

UNIVERSITÀ DEGLI STUDI DI PADOVA

FACOLTÀ DI INGEGNERIA



Corso di Laurea in
Ingegneria Elettronica

REALIZZAZIONE IN AMBIENTE LABVIEW
FPGA DI UN CONTROLLORE DI CORRENTE
PER INVERTER

RELATORE
Prof. Buso Simone

STUDENTE
Pravato Francesca

Anno Accademico 2012/2013

Indice

Sommario	ii
Introduzione	1
1 Modulazione PWM	3
1.1 Modulazione PWM analogica	4
1.2 Modulazione PWM digitale	7
1.2.1 Counter-based	7
1.2.1.1 Tipologie di counter-based	8
1.2.1.2 Segnale di sincronismo e campionamento del segnale	10
1.2.1.3 Frequenza di clock e risoluzione	11
1.2.2 Delay-line	12
1.2.3 Hybrid	15
1.2.4 Nuove tipologie di modulazione digitale	17
1.2.4.1 SCI DPWM	18
1.2.4.2 Limited-Pool Random Carrier-Frequency PWM .	20
1.3 Confronto tra modulazione PWM digitale ed analogica	23
2 Microcontrollori	27
2.1 Microcontrollore PIC	28
2.1.1 MCPWM (motor control PWM)	29
2.2 Microcontrollori RENESAS	31
3 CompactRIO	35
3.1 Struttura del CompactRIO	36
3.2 NI cRIO-9082	38
3.2.1 Moduli I/O.	39
3.3 FPGA	42
3.4 CompactRIO Scan Mode	44
3.5 Architettura di controllo	46
3.5.1 LabVIEW FPGA	47
3.5.2 LabVIEW Real-Time	50

3.5.3	LabVIEW per Windows	51
3.6	Sincronizzazione	52
4	GPIC	55
5	Realizzazione della PWM digitale in LabVIEW	59
5.1	PWM digitale con contatore leading-edge	59
5.1.1	Descrizione del programma	60
5.1.2	Test del programma	62
5.2	Modulazione PWM double-edge	63
5.2.1	Primo modulatore double-edge	63
5.2.1.1	Descrizione del programma	64
5.2.2	Secondo modulatore double-edge	65
5.2.2.1	Descrizione del programma	65
5.2.2.2	Test del programma	66
5.2.3	Test sui contatori double-edge	67
5.3	Test sulla velocità dei contatori al variare del numero di conteggi .	70
5.4	Test generali	71
6	Realizzazione del regolatore PI	73
6.1	Dimensionamento del K_I e del K_P	74
6.2	Tecnica anti-wind-up	77
6.3	Realizzazione del subVI del regolatore PI	78
7	Realizzazione del controllo di corrente	83
7.1	Realizzazione del sistema di acquisizione	84
7.2	Realizzazione del sistema di controllo	86
8	Emulatore di inverter	87
8.1	Funzionamento inverter	87
8.2	Emulatore	92
9	Risultati sperimentali	97
9.1	Risultati con riferimento fisso	98
9.2	Risultati con riferimento la tensione di rete	101
10	Conclusioni	107
	Ringraziamenti	109
	Elenco delle figure	113

Sommario

La tesi contiene le nozioni utili per realizzare un efficiente controllo di corrente e riporta i risultati ottenuti nell'applicazione di tale controllo ad un emulatore di inverter. Tale emulatore simula il comportamento di un inverter monofase dimensionato per una potenza nominale di 3kW connesso alla rete.

I dati sono stati acquisiti e controllati tramite l'utilizzo di strumenti "National Instruments" quali il "CompactRIO" e la NI Single-Board RIO "GPIC". Il controllo di corrente è stato realizzato in ambiente di programmazione LabVIEW FPGA.

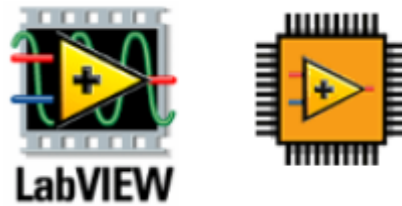


Figura 1: LabVIEW & LabVIEW FPGA

Introduzione

Con questa tesi si è realizzato un controllo di corrente per inverter, utilizzando come sistema di controllo alcuni strumenti della National Instruments, quali il "CompactRIO" e la "GPIC".



Figura 2: CompactRIO & GPIC

Il lavoro svolto in questi mesi si può suddividere principalmente in tre parti.

- La prima di ricerca bibliografica relativa alla modulazione PWM, incentrata particolarmente sulla digitale.
- La seconda di programmazione LabVIEW, indirizzata principalmente a verificare le effettive potenzialità e limiti del "CompactRIO".
- La terza, di programmazione LabVIEW, consiste nella realizzazione del controllo di corrente di inverter. Tale controllo è stato applicato ad un simulatore di inverter grazie all'utilizzo della NI Single-Board RIO "GPIC".

Di seguito sono elencati gli argomenti trattati nei vari capitoli che compongono questa tesi.

Il primo capitolo tratta la modulazione PWM. Nello specifico si è spiegato il funzionamento sia della modulazione PWM analogica, sia della modulazione PWM digitale. Dopo un rapido confronto sui relativi vantaggi e svantaggi dell'analogico e del digitale, vengono approfondite le varie tipologie esistenti di modulazione PWM digitale. Tale approfondimento ha lo scopo di facilitare la scelta

di quale tipologia implementare in vista del controllo di corrente dell'emulatore d'inverter.

Il secondo capitolo propone una visione generale sui microcontrollori, col fine di portare esempi reali di controllo digitale PWM. Si prendono in considerazione i microcontrollori PIC della Microchip technology ed i microcontrollori Renesas.

Il terzo capitolo è mirato alla descrizione dello strumento "CompactRIO". Il primo approccio pratico con tale strumento è stato a livello di test. Lo scopo primario, infatti, è stato quello di verificare le sue reali potenzialità, implementando dei semplici programmi ed analizzandone la tempistica.

Il quarto capitolo riassume brevemente le caratteristiche della NI Single-Board RIO "GPIC".

Il quinto capitolo tratta la realizzazione della PWM digitale tramite programmazione LabVIEW FPGA. I modulatori sono di tipo single-edge e double-edge.

Il sesto capitolo tratta la realizzazione del regolatore PI con linguaggio LabVIEW. In esso viene inoltre descritto il metodo con cui sono stati ricavati i parametri K_I e K_P e la tecnica anti-wind-up.

Il settimo capitolo descrive il funzionamento del controllo di corrente e quindi l'implementazione LabVIEW.

L'ottavo capitolo spiega il funzionamento di un generico inverter e descrive le caratteristiche dell'emulatore utilizzato nei test.

Il nono capitolo mostra i risultati ottenuti effettuando il controllo di corrente dell'emulatore di inverter utilizzando come acquirente/elaboratore dati la "GPIC".

Il decimo capitolo conclude la tesi esponendone i possibili utilizzi.

Capitolo 1

Modulazione PWM

In un convertitore, al fine di garantire una regolazione efficace della tensione di uscita al variare di:

- tensione di ingresso;
- parametri del circuito;
- corrente di carico;

risulta necessario un controllo a retroazione della tensione di uscita stessa. Tale controllo può venir effettuato grazie alla generazione di un appropriato segnale di comando che regoli la durata della chiusura e dell'apertura dell'interruttore. Tale tecnica va sotto il nome di modulazione PWM.

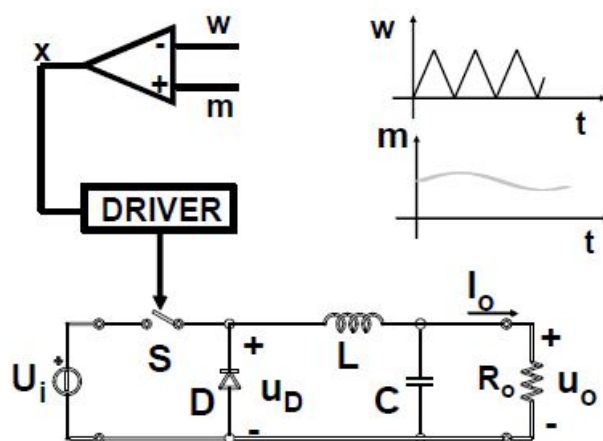


Figura 1.1: Controllo di un convertitore buck con modulazione PWM analogica [11]

La dicitura PWM corrisponde all'acronimo di Pulse Width Modulation e si tratta quindi di una tecnica di modulazione della larghezza degli impulsi.

Tale tipologia di modulazione si basa sulla comparazione di due segnali, uno detto modulante e l'altro portante. Il segnale in uscita dal comparatore sarà di valore logico alto, se il segnale modulante è maggiore del segnale portante. Viceversa nel caso contrario.

Le principali applicazioni della PWM si riscontrano nei convertitori di potenza, amplificatori audio e nel controllo dei motori elettrici.

Prima di trattare nel dettaglio la modulazione PWM digitale, che verrà utilizzata per il controllo dell'inverter, è di seguito brevemente illustrata la modulazione PWM analogica.

1.1 Modulazione PWM analogica

La modulazione PWM analogica utilizza come segnale portante un segnale triangolare e come segnale modulante un segnale di altro tipo. Come si vede in figura 1.2 [11], comparando il segnale portante ed il segnale modulante si ottiene il segnale logico di comando (x). Il segnale in uscita ricavato dalla modulazione PWM, avrà un valore medio proporzionale al segnale modulante. Nel caso in cui il segnale modulante sia di valore costante, si ottiene in uscita un segnale a onda quadra le cui caratteristiche rimangono costanti nel tempo, come illustrato in figura 1.3 [10].

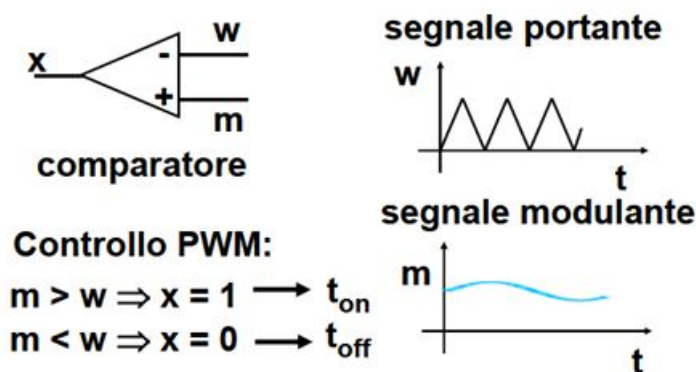


Figura 1.2: Principio di funzionamento della modulazione analogica [11]

Per capire come funziona il PWM, si prenda in considerazione la figura 1.3, dove è presente un segnale ad onda quadra con periodo fisso "T" e duty-cycle "d" variabile.

