitale, all'interno di un apposito contenitore chiuso verso l'esterno che rende disponibili, attraverso opportuni connettori, gli ingressi e le uscite delle schede per interfacciarsi con l'azionamento e la rete elettrica. In Fig. 3.1 viene mostrata la struttura del contenitore e la disposizione delle schede nonchè i collegamenti adottati.



Fig. 3.1: Box di acquisizione

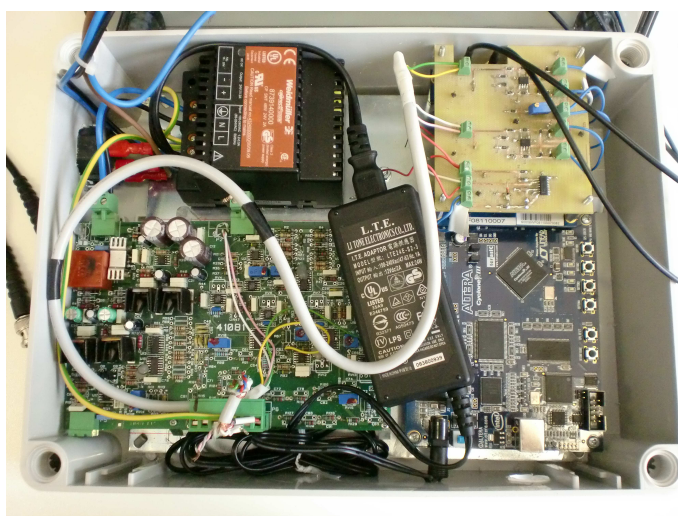


Fig. 3.2: Contenuto interno del contenitore

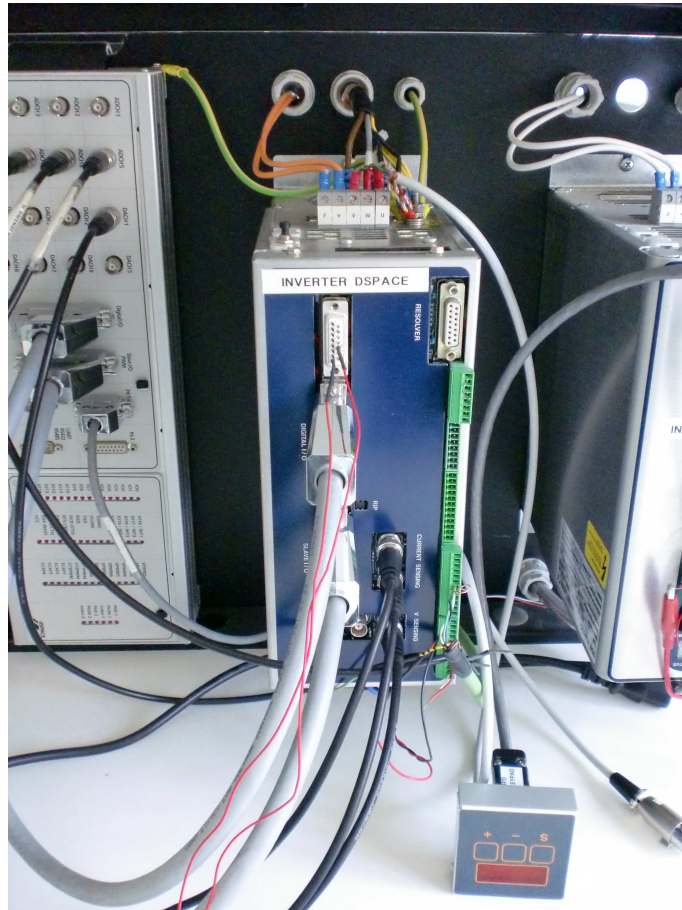


Fig. 3.3: Sincronismo DSPACE

3.2 Schema funzionale

Viene riportato di seguito lo schema funzionale del box di acquisizione Fig. 3.4.

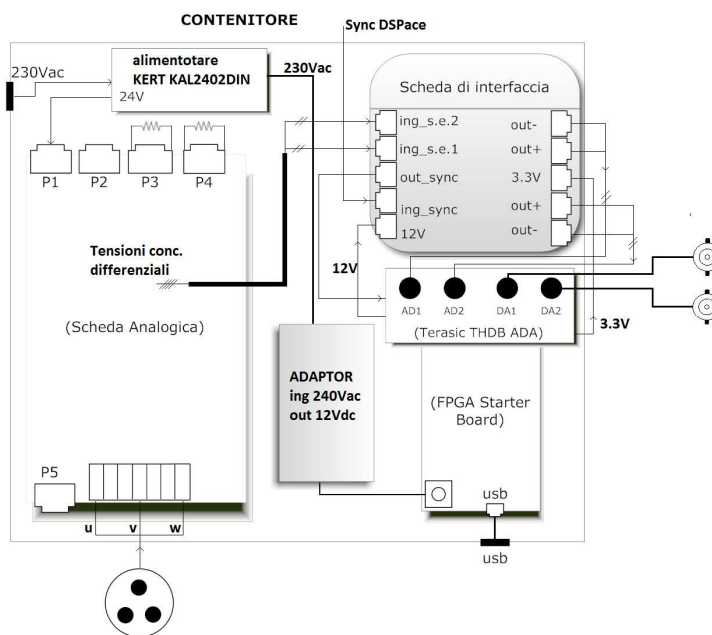


Fig. 3.4: Diagramma funzionale

3.3 Collegamento esterno

Il contenitore presenta sulla facciata superiore un connettore a tre vie femmina di ingresso attraverso il quale vengono rese disponibili al sistema di acquisizione le tre tensioni di fase del motore e due connettori bipolari di uscita i quali forniscono le due tensioni concatenate single-ended elaborate dalla scheda digitale interna (segnali proporzionali alla media su un periodo PWM delle tensioni concatenate del motore). Lateralmente sono previsti un connettore di alimentazione femmina per il collegamento alla rete elettrica dell'alimentatore interno, un connettore per l'acquisizione del segnale di sincronismo proveniente dalla DSPACE nonchè un connettore USB tipo B femmina per il collegamento della scheda FPGA al computer.

3.4 Collegamento interno

3.4.1 Scheda analogica

Per rendere disponibili i 24 Vdc alla scheda analogica si è utilizzato un alimentatore KERT KAL2402DIN, il quale, dalla rete di alimentazione, genera un segnale continuo a 24V, in grado di erogare fino a 2A. Tale alimentatore fornisce inoltre l'alimentazione alla scheda FPGA.

Le tre tensioni di fase (u,v,w) vengono portate in ingresso alla scheda analogica attraverso una morsettiera a sette vie, vedi fig1. Attraverso una rete resistiva esse arrivano in ingresso ad un amplificatore operazionale che calcola la differenza tra di esse e restituisce un segnale proporzionale alla tensione concatenata del motore, ridotta di un fattore 1:200, vedi Fig. 3.5. Dalle tre fasi del motore vengono generate le due tensioni concatenate ($OUT_{DIFF1} = V_{vu}$, $OUT_{DIFF2} = V_{wv}$).