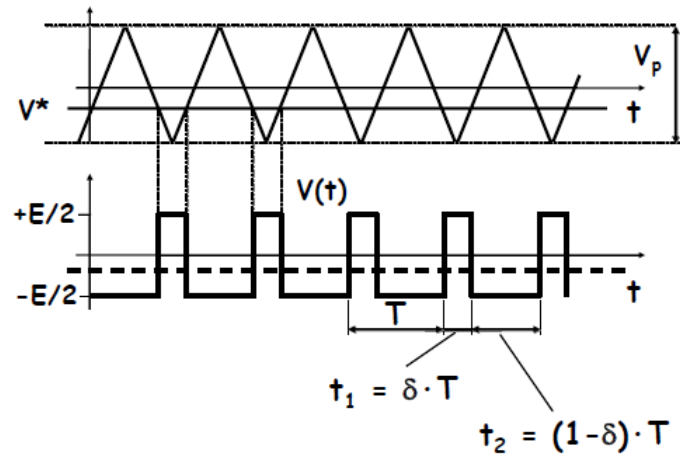


Figura 1.3: Modulazione analogica con modulante costante [10]

Si definisce d (duty-cycle) il valore generato proporzionale al segnale modulante, il cui valore è dato dal rapporto tra il tempo in cui l'onda quadra assume valore logico alto (t_1) ed il periodo T (periodo di modulazione) :

$$d = \frac{t_1}{T} = \frac{1}{2} + \frac{V^*}{V_p} \quad (1.1)$$

Nell'equazione 1.1 vengono presi in considerazione i valori V_p (l'ampiezza del segnale portante) e V^* (l'ampiezza picco-picco del segnale modulante) che si suppone, per una prima analisi, costante.

In figura 1.4 si vede che la tensione media V_{med} è compresa tra $-\frac{E}{2}$ e $\frac{E}{2}$. La relazione che lega il valore della tensione media prodotta con il duty-cycle (compreso tra 0 ed 1) è:

$$V_{med} = \left(d - \frac{1}{2}\right) \cdot E = E \cdot \frac{V^*}{V_p} \quad (1.2)$$

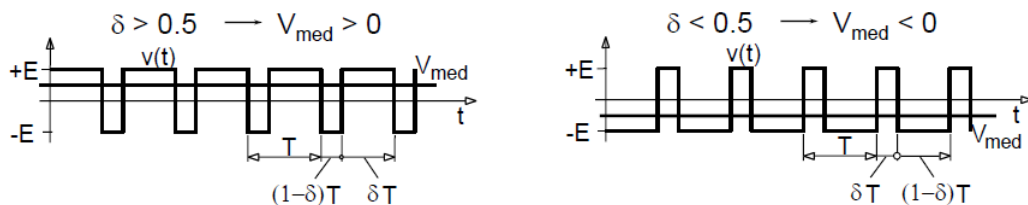


Figura 1.4: Andamento della modulante e della tensione media [11]

Nel caso in cui il segnale modulante $V(t)^*$ non sia costante, ma sia lentamente variabile nel tempo rispetto al periodo T , si trova che la tensione media $V_{med}(t)$, risulta pari a:

$$V_{med}(t) = (d(t) - \frac{1}{2}) \cdot E = E \cdot \frac{V(t)^*}{V_p} \quad (1.3)$$

Quindi l'andamento della tensione media segue quello della modulante come si vede in figura 1.5 [10].

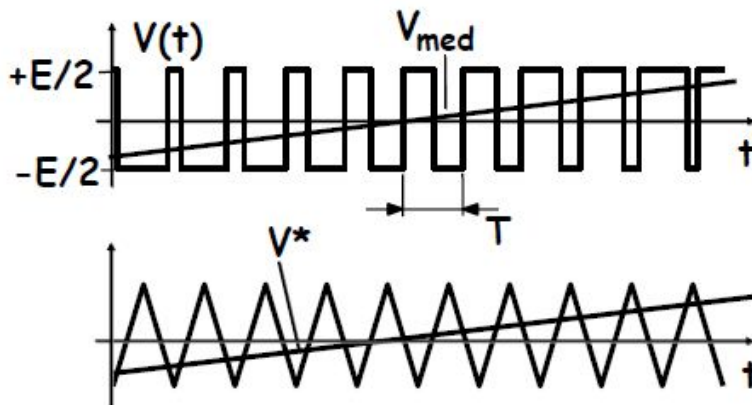


Figura 1.5: Andamento della modulante e della tensione media [10]

La modulazione PWM è un'operazione non lineare che provoca una distorsione del segnale. Tale distorsione possiede una caratteristica distribuzione in frequenza, che permette la ricostruzione dell'informazione associata al segnale modulante. Questa operazione, definita demodulazione, consiste nel filtraggio delle componenti armoniche della frequenza portante. In figura 1.6 si nota che il segnale demodulato presenta un ripple dovuto al contenuto armonico residuo.

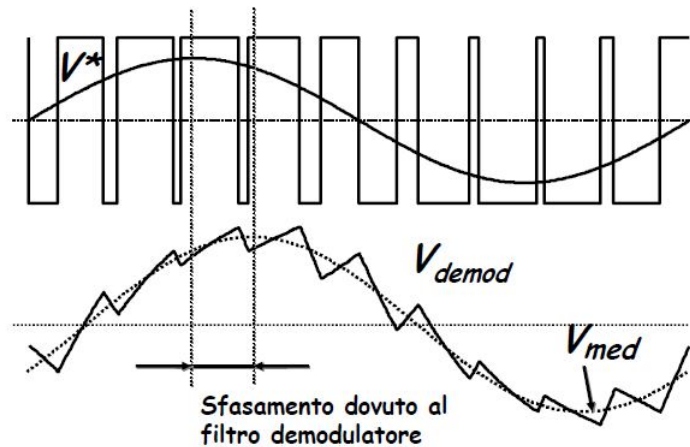


Figura 1.6: Segnale demodolato [10]

1.2 Modulazione PWM digitale

La PWM digitale, contrariamente alla analogica, utilizza un contatore binario al posto del segnale portante analogico triangolare e sostituisce al comparatore un circuito di "compare". Tra le varie tipologie esistenti di modulazione PWM digitale, si prendono in considerazione le seguenti:

1. counter-based;
2. delay-line;
3. hybrid.

Esse si distinguono l'una dall'altra per la differente realizzazione del circuito di conteggio. La counter-based utilizza un semplice contatore, la delay-line una linea di ritardo composta da celle e la hybrid implementa sia il contatore sia la linea di ritardo.

1.2.1 Counter-based

La modulazione counter-based, come precedentemente introdotto, si basa sull'utilizzo di un contatore binario che incrementa il conteggio ad ogni ciclo di clock. Il valore assunto da tale contatore viene costantemente confrontato con quello del duty-cycle grazie all'utilizzo del circuito di comparazione. Il risultato di tale paragone dà origine alla generazione dell'impulso la cui durata corrisponde al periodo in cui il valore del duty-cycle è maggiore di quello di conteggio. Come mostrato in figura 1.7, il segnale PWM viene ottenuto come uscita del flip-flop

S-R, avente come ingressi l'uscita del comparatore ed un segnale di sincronismo alla frequenza di switching.

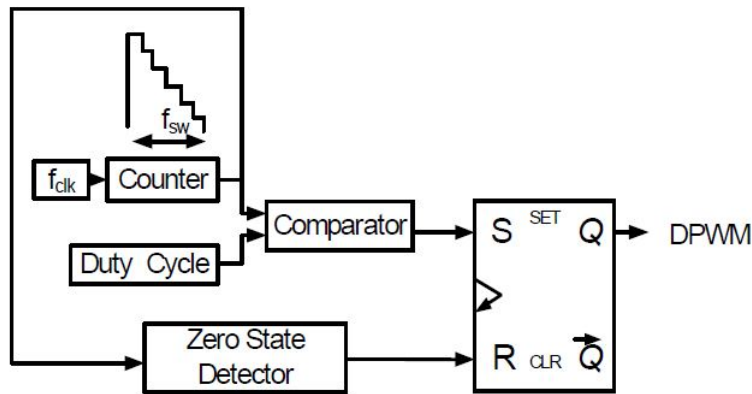


Figura 1.7: Leading-edge [36]

1.2.1.1 Tipologie di counter-based

A seconda di come vengono scelti i segnali di set e di reset si possono avere diverse tipologie di counter-based [12]:

- leading-edge;
- trailing-edge;
- double-edge.

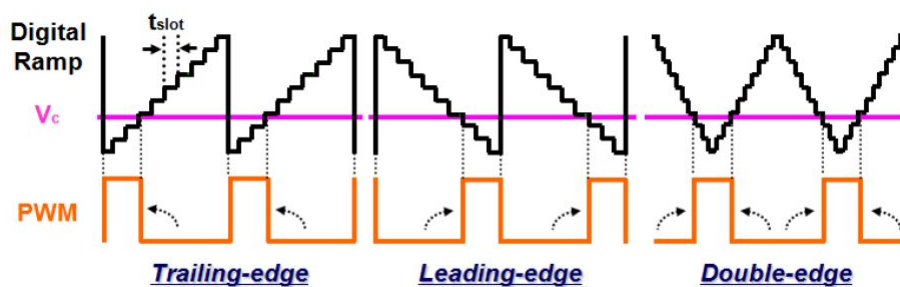


Figura 1.8: Tipologie di DPWM counter-based [12]

Le sottocategorie leading-edge e trailing-edge sono ottenute distinguendo il caso in cui l'uscita del comparatore svolge la funzione di set (leading-edge), o di

reset (trailing-edge) del flip-flop; la sottocategoria double-edge invece, viene realizzata ponendo la condizione che il segnale in uscita al contatore sia simmetrico rispetto al semi periodo.

Analizzando i singoli casi di modulazione si nota che, con la modulazione leading-edge, la generazione del segnale di set e quindi di creazione dell'impulso PWM, avviene nel momento in cui il valore assunto dal duty-cycle risulta maggiore di quello del contatore; il segnale di reset, invece, si ha ad ogni termine del ciclo di conteggio.

Nella modulazione trailing-edge avviene l'inversione della scala di conteggio e di conseguenza il segnale di set coincide con l'inizio di ogni periodo di conteggio; il segnale di reset, invece, si ha nel momento in cui il valore del duty-cycle risulta minore di quello del contatore. In entrambi i casi il periodo dell'impulso PWM coincide con il periodo in cui il valore del duty-cycle risulta maggiore di quello del contatore.

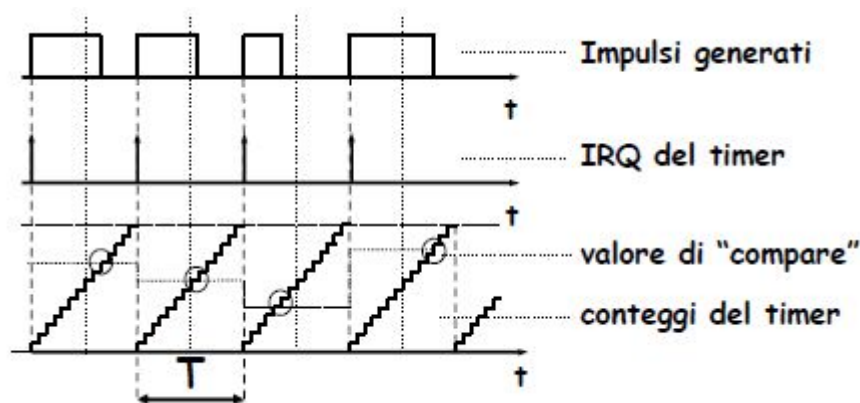


Figura 1.9: Leading-edge [10]

Con la tipologia double-edge, l'impulso PWM viene generato con l'utilizzo di una doppia scala di conteggio simmetrica e risulta di durata pari a quella del periodo in cui il valore del duty-cycle appare inferiore a quello del conteggio. Grazie a questa variante si assicura un maggior controllo, in quanto il valore in uscita al comparatore viene verificato due volte in un periodo, una per la rampa di salita e una per quella di discesa [9]. Questo vantaggio, purtroppo, presenta un costo in riferimento al raddoppio della frequenza di clock dovuta alla presenza della duplice rampa.

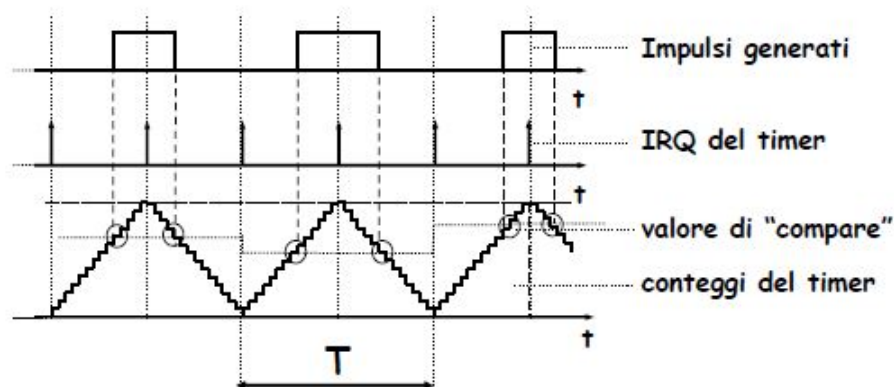


Figura 1.10: Dual-edge [10]

1.2.1.2 Segnale di sincronismo e campionamento del segnale

Nella sincronizzazione tra il modulatore PWM ed il convertitore ADC, risulta necessario prestare particolare attenzione ai valori attribuiti alle frequenze di modulazione e campionamento. Per la scelta di tali valori è importante tenere conto di alcune considerazioni:

- per ottenere elevate prestazioni è conveniente avere una frequenza di campionamento il più elevata possibile;
- al fine di garantire la stabilità è necessario imporre una frequenza di modulazione maggiore del valore della banda di controllo digitale; in caso contrario il sistema potrebbe diventare instabile a causa del ritardo della risposta del modulatore.

Viene quindi comunemente scelto di uguagliare le due frequenze.

Da notare che tale uguaglianza non soddisfa la condizione di Shannon; non è dunque possibile ricostruire il segnale continuo a partire dalla sequenza campionata e sussiste il rischio di incorrere così nel fenomeno dell'aliasing. Nella PWM digitale l'informazione necessaria, ai fini del corretto funzionamento del convertitore, risulta essere quello del valore medio del segnale, e, di conseguenza, l'errore di aliasing non viene ritenuto dannoso, bensì utile ai fini dell'estrazione di tale valor medio.

Come è possibile vedere in figura 1.11, effettuando il campionamento sempre nello stesso istante del periodo di modulazione, la grandezza campionata risulta priva di ripple residuo [9].

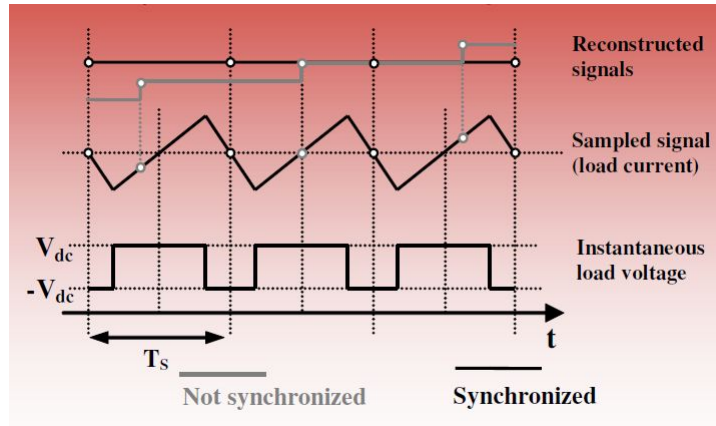


Figura 1.11: Campionamento con e senza sincronizzazione [9]

Si nota, inoltre, che il valore medio della corrente si può facilmente ottenere campionando in modo adeguato il segnale. Se ci si riferisce all'utilizzo della modulazione leading-edge e trailing-edge si deduce che campionando il segnale a metà del valore di "compare", si ottiene direttamente il valor medio del segnale. Nel caso della modulazione double-edge sfruttando la simmetria che la caratterizza, si intuisce che i due picchi del ripple coincidono con i due valori di "compare", che sono situati in modo speculare rispetto alla sommità del conteggio; in questo caso l'istante migliore di campionamento per ottenere il valore medio risulta quello centrale del conteggio.

1.2.1.3 Frequenza di clock e risoluzione

La frequenza di clock è ottenuta da:

$$f_{clk} = f_{sw} \cdot 2^n \quad (1.4)$$

Dove con n viene indicato il numero di bit del contatore e con f_{clk} la frequenza di ingresso.

La risoluzione del duty-cycle è data da:

$$\Delta D = \frac{f_{sw}}{f_{clk}} \quad (1.5)$$

mentre la risoluzione della tensione di uscita, ad esempio di un convertitore buck in CCM, è data da:

$$\Delta V = V_{in} \cdot \Delta D \quad (1.6)$$

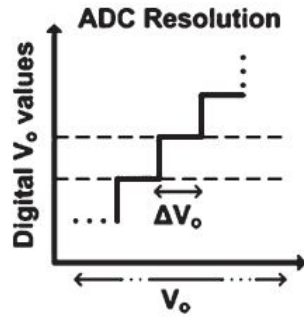


Figura 1.12: Tipologie di DPWM counter-based

Con riferimento alla figura 1.13, ricavata da test effettuati applicando la modulazione counter-based ad un convertitore buck si nota che, posta la risoluzione della tensione di uscita pari a 3mV e la tensione di ingresso pari a 12V, per avere una $f_{sw} = 300kHz$ risulta necessaria una $f_{clk} = 1.2GHz$. Per frequenze di clock superiori al GHz risulta decisamente complicata l'implementazione, dunque per elevate f_{sw} (elevate f_{clk}) si preferiscono altre tipologie di modulazione [12].

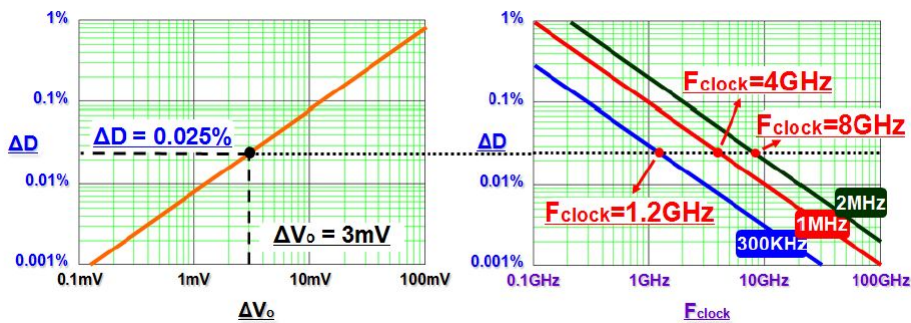


Figura 1.13: Double-edge [12]

La tipologia counter-based possiede il vantaggio di essere di facile realizzazione, in quanto è sufficiente implementare un contatore per realizzare il circuito di conteggio. Gli svantaggi si riferiscono all'elevato valore della frequenza di clock necessaria per il corretto funzionamento (figura 1.13) ed all'elevato consumo di potenza (circa 30mW [33]). Tale consumo può essere accettabile per convertitori DC-DC nel range di potenza dei watt ma non dei milliwatt.

1.2.2 Delay-line

La tipologia delay-line differisce dalla precedente per la sostituzione del contatore binario con una serie di celle di ritardo connesse in cascata (delay-line) il cui

numero è pari a 2^n , dove n è il numero di bit di cui è composto il duty-cycle [5]. La selezione delle celle viene effettuata tramite l'utilizzo di un multiplexer avente come ingressi le uscite delle celle, come segnale di controllo il duty-cycle e come uscita il set (leading-edge) oppure il reset (trailing-edge) del flip-flop come si vede in figura 1.14.