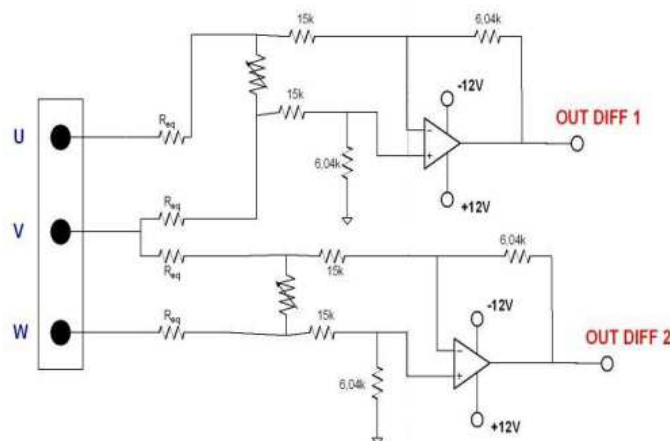


Fig. 3.5: Schema elettrico dello stadio di riduzione

Le due tensioni concatenate vengono quindi prelevate dalla scheda analogica e fornite in ingresso alla scheda di interfaccia mediante cavo schermato, vedi Fig. 3.4.

3.4.2 Scheda di interfaccia

La scheda di interfaccia riceve direttamente dalla scheda Terasic_THDB_ADA le alimentazioni +12V e +3.3V da fornire ai circuiti integrati utilizzati.

Tale scheda presenta 8 ulteriori connettori a vite a 2 vie:

- Connettore ING SYNC: Riceve il segnale di sincronismo dalla DSPACE da adattare alle specifiche dell'FPGA.
- Connettore OUT SYNC: Rende disponibile in uscita il segnale di sincronismo adattato al fine di renderlo compatibile con l'ingresso dell'FPGA.
- Connettori ING SINGLE ENDED CONC 1/2: Ricevono i segnali concatenati single-ended scalati di un fattore 1:200 estratti dalla scheda analogica.
- Connettori OUT+ (2 connettori a vite), OUT- (2 connettori a vite): I segnali concatenati vengono trasformati da single-ended a differenziali e resi disponibili in uscita mediante questi connettori alla scheda FPGA.

3.4.3 Scheda FPGA

Scheda FPGA_Cyclone_III_Starter_Board e Terasic_THDB_ADA vengono collegate assieme mediante connettore HSMC. La prima riceve l'alimentazione 12V dall'alimentatore interno e permette il collegamento al computer mediante connettore usb tipo B femmina. La scheda Terasic_THDB_ADA riceve dalla scheda di interfaccia le due tensioni differenziali, direttamente in ingresso ai due convertitori A/D. Direttamente in quanto i trasformatori presenti a monte dei convertitori introducevano un limite di banda inferiore, producendo una limitazione dovuta al fatto che le tensioni concatenate del motore hanno una banda di frequenza che si estende fino alla componente continua. Si è dovuto pertanto intervenire a livello hardware eliminandoli.

Il segnale di sincronismo adattato dalla scheda di interfaccia, in uscita dal connettore OUT SYNC, viene fornito alla scheda Terasic_THDB_ADA; il pin scelto per l'acquisizione di tale segnale è stato scelto sulla base di quelli liberi di ingresso: pin 11 dell'Header 2x20 J7.

Il risultato dell'elaborazione in uscita dai convertitori D/A viene reso disponibile per una sua visualizzazione tramite oscilloscopio, mediante connettori bipolari disposti sulla facciata superiore del contenitore principale.

Citazione Software

4.1 Introduzione

Per la realizzazione del programma da caricare nell'FPGA è stato utilizzato l'ambiente di sviluppo Altera Quartus II Web Edition versione 9.1 Service Pack 1, scaricabile gratuitamente dal sito dell'azienda riportato in [6]. Tale software di programmazione mette a disposizione una vasta gamma di blocchi preprogrammati facenti molte funzioni. In questo progetto tale funzionalità è stata largamente utilizzata. A seconda delle esigenze, sono stati inoltre creati dei blocchi, utilizzando il linguaggio di programmazione Verilog HDL.

4.2 Algoritmo di calcolo della media mobile

4.2.1 Estrazione del valore medio

Il software caricato nell'FPGA deve essere in grado di calcolare il valore medio, in ogni periodo di commutazione della PWM, dei segnali digitali campionati in ingresso. Quindi in generale, dal punto di vista matematico deve essere in grado di implementare l'equazione 4.2.1

$$V_N = \frac{\sum_{i=0}^N v_i}{N} \quad (4.2.1)$$

Tale equazione, restituisce la media degli N campioni processati. Immaginando ora che in un periodo di PWM siano presenti N campioni, se l'operazione di media venisse applicata su più periodi PWM, si otterrebbe la media mobile del segnale in ingresso. Tale equazione è implementabile, a patto di fare alcuni ragionamenti sulla divisione per gli N campioni. Nell'algoritmo di controllo, è necessario, per poter visualizzare i dati in uscita effettuare un troncamento degli stessi. L'operazione di troncamento è paragonabile alla divisione per N campioni nell'operazione di media.

Ovviamente troncando i dati, non si divide effettivamente per N, ma per un numero $\geq N$ (espresso in base 2), a seconda della frequenza di campionamento scelta. Ovvero, a parità di valori entranti in ingresso, maggiore sarà la frequenza di campionamento, maggiore sarà il numero accumulato nel periodo; quindi maggiore dovrà essere il troncamento da effettuare.

4.2.2 Descrizione generale dell'algoritmo

Si premette, come già spiegato nella descrizione hardware del sistema, che il progetto consiste nell'elaborazione di due concatenate del motore. Da qui in poi si farà riferimento ad una sola delle due. In quanto tutte le operazioni eseguite, e le considerazioni fatte per l'una, risultano identiche per l'altra.

L'algoritmo di controllo è stato riportato nello schema a blocchi di Fig. 4.1.

In ingresso al convertitore AD entra la tensione concatenata, prelevata dall'inverter, attenuata e resa differenziale. Il convertitore AD, operando con una frequenza pari a 20Mhz, trasforma il segnale di ingresso in un segnale digitale a 14bit in codifica Offset Binary. Il segnale digitalizzato entra nell'FPGA e viene processato dal software caricato nella scheda, come spiegato di seguito.