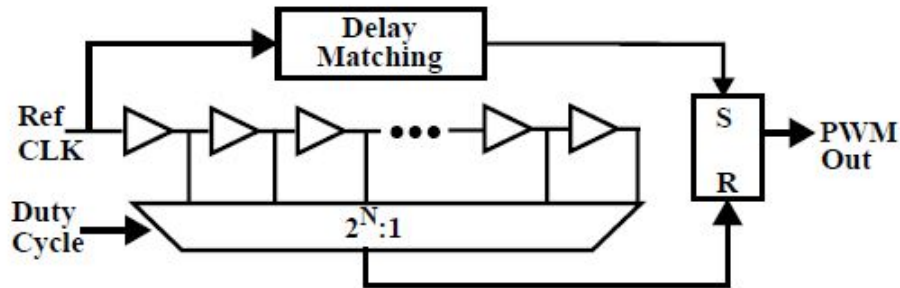


Figura 1.14: Delay-line [3]

Tale struttura riduce, rispetto alla modulazione counter-based, il consumo di potenza il quale si è stimato essere dell'ordine dei $10\mu W$ [34]. Allo stesso tempo, però, maggiore è la risoluzione desiderata e maggiore sarà il numero di elementi di ritardo, i quali portano ad una considerevole occupazione di area. Tale modulazione risulta infine molto sensibile alle variazioni di temperatura e di processo a causa dei componenti che compongono le celle di ritardo. Tale dipendenza è esprimibile con una costante K (presente nell'equazione 1.7) che varia a seconda del dispositivo utilizzato. In figura 1.15 viene riportato un esempio di struttura di una cella di ritardo.

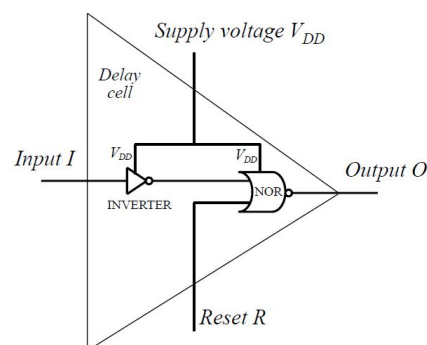


Figura 1.15: Cella di ritardo [3]

La tipologia delay-line presenta due sottocategorie:

- open-loop;

- ring-oscillator.

Il primo si basa sull'utilizzo di un oscillatore esterno, necessario per imporre la frequenza di switching, mentre nel secondo caso è la linea di ritardo stessa che impone tale valore. L'effettiva differenza di funzionamento si riscontra a seguito di una variazione dei parametri del circuito, ad esempio per una variazione di temperatura. Con il sistema open-loop può esserci una divergenza tra il valore del duty-cycle eseguito (quello di controllo del multiplexer) e quello di comando (segnale di ingresso della cascata di celle), con la conseguente generazione di malfunzionamenti non presenti nel caso con un ring-oscillator [3].

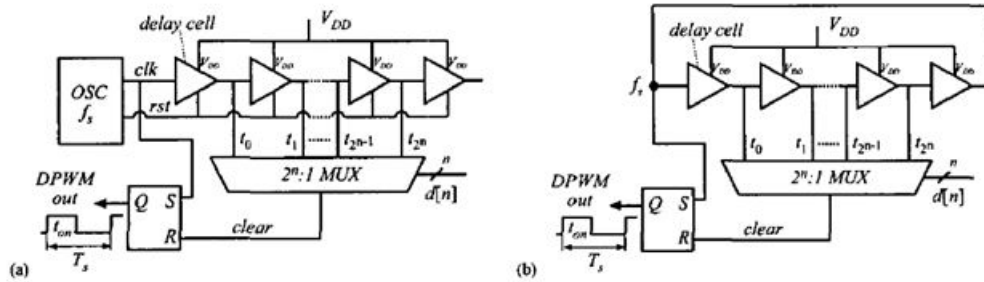


Figura 1.16: a) delay-line open-loop b) delay-line ring-oscillator [3]

Il tempo di ritardo con cui si propaga il segnale lungo la cascata di elementi è dato da:

$$t_d = K \frac{V_{dd}}{(V_{dd} - V_{th})^2} \quad (1.7)$$

dove V_{th} è la tensione di soglia. Risulta quindi evidente la dipendenza diretta che esiste tra V_{dd} e t_d .

Viene di seguito riportato un esempio di Delay-line che varia dalla precedente nella struttura (in quanto viene utilizzato un encoder), ma non nel funzionamento.

L'uscita dei singoli flip-flop, aventi come ingressi le uscite degli elementi di ritardo ed il segnale di sample (segnale di campionamento), compongono le entrate dell'encoder. L'immagine seguente si riferisce ad un test, dove il segnale di ingresso (segnale di test) si propaga fino al sesto elemento di ritardo e quindi l'uscita viene letta come 11 111 100 = ($q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8$) [6].

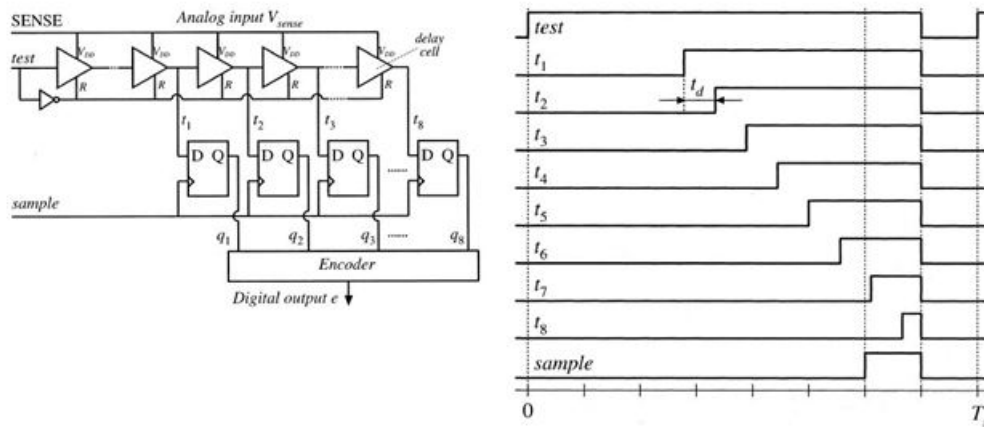


Figura 1.17: delay-line test [6]

Con questa modulazione è permesso il campionamento alle alte frequenze, in un range compreso tra 100kHz ed i MHz.

1.2.3 Hybrid

Dall'unione delle due precedenti tipologie di modulazione si ottiene la Hybrid DPWM, la cui struttura presenta sia il contatore binario, il cui segnale viene confrontato con quello del duty-cycle, sia la cascata di celle il cui ingresso è esattamente l'uscita del comparatore.

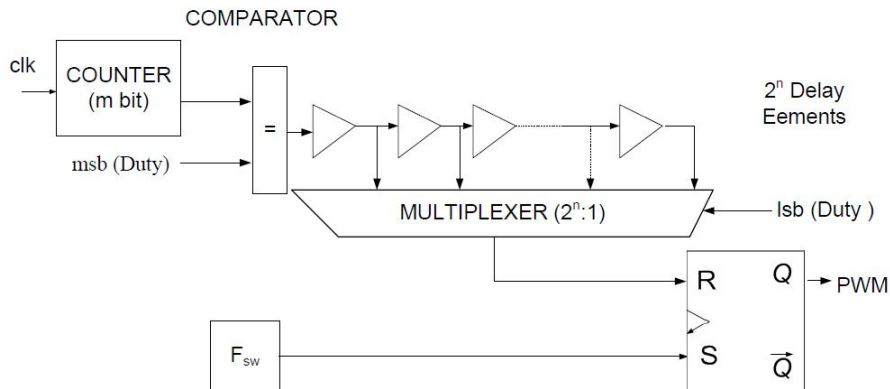


Figura 1.18: Hybrid [36]

Per modificare la frequenza di switching è sufficiente variare la delay-line aggiungendo delle ulteriori celle tra l'uscita del flip-flop ed il successivo elemento di

ritardo, mentre per quanto riguarda il contatore, esso incrementa il conteggio ad ogni ciclo di clock.

Per ottenere n -bit di risoluzione risulta necessario imporre:

$$n = n_d + n_c \quad (1.8)$$

dove n_c sono i bit del contatore ed n_d i bit della delay-line.

La risoluzione del duty-cycle è data da:

$$\Delta D = \frac{f_{sw}}{f_{clk} \cdot n_d} \quad (1.9)$$

Con la figura 1.19 si mostra sia il funzionamento sia la struttura della modulazione hybrid [6].

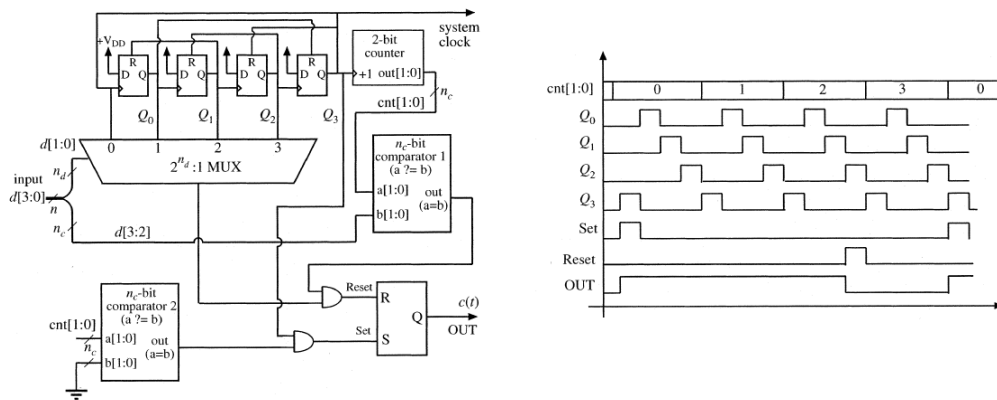


Figura 1.19: Hybrid test [6]

All'inizio del periodo di switching l'uscita del flip-flop è posta a set e dunque l'uscita è alta. Nel momento in cui l'uscita del contatore risulta uguale al valore del bit più significativo del duty-cycle e l'impulso di start raggiunge l'altezza desiderata della delay-line (selezionata tramite il bit meno significativo del duty-cycle n_d), l'uscita del flip-flop viene resettata e quindi l'impulso scende.

Si riporta di seguito un ulteriore esempio di modulazione Hybrid dove il segnale di clock viene assegnato esternamente.

Con il termine del_{clk} viene indicato l'esito del confronto tra l'uscita del contatore ed i 3 bit più significativi del comando di duty-cycle, segnale che si propaga poi attraverso la linea di ritardo [7].