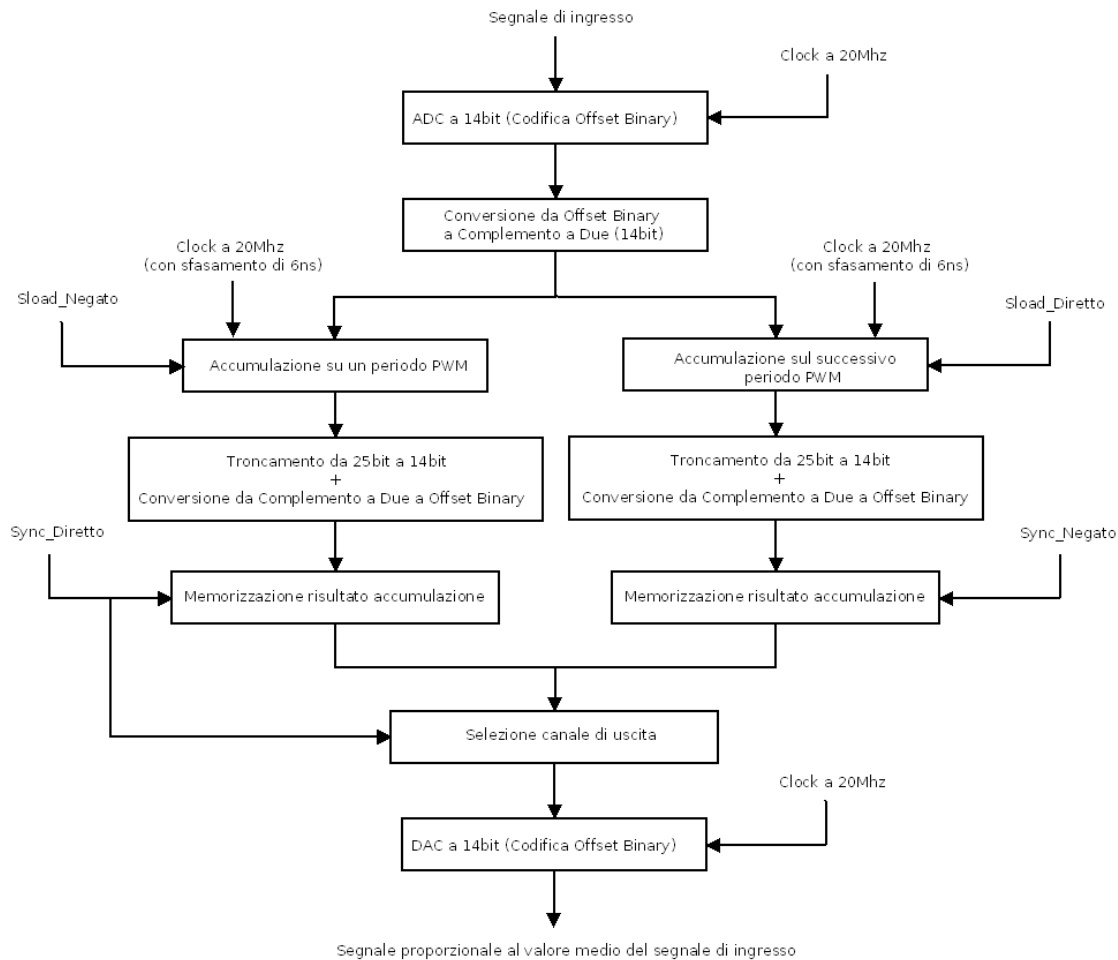


Fig. 4.1: Algoritmo di calcolo della media mobile

Il segnale digitale subisce una conversione da Offset Binary a Complemento a Due; si preferisce questo in quanto tale codifica permette ai campioni digitali di ingresso, all'uscita dell'AD, di essere sommati indipendentemente dal loro segno. Successivamente viene portato contemporaneamente in ingresso a due accumulatori, che operano a 20Mhz con uno sfasamento iniziale di 6ns, e che sono attivi in maniera alternata, grazie all'utilizzo dei segnali *Sload_Diretto* e *Sload_Negato*. Tali segnali sono ricavati dal segnale di Sincronismo PWM come riportato nella Fig. 4.2.

Ogni accumulatore nel periodo di PWM in cui è attivo effettua la sommatoria dei dati in ingresso a 14bit; la somma totale viene aggiornata progressivamente su un registro a 25bit, interno a ciascun accumulatore. Lo sdoppiamento del canale è stato necessario per fornire agli accumulatori il tempo di resettarsi. Ciò avviene nel periodo PWM in cui l'accumulatore dell'altro ramo sta effettuando la sommatoria.

Pochi istanti prima del reset dell'accumulatore, il dato di uscita (contenente la sommatoria dei dati in ingresso in un periodo PWM) viene riportato su 14bit e riconvertito in Offset Binary (per essere compatibile con il DAC d'uscita); infine viene memorizzato in un registro apposito. Tale registro aggiorna la sua uscita in funzione dei segnali *Sync_Diretto* e *Sync_Negato*, (vedi Fig. 4.2) a seconda del canale di riferimento, con la seguente modalità.

Quando l'accumulatore del proprio ramo è attivo l'uscita del registro è trasparente all'ingresso, pochi istanti prima del reset dell'accumulatore, il registro blocca l'uscita al valore finale della sommatoria sul periodo, uscente dall'accumulatore. Infine il bloc-

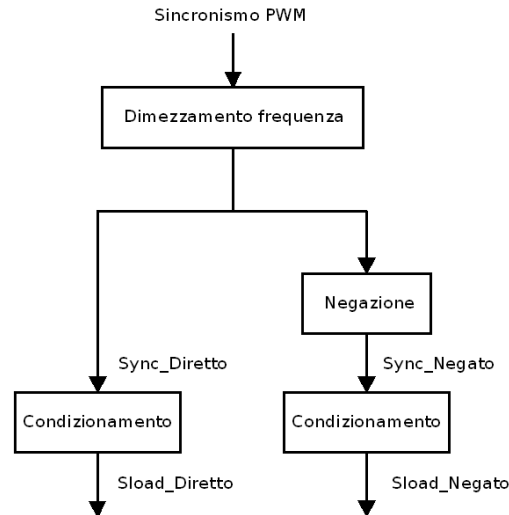


Fig. 4.2: Segnali di sincronismo

co **Selezione canale di uscita**, seleziona in maniera alternata, il canale corretto da riportare in uscita. Questo è possibile grazie all'utilizzo del segnale **Sync_Diretto**.

A valle dell'elaborazione software il convertitore DA, operando con una frequenza di 20Mhz, effettua la conversione da digitale ad analogico.

L'ADC produce dei dati d'uscita a 14bit in formato Offset Binary (OB). Tale formato ha delle corrispondenze col Complemento a Due (C2) come si può osservare nella Tab. 4.1.

Valore Decimale	Offset Binary	Complemento a Due
7	1111	0111
6	1110	0110
5	1101	0101
4	1100	0100
3	1011	0011
2	1010	0010
1	1001	0001
0	1000	0000
-1	0111	1111
-2	0110	1110
-3	0101	1101
-4	0100	1100
-5	0011	1011
-6	0010	1010
-7	0001	1001
-8	0000	1000

Tab. 4.1: Confronto codifiche (Esempio a 4bit)

In particolare entrambe le codifiche, a parità di numero di bit a disposizione, rappresentano la stessa quantità di numeri. Se viene chiamato M il numero di bit a disposizione, entrambe le codifiche possono rappresentare 2^M numeri. Come si può osservare dalla Tab. 4.1 le due codifiche si differenziano solamente per il MSB (Most Significant Bit). Infatti mentre nella codifica in Complemento a Due i numeri negativi hanno un 1 come MSB e i numeri positivi hanno uno 0; per la codifica Offset Binary vale il contrario. Quindi,

intuitivamente risulta facile il passaggio da una codifica all'altra. Sarà sufficiente negare il bit più significativo.