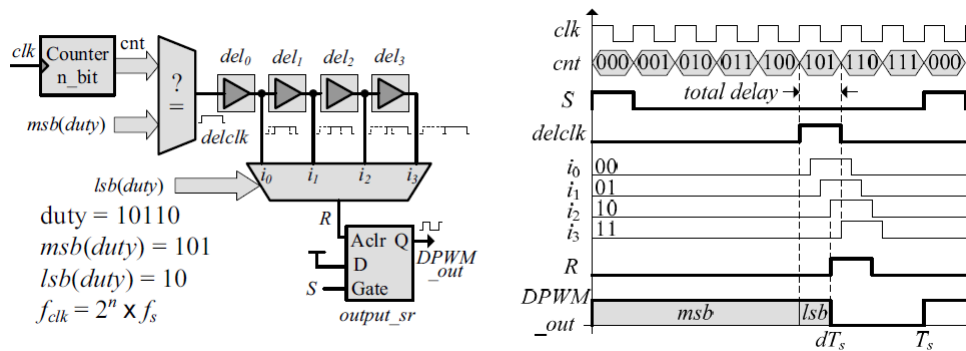


Figura 1.20: Hybrid test con assegnazione esterna del segnale di clock [7]

Risulta conveniente l'utilizzo di questa modulazione nel caso sia richiesta una elevata risoluzione (come si può vedere dal confronto tra l'equazione 1.5 e l'equazione 1.9). Inoltre l'area occupata per la realizzazione della modulazione hybrid risulta di 9 volte inferiore a quella occupata dalla delay-line [35].

In figura 1.21 sono riportati alcuni valori relativi alla potenza che viene dissipata in un convertitore con la modulazione hybrid. Si nota che nel caso di un conteggio pari a 1024 la potenza dissipata è di circa $300\mu W$.

Parameter	Value
1024 Taps, 500kHz	
Min. Supply Voltage	2.05V
PLL & Logic Current	199.3 μ A
Analog Circuits Current	1.5 μ A
256 Taps, 500kHz	
Min. Supply Voltage	1.35V
PLL & Logic Current	42.8 μ A
Analog Circuits Current	1.5 μ A

Figura 1.21: Potenza dissipata con il valore di conteggio pari a 1024 e 256 con la tipologia hybrid [35]

1.2.4 Nuove tipologie di modulazione digitale

Dopo aver concluso la trattazione delle tre principali tipologie di modulazione PWM digitale, si presentano ora due modelli che apportano alcune varianti alla counter-based, alla delay-line ed alla hybrid. Oltre alla descrizione dell'architettura e del funzionamento delle tipologie di modulazione modificate, vengono riportati alcuni esiti di test effettuati con i relativi confronti con la tipologia originale.

1.2.4.1 SCI DPWM

Essa è una variante della tipologia counter-based, la cui particolarità consiste nell'utilizzo di un diverso segnale modulante che viene ricavato dalla variazione di tensione presente ai capi della capacità. Questa tipologia di modulazione permette di ottenere una elevata risoluzione senza il bisogno di aumentare la frequenza di clock, consente un minor consumo di energia ed una minor complessità dell'hardware. Risulta però necessario scegliere una capacità avente una tolleranza minore del 5%, in quanto l'errore sul duty-cycle dipende da quello della capacità.

L'equazione che definisce l'andamento della tensione ai capi della capacità è data da:

$$V_c = \int_0^{T_K} \frac{i}{C} dt = \int_0^{T_K} \frac{I}{K} \cdot \frac{1}{C} dt = \frac{I}{C} \cdot \int_0^{T_K} \frac{1}{K} dt \quad (1.10)$$

Dove T_K è dato da:

$$T_K = K \cdot T_{clk} \quad (1.11)$$

T_K coincide con il tempo di on dell'impulso PWM controllato dal valore K .

Il duty-cycle si ricava da:

$$d = \frac{T_K}{T_{pwm}} = \frac{K \cdot T_{clk}}{2^n \cdot T_{clk}} = \frac{K}{2^n} \quad (1.12)$$

In figura 1.22 viene mostrata sia l'architettura sia il funzionamento della modulazione SCI.

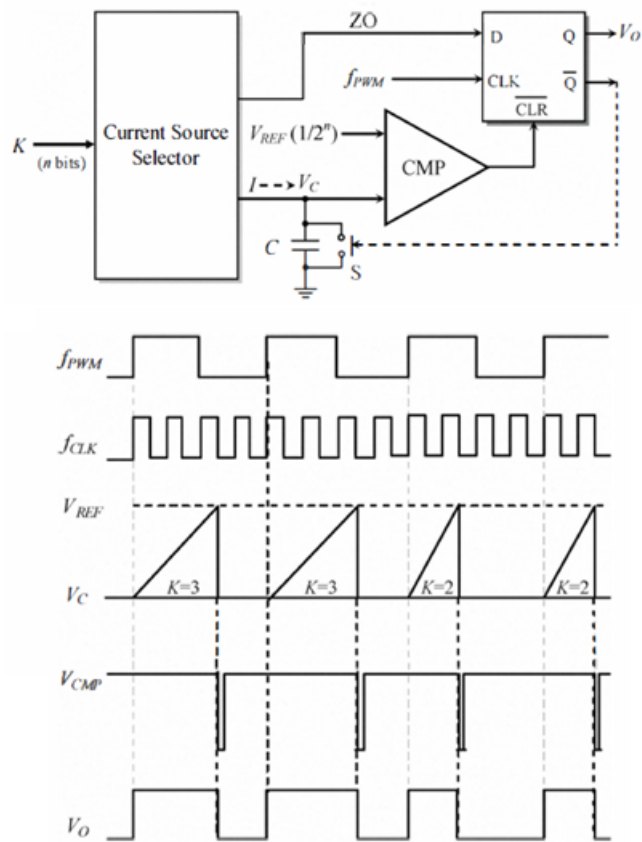


Figura 1.22: SCI DPWM [1]

Viene qui riportata una tabella di confronto che paragona i requisiti hardware necessari per la modulazione counter-based, delay-line e SCI [1].

DPMM Types	Counter-Based DPWM (CNT DPWM)	Delay-Line DPWM (DLY DPWM)	Proposed DPWM (SCI DPWM)
Required Hardware	<ul style="list-style-type: none"> • 16 MHz clock • 4-bit counter × 1 • 4-bit comparator × 1 	<ul style="list-style-type: none"> • 1 MHz clock • Delay cell × 16 • Bias current source in delay cell × 16 • 16:1 multiplexer × 1 • OP Amplifier × 1 • SR flip-flop × 1 	<ul style="list-style-type: none"> • 1 MHz clock • Current source × 15 • 4:16 decoder × 1 • Capacitor × 1 • MOS switches × 15 • Comparator × 1 • D flip-flop × 1

Figura 1.23: Tabella di confronto [1]

1.2.4.2 Limited-Pool Random Carrier-Frequency PWM

Si riporta nel seguito questa possibile variante applicata ai modelli delay-line e hybrid. Essa include l'utilizzo di un generatore di numeri casuali al fine di produrre una serie di valori da assegnare alla frequenza di switching. Si riscontra infatti che se si varia il valore di tale frequenza in un intervallo scelto accuratamente, è possibile allargare lo spettro del segnale modulato, riducendo l'energia concentrata ad una specifica frequenza. Questa affermazione viene di seguito confermata dalle immagini di figure 1.25, 1.26 e 1.27.

Questo generatore emette quindi valori entro un determinato range, producendo così la frequenza di switching che verrà utilizzata nel periodo successivo a quello dell'emissione del numero casuale, mentre il valore di duty-cycle rimane costante. Dato che l'aspetto dello spettro dipende dalla qualità con cui vengono generate le frequenze di switching, acquista particolare importanza il T_{rand} , cioè il periodo con cui si ripete la sequenza di numeri casuali.

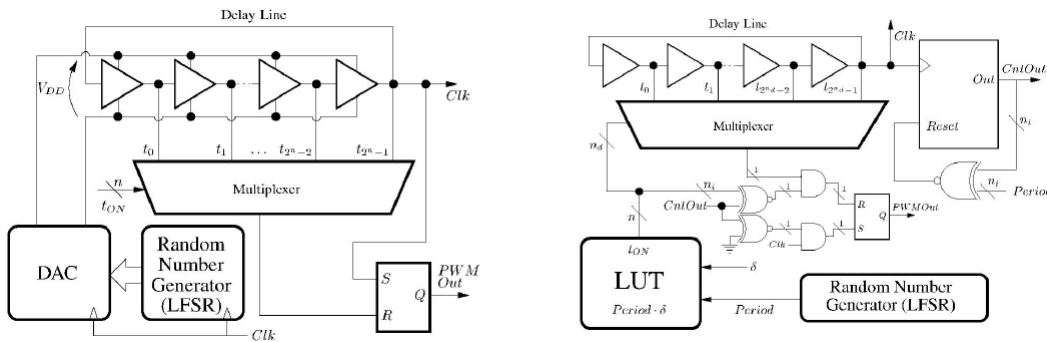


Figura 1.24: Delay-line ed hybrid random pulse width modulator [2]

Di seguito vengono presentate le immagini dell'analisi spettrale misurate in uscita del modulatore, sia con la presenza del generatore di numeri casuali sia in assenza di esso. Il test è effettuato utilizzando un convertitore buck a cui sono stati impostati i seguenti parametri: $V_{in} = 5V$, $V_0 = 1.2V$, $L=390nH$, $C=826\mu F$, $ESR=0.5m\ \Omega$ [2]. Le immagini mostrano che l'esito ottenuto con l'utilizzo del RCFMFD è migliore rispetto a quello classico. Si verifica anche che in questo modo avviene effettivamente una riduzione dell'ampiezza spettrale di circa 15dB.