ANALISI TEORICA DEL SISTEMA DI ACQUISIZIONE DELLE TENSIONI

In questo capitolo si cercherà di analizzare, dal punto di vista teorico, le prestazioni del sistema di acquisizione delle tensioni. In particolare verranno esplicitati i passaggi utilizzati ai fini di ottenere l'attenuazione totale del sistema, e la sensibilità in ingresso del sistema stesso.

5.1 Sistema di acquisizione a 20Mhz

5.1.1 Attenuazione

Si immagina di avere in ingresso al sistema la massima tensione positiva pari a +300V, lo stadio di ingresso la attenua (di un fattore 1:200) portandola a +1,5V; la scheda di interfaccia introduce un'ulteriore attenuazione, (pari ad un fattore 1:1,5) in maniera da ottenere in ingresso al convertitore AD una tensione differenziale del valore di +1V (+0,5V su un canale differenziale e -0,5V sull'altro). Tale valore di tensione è il massimo positivo accettato in ingresso all'ADC, che pertanto lo converte in digitale, con il corrispondente massimo numero positivo esprimibile in Offset Binary a 14bit, che, convertito in decimale è dato dalla 5.1.1.

$$+ 2^{14-1} - 1 = +8191 \quad (5.1.1)$$

Tale numero è portato in ingresso ad uno dei due accumulatori del programma caricato nell'FPGA. Pertanto l'uscita dell'accumulatore corrisponderà al numero decimale dato dalla 5.1.2.

$$+ 8191 \cdot (20Mhz/10Khz) = +8191 \cdot 2000 = +16382000 \quad (5.1.2)$$

A questo punto il risultato dell'accumulazione viene troncato per ottenere i 14bit di uscita. Per passare da 25bit a 14bit occorre effettuare uno shift logico a destra di 11 posizioni; il che equivale a dividere il numero decimale ottenuto per $2^{11} = 2048$. Quindi si ottiene il risultato dato dalla 5.1.3.

$$\frac{+16382000}{2048} = +7999,02344 \quad (5.1.3)$$

Del quale si deve considerare il valore intero +7999, che, entrando in ingresso al convertitore DA, viene trasformato in un segnale analogico. Come visto alla fine della sottosezione 2.4.3 la sensibilità d'uscita del DAC è pari a $58,594\mu V$. Pertanto, il valore di tensione d'uscita corrispondente al numero +7999, è dato dalla 5.1.4.

$$+ 7999 \cdot 58,594\mu V + V_{OFFSET} = 468,693mV + 480mV \quad (5.1.4)$$

Poiché per il calcolo dell'attenuazione totale (rispetto ai +300V di ingresso) non bisogna considerare l'Offset; l'attenuazione introdotta dall'intero sistema è data dalla 5.1.5.

$$\frac{300V}{468,693mV} = 640,078 \quad (5.1.5)$$

Si fa notare che nel calcolo dell'attenuazione totale del sistema, le conversioni da OB a C2 e viceversa presenti nel programma, risultino del tutto ininfluenti, e quindi non sono state prese in considerazione nei passaggi affrontati.

5.1.2 Sensibilità

Il numero N di campioni digitalizzati dal convertitore AD in ogni periodo PWM è dato dalla 5.1.6.

$$N = \frac{20Mhz}{10Khz} = 2000 \text{ campioni per periodo PWM} \quad (5.1.6)$$

Come visto anche sopra, il numero T equivalente alla divisione decimale, corrispondente al troncamento dei dati in uscita dall'accumulatore, è dato dalla 5.1.7.

$$T = 2^{11} = 2048 \quad (5.1.7)$$

Come precedentemente accennato in sottosezione 3.4.1, poiché il numero T risulta strettamente maggiore del numero N , si avrà una diminuzione della sensibilità del sistema.

Verranno di seguito riportati i passaggi effettuati a ritroso, a partire dal dato digitale di uscita del programma, ai fini di calcolare la sensibilità del sistema stesso.

Minimo numero positivo in uscita dopo il troncamento da 25bit a 14bit e trasformazione da C2 a OB:

OB: 1 0 0 0 0 0 0 0 0 0 0 0 1

prima del troncamento:

OB: 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0

prima della trasformazione da C2 a OB:

C2: 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0

conversione decimale del numero:

DEC: 2048 (= 2^{11})

Questo è il numero salvato nel registro di uscita dell'accumulatore, poiché esso in un periodo effettua 2000 sommatorie, allora, il numero più piccolo in ingresso all'accumulatore affinché io veda in uscita un 1 (decimale), è dato dalla 5.1.8.

$$\frac{T}{N} = 2048/2000 = 1,024 \quad (5.1.8)$$

Ovviamente il convertitore AD trasforma segnali analogici in valori discreti, quindi non può convertire un dato di ingresso con un valore di uscita pari a 1,024. Bensì, può produrre un 1, che se accumulato 2000 volte, realizzerebbe un zero in uscita. Oppure potrebbe produrre un 2, che, se accumulato 2000 volte realizzerebbe un'uscita positiva. Ciò potrebbe far pensare che il 2 sia il numero istantaneo minimo in ingresso che fa sì che il sistema produca un uscita positiva. Ragionando in questo modo si fa un errore di valutazione perché non si ragiona sul valore medio in ingresso.

In realtà il minimo numero totale accumulato che fa scattare un uno in uscita è il 2048, che, come dato dalla 5.1.8, corrisponde al numero di ingresso 1,024. Ma questo numero

rappresenta il valore medio, che se entrasse nell'accumulatore, per 2000 accumulazioni, produrrebbe il numero 2048, visibile come un uno in uscita dopo il troncamento. Quindi nell'esempio appena proposto il numero 2048 viene prodotto da un'accumulazione di 2000 campioni di valore 1,024 non rappresentabili dall'ADC. Ma tale risultato può essere raggiunto da una sommatoria di 2000 campioni interi (con valore medio 1,024) rappresentabili dall'ADC. Quindi il numero corretto da prendere in considerazione nei seguenti passaggi è proprio 1,024.

Poiché il numero massimo di uscita dall'ADC, convertito in decimale, è dato dalla 5.1.1. Ed essendo tale valore corrispondente ad una tensione di ingresso pari ad +1V. La risoluzione di ingresso dell'ADC è data dalla 5.1.9.

$$\frac{+1V}{+8191} = 122,085\mu V \quad (5.1.9)$$

Pertanto la sensibilità in ingresso all'FPGA è data dalla 5.1.10

$$122,085\mu V \cdot 1,024 = 125,015\mu V \quad (5.1.10)$$

Ciò significa che il programma caricato nell'FPGA è sensibile a tensioni medie in ingresso pari a 125,015 μ V.

Infine ricordando che, prima di giungere al convertitore AD, il segnale subisce due attenuazioni pari a 1:1,5 e 1:200.

L'equazione 5.1.11 fornisce la sensibilità del sistema completo.

$$125,015\mu V \cdot 1,5 \cdot 200 = 37,505mV \quad (5.1.11)$$

Ricapitolando, affinché in uscita dall'accumulatore sia presente un 1; in ingresso al sistema deve esserci un segnale con valore medio sul periodo (100 μ s) maggiore o uguale a 37,505 mV.

5.2 Sistema di acquisizione a 1,28Mhz

Oltre al programma caratterizzato da una frequenza di campionamento del convertitore AD pari a 20Mhz, è stata successivamente realizzata un'altra versione con campionamento pari a 1,28Mhz. La frequenza di campionamento è stata scelta in maniera da ottenere l'uguaglianza $T = N$, in modo che il programma caricato nell'FPGA fornisca, in uscita, effettivamente il valore medio dei dati in ingresso all'FPGA e non un valore proporzionale. Di seguito viene approfondito tale passaggio.

Il numero N di campioni digitalizzati dal convertitore AD in ogni periodo PWM è dato dalla 5.2.1.

$$N = \frac{1,28Mhz}{10Khz} = 128 \text{ campioni per periodo PWM} \quad (5.2.1)$$

L'equazione 5.1.1 fornisce il massimo numero rappresentabile in uscita dall'ADC (convertito in decimale). Se per un intero periodo PWM tale valore entrasse in un accumulatore, il registro di uscita dell'accumulatore stesso dovrebbe contenere il numero decimale dato dalla 5.2.2.

$$+8191 \cdot 128 = 1048448 = \text{max } N^{\circ} \text{accumulabile} \quad (5.2.2)$$

Tale numero risulta rappresentabile con 21bit, infatti si ottiene: $2^{21-1} - 1 = 1048575$ che risulta maggiore del numero ricavato dalla 5.2.2. Pertanto il registro di uscita di ciascun accumulatore deve contenere 21bit.

Poiché l'ingresso del convertitore DAC accetta segnali a 14bit, deve essere effettuata un'operazione di troncamento dei 7 bit meno significativi dell'uscita degli accumulatori. Con questo dimensionamento il numero T , equivalente alla divisione decimale, corrispondente al troncamento dei dati in uscita dall'accumulatore, è dato dalla 5.2.3.

$$T = 2^7 = 128 \quad (5.2.3)$$

Che, come ci si aspettava, corrisponde al numero N di campioni processati dal programma in ciascun periodo PWM (vedi 5.2.1). Vedremo come tale accortezza produca un leggero migliramento della sensibilità del sistema.

5.2.1 Attenuazione

Come per il calcolo dell'attenuazione del sistema a 20Mhz si immagina di avere in ingresso al sistema la massima tensione positiva pari a +300V, nella 5.2.4 si schematizzano i passaggi, già ampiamente spiegati:

$$(+300V) \rightarrow (+1,5V) \rightarrow (+1V) \rightarrow (+8191) \rightarrow (+8191 \cdot 128) = +1048448 \quad (5.2.4)$$

A questo punto il risultato dell'accumulazione viene troncato per ottenere i 14bit di uscita. Per passare da 21bit a 14bit occorre effettuare uno shift logico a destra di 7 posizioni; il che equivale a dividere il numero decimale ottenuto per $2^7 = 128$. Quindi si ottiene il risultato dato dalla 5.2.5.

$$\frac{+1048448}{128} = +8191 \quad (5.2.5)$$

Il numero +8191 entrando in ingresso al convertitore DA, viene trasformato in un segnale analogico. Come riportato prima, la sensibilità d'uscita del DAC è pari a $58,594\mu V$. Pertanto, il valore di tensione d'uscita corrispondente al numero +8191, è dato dalla 5.2.6.

$$8191 \cdot 58,594\mu V + V_{OFFSET} = 479,943mV + 480mV \quad (5.2.6)$$

Poiché per il calcolo dell'attenuazione totale (rispetto ai +300V di ingresso) non bisogna considerare l'Offset; l'attenuazione introdotta dall'intero sistema è data dalla 5.2.7.

$$\frac{300V}{479,943mV} = 625,074 \quad (5.2.7)$$

L'attenuazione risulta inferiore rispetto a quella introdotta dal sistema con campionamento a 20Mhz. Il che risulta corretto, in quanto, mentre nel sistema con campionamento a 1,28Mhz il dato d'uscita decimale è +8191, nel sistema con campionamento a 20Mhz, il dato decimale d'uscita è +7999. Tale risultato viene convertito in un valore di tensione inferiore rispetto alla conversione analogica del numero +8191. Pertanto il sistema con campionamento a 1,28Mhz è caratterizzato da un'attenuazione inferiore rispetto a quello con campionamento a 20Mhz.

5.2.2 Sensibilità

Il numero N di campioni digitalizzati dal convertitore AD in ogni periodo PWM è dato dalla 5.2.1.

Il numero T , equivalente alla divisione decimale, corrispondente al troncamento dei dati in uscita dall'accumulatore, è dato dalla 5.2.3. Pertanto, come già fatto notare, N e T sono entrambi uguali a 128.

Come effettuato per la versione del programma a 20Mhz, di seguito vengono riportati i passaggi effettuati a ritroso, a partire dal dato digitale di uscita del programma, ai fini

di calcolare la sensibilità del sistema stesso.

Minimo numero positivo in uscita dopo il troncamento da 21bit a 14bit e trasformazione da C2 a OB:

OB: 1 0 0 0 0 0 0 0 0 0 0 0 0 1

prima del troncamento:

OB: 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0

prima della trasformazione da C2 a OB:

C2: 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0

conversione decimale del numero:

DEC: 128 (= 2^7)

Questo è il numero salvato nel registro di uscita dell'accumulatore, poiché esso in un periodo effettua 128 sommatorie, allora, il numero più piccolo in ingresso all'accumulatore affinché io veda in uscita un 1 (decimale), è dato dalla 5.2.8.

$$\frac{T}{N} = \frac{128}{128} = 1 \quad (5.2.8)$$

La risoluzione di ingresso dell'ADC è data dalla 5.1.9. Per calcolare la sensibilità in ingresso all'FPGA dovrei moltiplicare pertanto il valore $122,085 \mu\text{V}$ per il rapporto T/N . Essendo tale rapporto uguale a uno, la risoluzione dell'ADC coincide con la sensibilità di ingresso dell'FPGA. Quindi il programma caricato nell'FPGA è sensibile a tensioni medie di ingresso pari a $122,085 \mu\text{V}$.

Infine ricordando che, prima di giungere al convertitore AD, il segnale subisce due attenuazioni pari a 1:1,5 e 1:200. L'equazione 5.2.9 fornisce la sensibilità del sistema completo.

$$122,085\mu\text{V} \cdot 1,5 \cdot 200 = 36,626\text{mV} \quad (5.2.9)$$

Ricapitolando, affinché in uscita dall'accumulatore sia presente un 1; in ingresso al sistema deve esserci un segnale con valore medio sul periodo ($100\mu\text{s}$) maggiore o uguale a $36,626 \text{ mV}$.