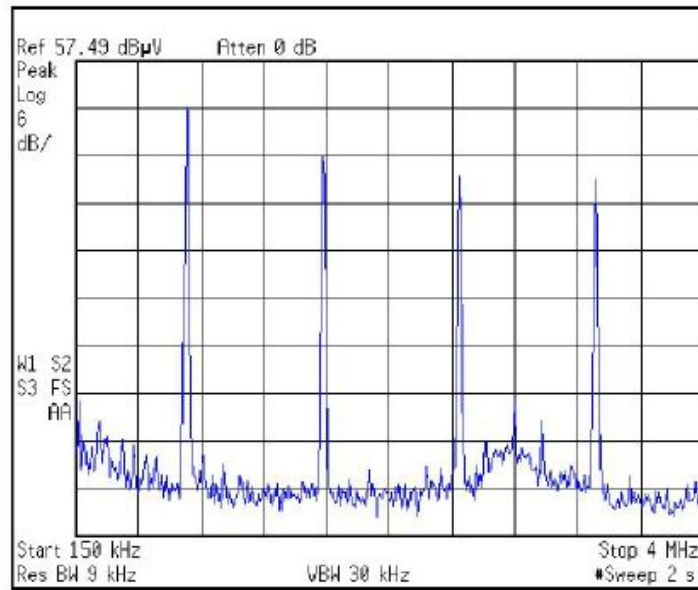


Figura 1.25: Analisi spettrale priva di modulazione random ad $f_{sw} = 833kHz$ [2]

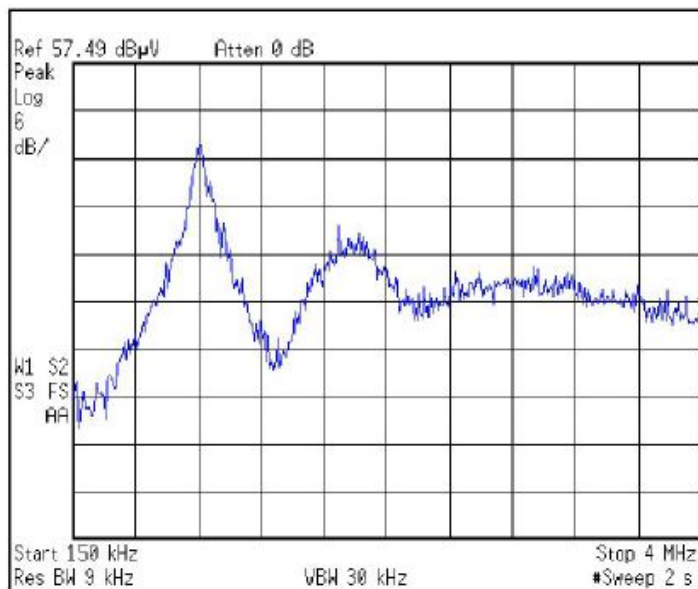


Figura 1.26: Analisi spettrale con modulazione random a $f_{sw1} = 833kHz$ e $f_{sw2} = 1.041MHz$ [2]

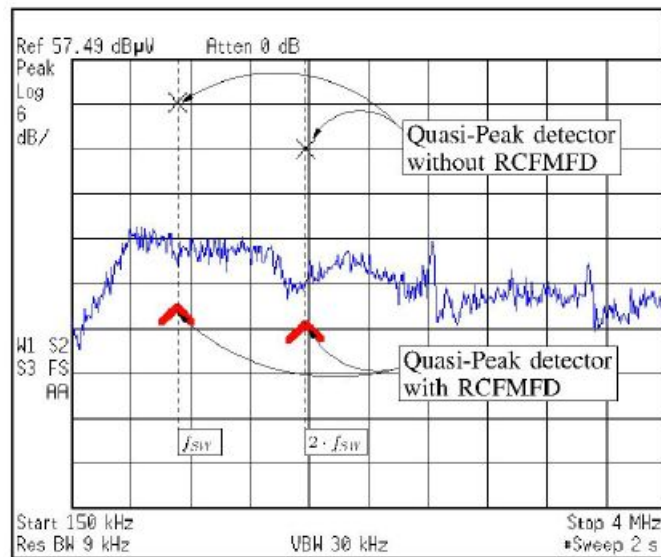


Figura 1.27: Analisi spettrale con modulazione random a $f_{sw1} = 500kHz$, $f_{sw2} = 890kHz$, $f_{sw3} = 1.200MHz$ [2]

La condizione migliore per applicare questo tipo di modulazione è quella di garantire che gli impulsi siano centrati nel periodo di switching, come mostrato in figura 1.28.

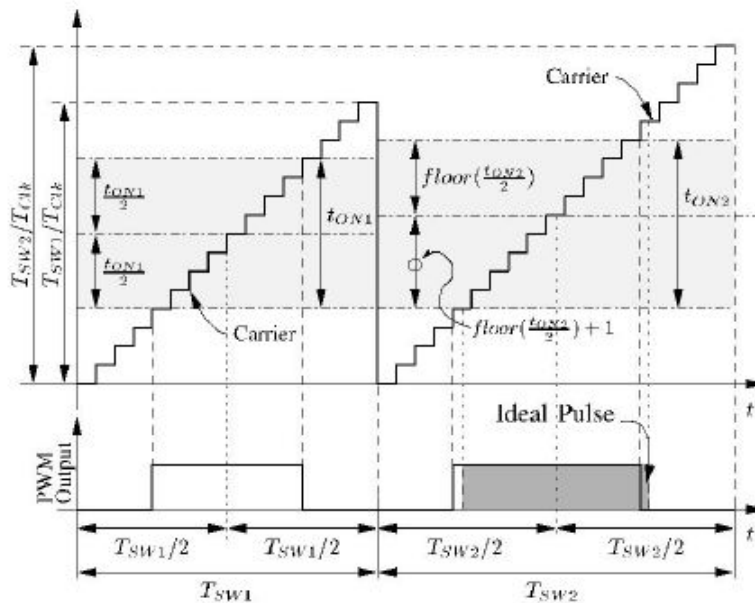


Figura 1.28: Metodo che garantisce gli impulsi centrati [2]

Il sistema ha il suo fondamento nel confronto tra la rampa di conteggio e due differenti valori $T_{sw}[\frac{(1-d)}{2}]$ e $T_{sw}[\frac{(1-d)}{2} + d]$; in questo modo la durata del segnale PWM risulta pari al tempo che intercorre tra il raggiungimento da parte della rampa dei due valori di paragone. Questo sistema assicura la massima precisione possibile e la centralizzazione dell'impulso nel periodo.

1.3 Confronto tra modulazione PWM digitale ed analogica

Conclusa la trattazione sulla modulazione PWM analogica e digitale si effettua ora un loro confronto, al fine di valutare i pregi ed i difetti di entrambe.

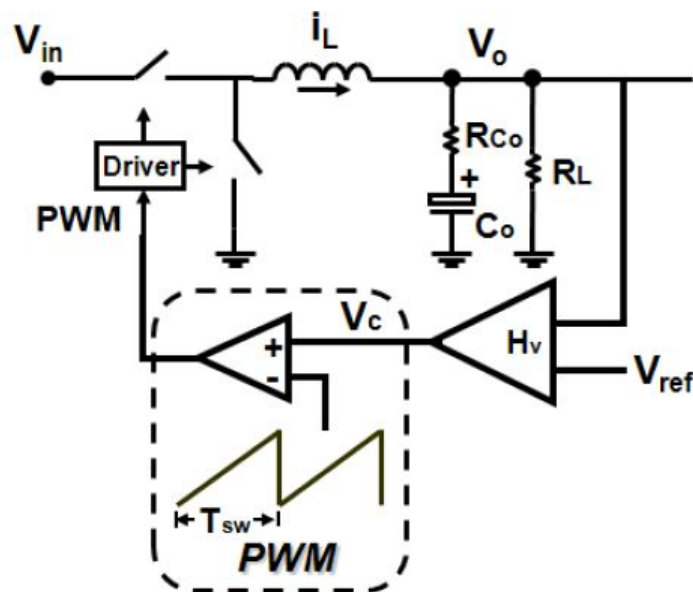


Figura 1.29: Modulazione analogica [12]

L'utilizzo della modulazione PWM analogica possiede il vantaggio di essere di facile realizzazione e non necessita di circuiteria particolarmente complicata. A sfavore si riscontra una elevata sensibilità sia alle variazioni termiche che comportano un cambiamento di valore dei componenti elettronici di cui è composto (e di conseguenza una variazione dei parametri del regolatore), sia al rumore durante l'elaborazione e la trasmissione delle grandezze.

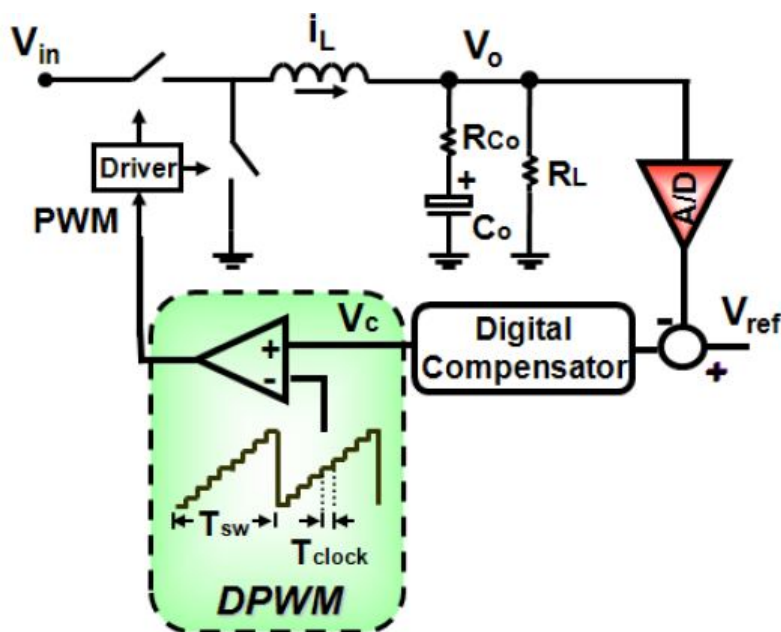


Figura 1.30: Modulazione digitale [12]

I vantaggi che si riscontrano nell'applicazione del PWM digitale rispetto all'analogico si riferiscono, in primo luogo, ad un minor costo e tempo di realizzazione. Infatti l'impiego di software CAD (computer aided design) automatici di sviluppo permette una evidente semplificazione nella sintesi del layout del circuito. Inoltre le strutture digitali, contrariamente a quelle analogiche, sono direttamente scalabili in presenza di miglioramenti tecnologici dei livelli di integrazione e quindi facilmente aggiornabili per i futuri livelli di integrazione; si può dire quindi che sono dotati di una maggiore flessibilità. Grazie al controllo digitale viene migliorata l'efficienza del sistema ed è garantita una maggiore affidabilità senza la necessità di procedura di taratura. Esiste infine la possibilità di eseguire funzioni più avanzate e sofisticate grazie alle notevoli potenzialità di calcolo.

Un difetto che si riscontra in un controllo di tipo digitale, che non è presente in quello analogico, è il ritardo introdotto dall'unità che esegue il calcolo del segnale modulante. Infatti, una volta che le variabili da controllare sono state campionate e convertite in digitale, esse devono essere processate dall'unità di calcolo che impiega un tempo non nullo per eseguire il compito. Il rischio è quello di aggiornare la grandezza di controllo (duty-cycle), sulla base di uno stato del sistema precedente rispetto a quello in cui si trova nel momento in cui si esegue l'aggiornamento della modulante. Questo ritardo può portare, nel caso peggiore, all'instabilità del sistema, oppure può causare oscillazioni statiche. Tali oscillazioni, anche se non portano all'instabilità hanno l'effetto di aumentare

il ripple delle grandezze controllate, il che ha un forte impatto sugli stress dei componenti utilizzati e complicano il dimensionamento dei filtri (le oscillazioni si presentano a frequenze minori di quelle della portante). Questo tempo di risposta alle variazioni presenti nel sistema modulante, nel caso peggiore, risulta pari a T . Il problema può essere in parte attenuato nel caso in cui sia possibile l'aggiornamento del duty-cycle, anche a metà periodo di modulazione, riducendo così il tempo massimo di risposta da un periodo ad un semi periodo.

Capitolo 2

Microcontrollori

Vengono ora trattati alcuni modelli di microcontrollori utilizzati per generare dei segnali PWM nel controllo digitale. L'utilizzo del microcontrollore è una delle strade che si possono percorrere. Una valida alternativa per l'implementazione del sistema di controllo consiste nell'utilizzo della FPGA. Di seguito verranno introdotte le caratteristiche dei microcontrollori, e verranno apportati alcuni esempi di famiglie di microcontrollori presenti in commercio con lo scopo di avere un termine di paragone in riferimento ai limiti ed alle doti del CompactRIO.

I microcontrollori sono dei microprocessori integrati in un solo chip, specializzati nelle applicazioni di controllo. Essi sono progettati per interagire direttamente con il mondo esterno tramite un programma presente nella memoria interna e grazie all'utilizzo di pin configurabili dal programmatore.

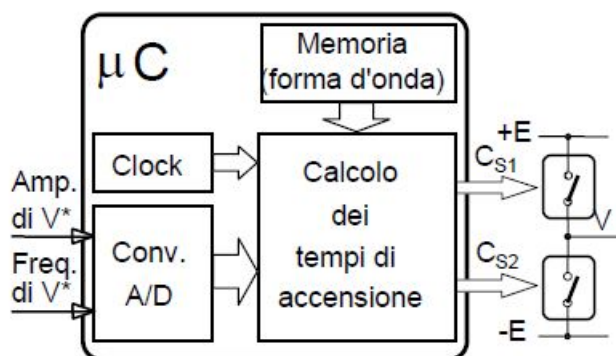


Figura 2.1: Modulazione digitale a due livelli con microcontrollore [11]

I microcontrollori comprendono:

- una CPU,
- un certo quantitativo di memoria RAM e memoria ROM (PROM, EPROM, EEPROM, FlashROM),

- una serie d'interfacce di I/O (input/output) standard.

Le periferiche integrate sono la vera forza di questi dispositivi, tra le quali come si vede in figura, si possono avere:

- convertitori ADC, convertitori DAC multicanale,
- timer/counters,
- numerose porte esterne bidirezionali bufferizzate,
- I/O seriali (USART, I²C, SPI, CAN, USB),
- PWM [20].

Per la programmazione dei microcontrollori il migliore linguaggio possibile, a livello di ottimizzazione della memoria occupata e dell'efficienza della gestione delle periferiche è l'Assembly, ma questo è anche il più difficile da utilizzare. Per questo motivo viene comunemente utilizzato il linguaggio di programmazione C.

Contrariamente, per la programmazione delle FPGA si utilizza il linguaggio VHDL e per il CompactRIO il LabView. Sicuramente tra i tre linguaggi di programmazione il più semplice ed intuitivo da apprendere è il linguaggio LabView, in quanto trattasi di un linguaggio grafico. Tra i tre sistemi di controllo il CompactRIO risulta purtroppo essere il più costoso.

2.1 Microcontrollore PIC

PIC "Programmable Interface Controller", è una famiglia di circuiti integrati a semiconduttore con funzioni di microcontrollore della Microchip technology.

Caratteristiche generali dei PIC [21]:

- sono dei microcontrollori RISC (Reduced Instruction Set Computer),
- dispongono di un watchdog, (controllo di sicurezza interno),
- utilizzano la tecnologia Pipeline, cioè lavorano in parallelismo, (mentre viene interpretata una istruzione contemporaneamente ne viene eseguita un'altra già precedentemente interpretata),
- il segnale di clock, cioè l'oscillatore utilizzato per sincronizzare le operazioni, può esser o di tipo circuitale oppure al quarzo,
- alcuni bit di configurazione vengono salvati in memoria interna non volatile EEPROM,

- risulta possibile impostare la modalità dormiente del PIC nei periodi di non utilizzo, in modo da permettere il risparmio di energia,
- nei PIC 18 le istruzioni non di salto sono eseguite in un ciclo macchina (4 periodi di clock), mentre quelle di salto in due cicli macchina,
- i PIC si classificano in tre famiglie:
 - Basso, medio e alto livello a 8bit (il livello cambia in base al numero di istruzioni possedute),
 - PIC 24 a 16 bit,
 - PIC 32 a 32 bit.

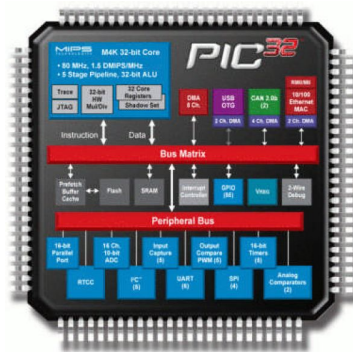


Figura 2.2: Pic 32[32]

2.1.1 MCPWM (motor control PWM)

Sono dei controllori specificatamente studiati per il comando di motori.

Si prende in considerazione il prodotto commerciale della famiglia Microchip dsPIC30F.

In commercio esistono due diverse versioni, a 6 ed a 8 uscite le cui caratteristiche sono mostrate in tabella 2.3

Feature	6-Output MCPWM Module	8-Output MCPWM Module
I/O Pins	6	8
PWM Generators	3	4
Fault Input Pins	1	2
Dead Time Generators	1	2

Figura 2.3: Caratteristiche dsPIC30F[22]

La base dei tempi del PWM viene determinata da un timer a 15 bit con prescaler e postscaler accessibile tramite il registro PTMR. Un secondo registro è il PTPER, il cui ruolo è quello di selezionare il counting period del PTMR.

Nelle figure 2.4 e 2.5 viene mostrato l'utilizzo dei registri sopracitati.

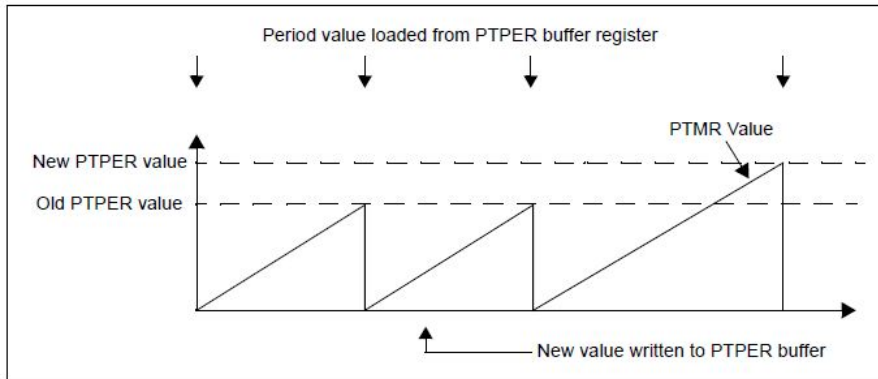


Figura 2.4: PWM Leading-Edge[22]

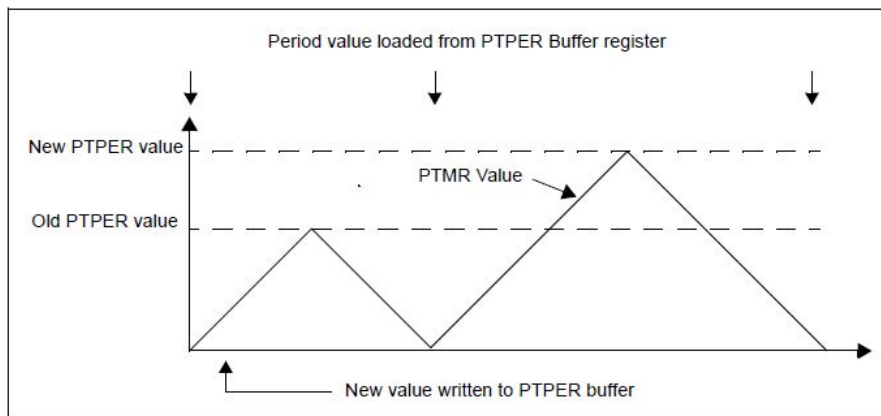


Figura 2.5: PWM Dual-Edge[22]

In riferimento alla risoluzione, essa risulta ricavabile da:

$$Risoluzione = \frac{\log\left(\frac{T_{PWM}}{T_{clk}}\right)}{\log(2)} \quad (2.1)$$

Dove T_{PWM} è il periodo di modulazione e T_{clk} il segnale di clock. L'equazione 2.1 è ricavata dall'equazione 1.4 portando a incognita il valore n (numero di bit di risoluzione) e applicando un cambiamento di base al logaritmo.

In tabella 2.6 e 2.7 vengono riportati alcuni valori di frequenza e risoluzione PWM.