Il miglioramento ottenuto (rispetto al sistema con campionamento a 20Mhz) non è molto significativo a livello di performance. Ma utile per capire che la sensibilità del sistema di ingresso risulta inversamente proporzionale al rapporto T/N . Se tale rapporto risulta essere uguale a uno, la sensibilità del sistema dipende soltanto dalla risoluzione di ingresso del convertitore AD (e ovviamente dagli stadi di riduzione). All'aumentare di tale rapporto la risoluzione progressivamente cala, a causa della penalizzante operazione di troncamento.

RISULTATI SPERIMENTALI

In questo capitolo verranno riportati i risultati ottenuti dalle misurazioni sperimentali effettuate sul sistema di misura delle tensioni dell'inverter. Verranno illustrate le differenze calcolate tra le misurazioni reali e i valori teorici. Verranno inoltre confrontati il sistema con campionamento a 20Mhz e il sistema con campionamento a 1,28Mhz.

6.1 Misure in continua

Per prima cosa sono state misurate le prestazioni del sistema, ponendo in ingresso una tensione continua all'interno del range $\pm 300V$. In particolare sono stati misurati i valori di uscita del sistema, fornendo delle tensioni di ingresso all'interno del range con intervalli di 10V.

Dalle misurazioni così ottenute, è stato ricavato il valore medio di attenuazione totale, e successivamente la sua deviazione standard. Poi si è passati ad analizzare la linearità del sistema. Con Excel è stato ricavato il coefficiente angolare del sistema linearizzato, come riportato nella Fig. 6.1 (sistema con campionamento a 20Mhz) e nella Fig. 6.2 (sistema con campionamento a 1,28Mhz). Grazie al quale è stata calcolata l'entità dello scostamento del sistema reale da quello linearizzato. In particolare sono stati ricavati l'errore massimo e l'errore medio percentuali.

6.1.1 Sistema con campionamento a 20Mhz

Il valore medio dell'attenuazione totale e la deviazione standard, del sistema con campionamento a 20Mhz, sono riportati rispettivamente nella 6.1.1, e nella 6.1.2.

$$ATTENUAZIONE MEDIA = 635,205 \quad (6.1.1)$$

Tale valore di attenuazione si discosta di 0,76% dai calcoli teorici visti in sottosezione 5.1.1.

$$DEVIAZIONE STANDARD = 2,164V \quad (6.1.2)$$

Il coefficiente angolare del sistema linearizzato, ottenuto dal grafico in Fig. 6.1, risulta uguale a 0,0015734693 (precisione fino alla decima cifra decimale).

Dal quale (attraverso una semplice moltiplicazione) sono stati ricavati tutti i valori di uscita linearizzati (a partire dalle tensioni di ingresso continue misurate).

Infine dalla 6.1.3 sono stati ricavati tutti gli errori percentuali tra le uscite misurate e le uscite linearizzate.

$$E_{\%} = \frac{OUT_{misurata} - OUT_{linearizzata}}{OUT_{misurata}} \cdot 100 \quad (6.1.3)$$

In modo da ottenere l'errore massimo e l'errore medio percentuali riportati rispettivamente nelle 6.1.4 e 6.1.5.

$$E_{max} \% = 1,109\% \quad (6.1.4)$$

$$E_{mean} \% = 0,053\% \quad (6.1.5)$$

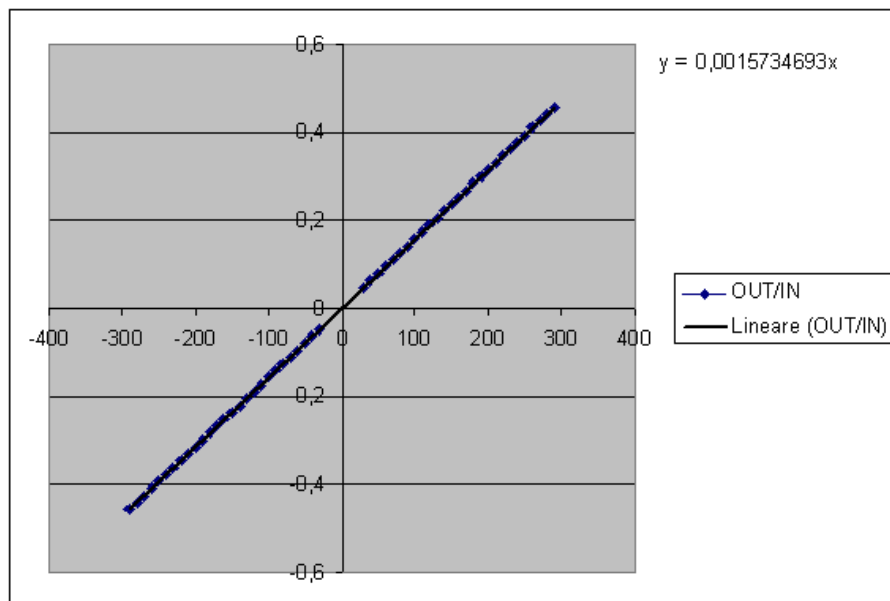


Fig. 6.1: Linearizzazione del sistema con campionamento a 20Mhz

6.1.2 Sistema con campionamento a 1,28Mhz

Il valore medio dell'attenuazione totale e la deviazione standard, del sistema con campionamento a 1,28Mhz, sono riportati rispettivamente nella 6.1.6, e nella 6.1.7.

$$ATTENUAZIONE MEDIA = 616,133 \quad (6.1.6)$$

Tale valore di attenuazione si discosta di 1,45% dai calcoli teorici visti in sottosezione 5.1.1.

$$DEVIAZIONE STANDARD = 0,471V \quad (6.1.7)$$

Il coefficiente angolare del sistema linearizzato, ottenuto dal grafico in Fig. 6.2, risulta uguale a 0,0016224730 (precisione fino alla decima cifra decimale).

Grazie al quale sono stati ricavati l'errore massimo e l'errore medio percentuali (tra i valori misurati e i valori linearizzati) riportati rispettivamente nelle 6.1.8 e 6.1.9.

$$E_{max} \% = 1,170\% \quad (6.1.8)$$

$$E_{mean} \% = 0,034\% \quad (6.1.9)$$

6.1.3 Confronto tra i due sistemi

Dai risultati ottenuti si può notare come le prestazioni dei due sistemi siano confrontabili. In particolare gli errori massimi e medi risultano molto vicini tra loro.

Il sistema con campionamento a 1,28Mhz ha una deviazione standard pari a 0,471V che risulta migliore di quella pari a 2,164V, calcolata per il sistema con campionamento a 20Mhz.

Infine il sistema con campionamento a 1,28Mhz è dotato della migliore sensibilità possibile, dipendente soltanto dalla risoluzione del convertitore AD e dagli stadi di attenuazione di ingresso.

Per tanto si può concludere che il sistema con campionamento a 1,28Mhz risulta maggiormente ottimizzato rispetto a quello con campionamento a 20Mhz.

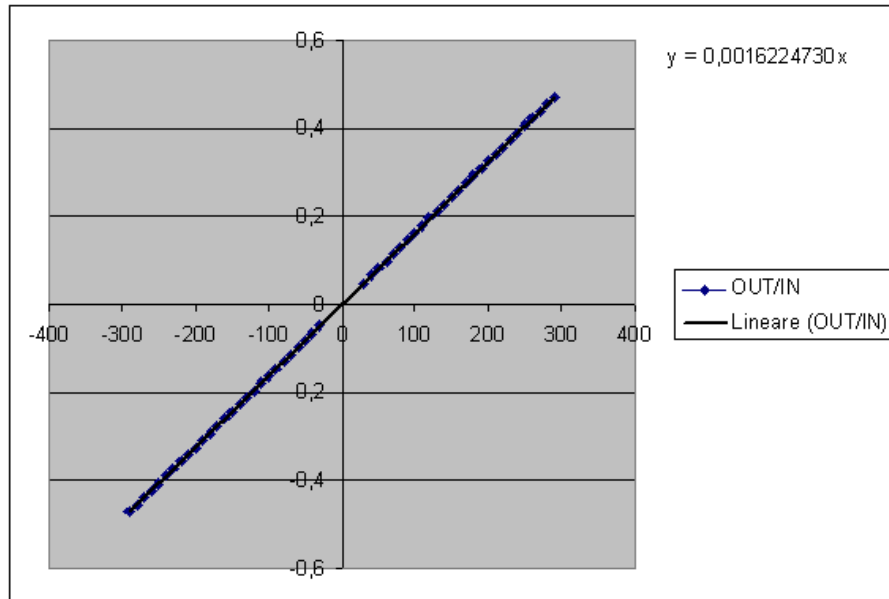


Fig. 6.2: Linearizzazione del sistema con campionamento a 1,28Mhz

6.2 Misurazioni sperimentali delle tensioni in inverter trifase

Nelle figure Fig. 6.3 e Fig. 6.4 si riportano i risultati sperimentali ottenuti, relativi ad una sola tensione concatenata del motore. Le figure sono rispettivamente ricavate utilizzando il sistema con campionamento a 20Mhz e il sistema con campionamento a 1,28Mhz.

In particolare un motore sincrono a magneti permanenti, è stato alimentato con una tensione di fase di ampiezza 70V, e una frequenza di 50Hz (tramite un controllo V/Hz). La frequenza di modulazione PWM utilizzata è pari a 10Khz. Il segnale di colore blu rappresenta la tensione di alimentazione concatenata fornita al motore, mentre quello di colore nero rappresenta la tensione media ottenuta dall'elaborazione del sistema di acquisizione, infine quello di colore rosso rappresenta la tensione media ottenuta con Matlab.

In entrambe le figure si può osservare come il risultato dell'elaborazione del sistema di acquisizione delle tensioni risulta prossimo al calcolo ideale ottenuto mediante l'algoritmo in Matlab.

Negli ingrandimenti è stato catturato un punto caratterizzato da una non linearità causata dall'introduzione dei tempi morti negli interruttori dell'inverter. Nelle 6.2.1 e 6.2.2 vengono riportati gli errori medi percentuali tra il calcolo effettuato con matlab e le misurazioni sperimentali.

$$ERRORE\%_{(20Mhz)} = 0,18\% \quad (6.2.1)$$

$$ERRORE\%_{(1,28Mhz)} = 0,71\% \quad (6.2.2)$$

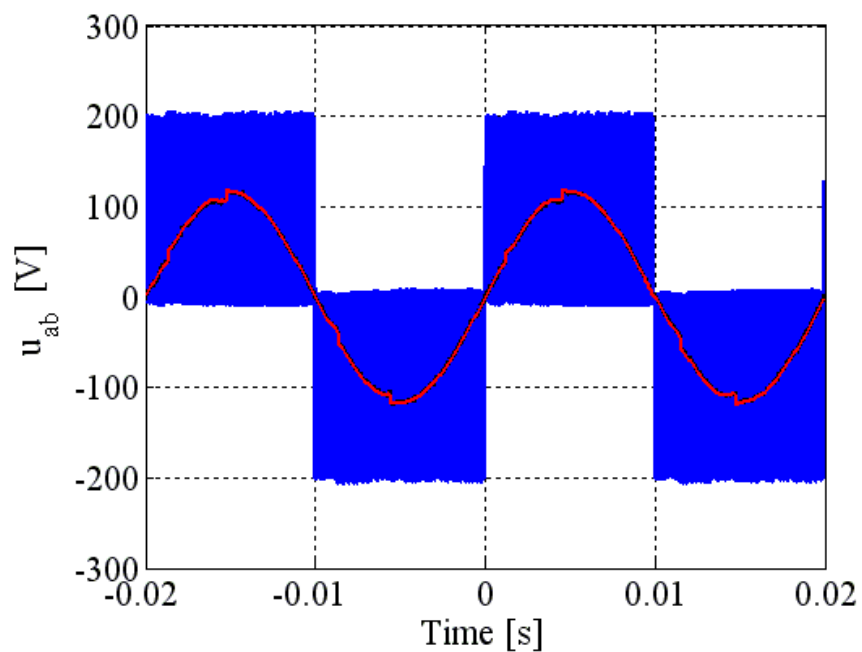


Fig. 6.3: Tensione Media Concatenata (Campionamento a 20Mhz)

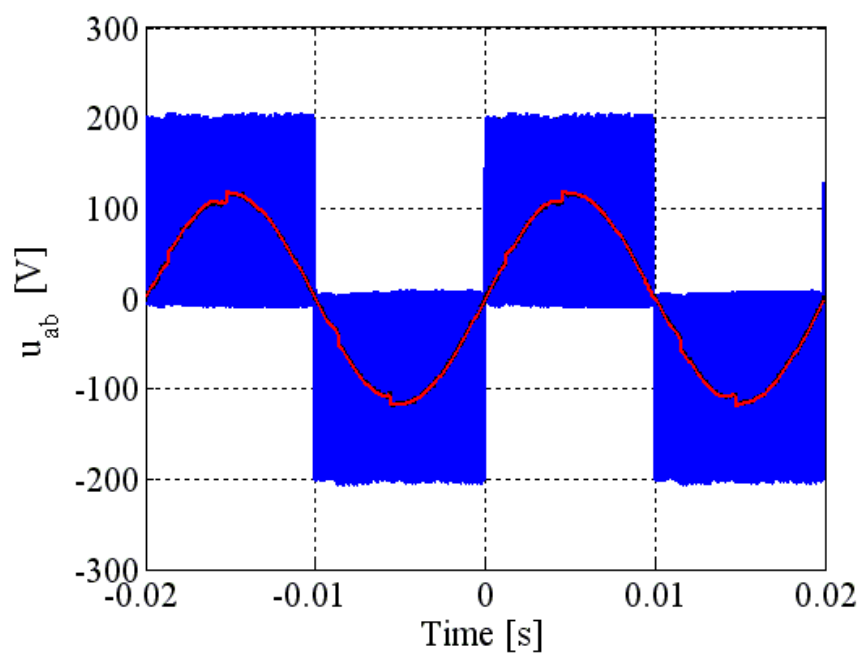


Fig. 6.4: Tensione Media Concatenata (Campionamento a 1,28Mhz)

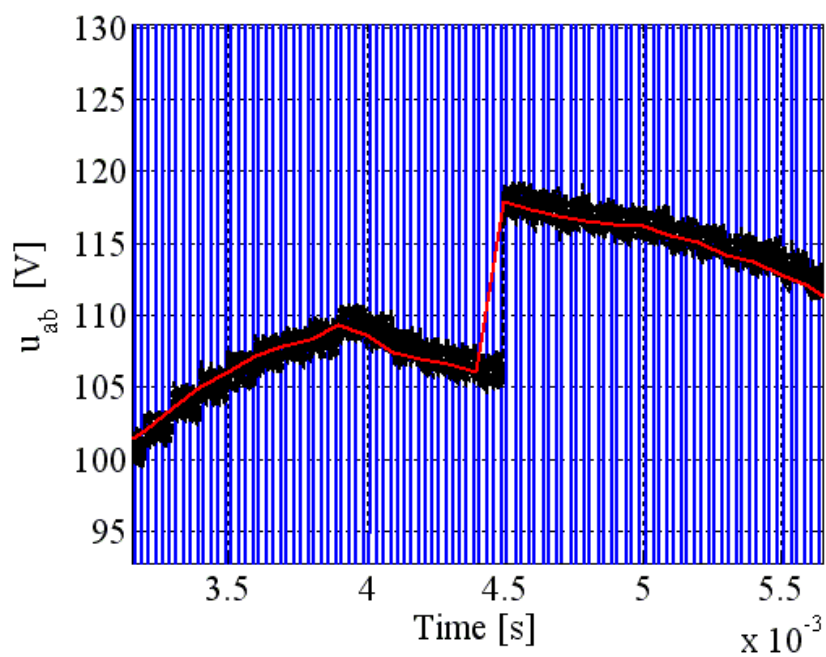


Fig. 6.5: Tensione Media Concatenata (Campionamento a 20Mhz) (Ingrandimento)

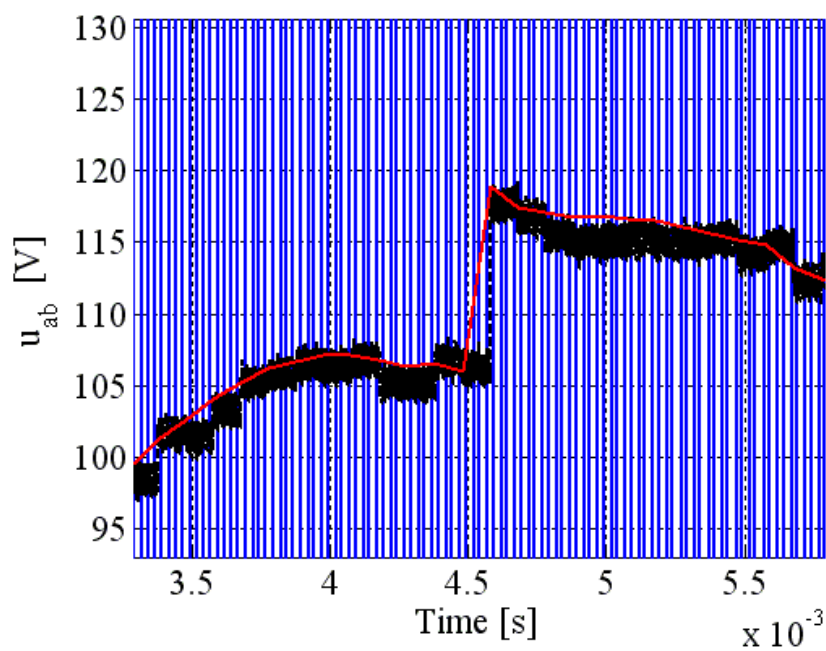


Fig. 6.6: Tensione Media Concatenata (Campionamento a 1,28Mhz) (Ingrandimento)

Conclusioni

L'obiettivo principale dell'intero progetto può ritenersi raggiunto: è stato possibile calcolare le componenti fondamentali delle tensioni concatenate di un motore in corrente alternata attraverso l'utilizzo di un dispositivo logico digitale (FPGA).

Ovviamente ci sono moltissime altre possibilità di sviluppare il lavoro svolto e di migliorare quello che è stato compiuto, approfondendo tutti quegli aspetti che fino a questo punto non sono stati presi in considerazione in maniera approfondita. In primo luogo andrebbe realizzato un software per l'FPGA più flessibile, ovvero in grado di elaborare i dati anche a fronte di variazioni della frequenza di modulazione PWM; difatti, diminuendo tale frequenza andremmo incontro a fenomeni di overflow dovuti al fatto che gli accumulatori accumulerebbero un numero maggiore di dati, mentre aumentandola i problemi andrebbero ad incidere sulla sensibilità in quanto il numero accumulato sarebbe piccolo ed il troncamento risulterebbe più pesante. L'algoritmo da noi sviluppato lavorava con una frequenza PWM pari a 10KHz costanti.

Un secondo aspetto riguarda la realizzazione di un'unica scheda contenente l'hardware di acquisizione e conversione delle tensioni, in particolare ci si dovrebbe concentrare sulla scelta del convertitore. Inizialmente si è pensato di lavorare con un campionamento a 20Mhz per verificare l'effettiva realizzabilità del sistema. In questo modo è stata sfruttata l'elevata velocità di campionamento dei convertitori AD. In una fase successiva si è passati a lavorare ad una frequenza molto più bassa pari a 1,28Mhz, con risultati sicuramente paragonabili a quelli ottenuti con un campionamento a frequenza più elevata, se non migliori, il che comporterebbe l'utilizzo di un ADC dal costo inferiore. La scelta di lavorare a frequenze di campionamento più basse, permette quindi di ridurre i costi mantenendo le prestazioni del sistema. Infine il programma sfrutta una minima parte delle risorse hardware della scheda FPGA, il che permetterebbe di utilizzare una FPGA meno evoluta e quindi, più economica.

Ringraziamenti

Ringrazio tutti coloro che mi hanno sostenuto, sia direttamente sia indirettamente, nella realizzazione di questa tesi.

Bibliografia

- [1] S.Bolognani, “Azionamento IPM e sensorless con motore BL,” in *Dispense Del Corso Di ‘Azionamenti Elettrici’*, Jun. 2008.
- [2] —, “Invertitore,” in *Dispense Del Corso Di ‘Azionamenti Elettrici’*, Feb. 2008.
- [3] Altera, “Reference manual cyclone_3_fpga_starter_board,” http://www.altera.com/literature/manual/rm_ciii_starter_board.pdf.
- [4] A. Devices, “Datasheet ad9248,” http://www.analog.com/static/imported-files/data_sheets/AD9248.pdf.
- [5] —, “Datasheet ad9767,” http://www.analog.com/static/imported-files/data_sheets/AD9763_9765_9767.pdf.
- [6] Altera, “Quartus2 download page,” <https://www.altera.com/download/software/quartus-ii-we>.