Tcy (Fcy)	PTPER Value	PDCx Value for 100%	PWM Resolution	PWM Frequency
33 ns (30 MHz)	0x7FFE	0xFFFE	16 bits	915 Hz
33 ns (30 MHz)	0x3FE	0x7FE	11 bits	29.3 kHz
50 ns (20 MHz)	0x7FFE	0xFFFE	16 bits	610 Hz
50 ns (20 MHz)	0x1FE	0x3FE	10 bits	39.1 kHz
100 ns (10 MHz)	0x7FFE	0xFFFE	16 bits	305 Hz
100 ns (10 MHz)	0xFE	0x1FE	9 bits	39.1 kHz
200 ns (5 MHz)	0x7FFE	0xFFFE	16 bits	153 Hz
200 ns (5 MHz)	0x7E	0xFE	8 bits	39.1 kHz

Figura 2.6: Modalità free running[22]

Tcy (Fcy)	PTPER Value	PDCx Value for 100%	PWM Resolution	PWM Frequency
33 ns (30 MHz)	0x7FFE	0xFFFE	16 bits	458 Hz
33 ns (30 MHz)	0x3FFE	0x7FFE	15 bits	916 Hz
50 ns (20 MHz)	0x7FFE	0xFFFE	16 bits	305 Hz
50 ns (20 MHz)	0x1FFE	0x3FFE	14 bits	1.22 kHz
100 ns (10 MHz)	0x7FFE	0xFFFE	16 bits	153 Hz
100 ns (10 MHz)	0xFFE	0x1FFE	13 bits	1.22 kHz
200 ns (5 MHz)	0x7FFE	0xFFFE	16 bits	76.3 Hz
200 ns (5 MHz)	0x7FE	0xFFE	12 bits	1.22 kHz

Figura 2.7: Up/down counting mode [22]

2.2 Microcontrollori RENESAS

Al fine di avere ulteriore materiale di confronto, dopo aver trattato i microcontrollori PIC, vengono ora analizzati i microcontrollori RENESAS.

Nello specifico si prende in considerazione il prodotto commerciale RX630 che possiede le seguenti caratteristiche:

- opera ad una frequenza massima di 100MHz,
- 32bit RX CPU,
- possiede un watchdog timer,
- il minimo tempo di esecuzione risulta essere pari ad un ciclo di clock [25].

In figura 2.8 viene mostrato un esempio di realizzazione della modulazione PWM con RX630.

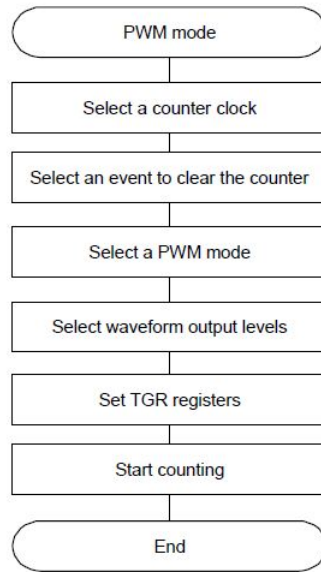


Figura 2.8: Realizzazione della modulazione PWM[23]

Caratteristiche simili a quelle del microcontrollore RX630 sono presenti ad esempio anche nel microcontrollore R8C/35A, di cui verrà esposto l'utilizzo del metodo di campionamento precedentemente descritto (richiamato dall'immagine di figura 2.9), grazie all'ausilio di due timer di 16 bit, chiamati RC ed RD.

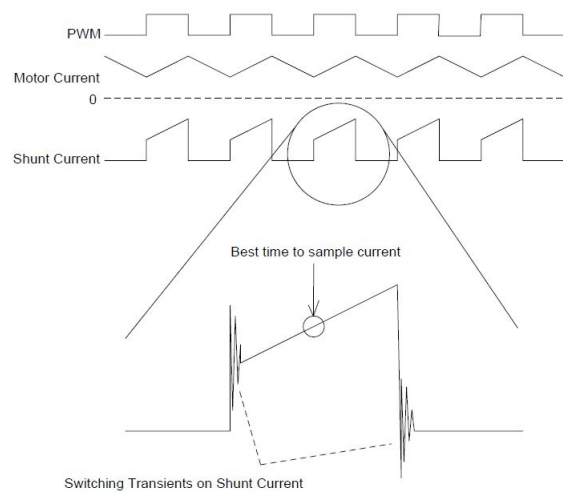


Figura 2.9: Funzionamento di un motore [14]

Nel caso di una modulazione Single-Edge si utilizza il timer RC, i cui registri svolgono la funzione di determinare diversi istanti riferiti ai corrispettivi livelli del contatore, stabilendo così il periodo di modulazione ed il duty-cycle. Il limite del

metodo è il dovere restringere l'intervallo di variazione di duty-cycle in modo tale da ottenere delle misurazioni corrette del segnale. Il campionamento infatti, deve essere effettuato all'interno del periodo dell'impulso e quindi risulta necessaria una scelta corretta dei clock del timer e del convertitore analogico-digitale.

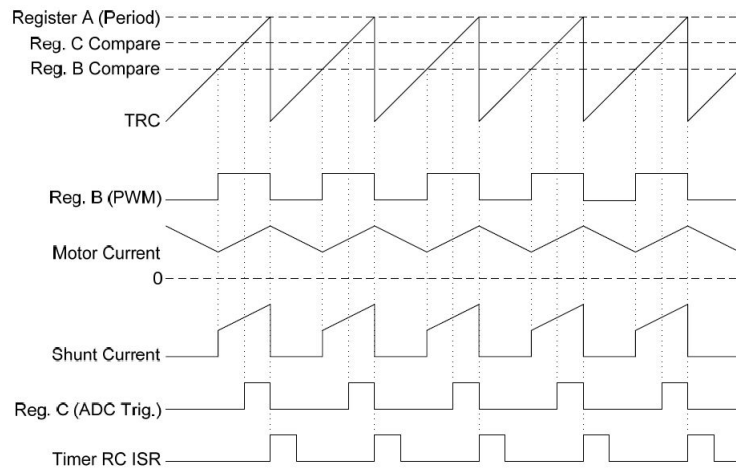


Figura 2.10: Controllo di un motore DC [14]

Nella figura 2.11 viene mostrato il funzionamento del controllo di un motore brushless DC dove è possibile notare come nella modulazione double-edge sia conveniente campionare nel punto di picco del contatore.

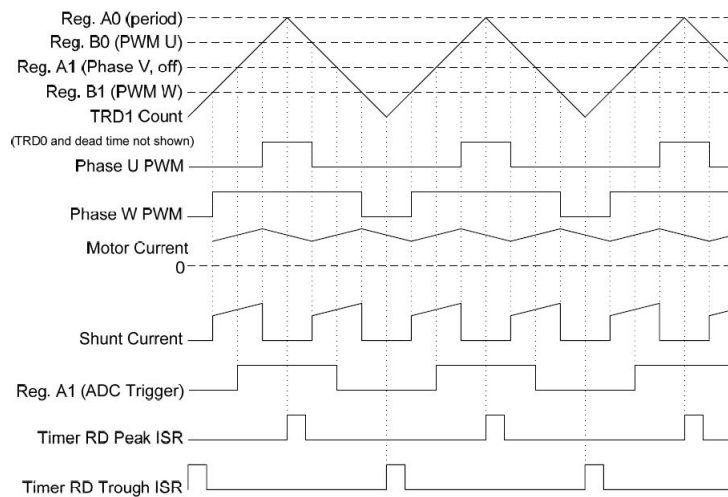


Figura 2.11: Controllo di un motore brushless DC [14]

Capitolo 3

CompactRIO

Il CompactRIO è un sistema di monitoraggio e di controllo riconfigurabile, esso unisce un'architettura embedded con moduli di I/O industriali di dimensioni ridotte, estremamente robusti e tecnologia FPGA I/O (RIO) riconfigurabile per il controllo e l'acquisizione dati.



Figura 3.1: CompactRIO [16]

Il CompactRIO è programmabile con gli strumenti di programmazione grafica di NI LabVIEW. Questa programmazione si basa sul flusso di dati ed i programmi realizzati sono chiamati VI (Virtual Instrument), in quanto essi graficamente riproducono l'aspetto di strumenti reali e ne simulano perfettamente le azioni, senza scrivere alcuna riga di codice.

3.1 Struttura del CompactRIO

Il compactRIO, come si vede in figura 3.2, è composto dal CHASSIS dotato di chip FPGA riconfigurabile, Real Time controller e da moduli industriali di I/O.

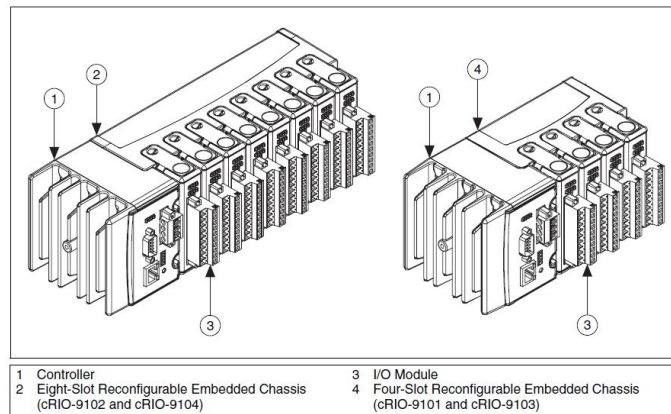


Figura 3.2: Struttura del compactRIO [15]

Questo sistema permette un'architettura a basso costo con accesso aperto alle risorse hardware di basso livello, inoltre assicura flessibilità nella temporizzazione, nella sincronizzazione e nella procedura di trigger.

- CHASSIS: è il dispositivo principale del sistema CompactRIO in quanto integra il nucleo FPGA I/O riconfigurabile che è collegato ad ogni modulo I/O [16].



Figura 3.3: CHASSIS da 8 slot[16]

- CONTROLLER: il real-time controller è un processore molto potente, a cui è collegato il chip FPGA attraverso un bus PCI ad alta velocità. Il processore real-time ed il chip FPGA riconfigurabile sono integrati all'interno di un singolo CHASSIS. In figura 3.5 sono presenti:

- una porta seriale;
- doppia tensione di alimentazione agli ingressi (11-30 VDC);
- interruttori DIP, LED (per indicare lo stato);
- clock real-time;
- clock di allarme.



Figura 3.4: NI cRIO-9014 Real-Time Controller [19]

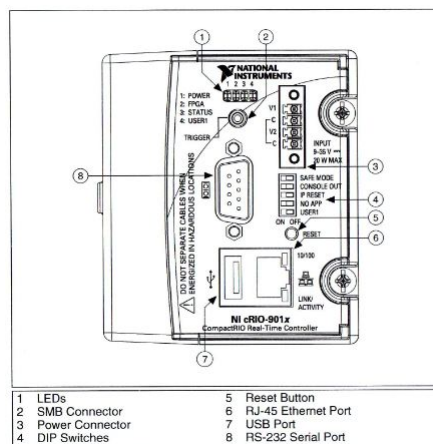


Figura 3.5: Descrizione del controllore [15]

- MODULI DI I/O: i moduli sono necessari per comunicare da e verso i dispositivi esterni come ad esempio sensori ed attuatori (figura 3.7).



Figura 3.6: Modulo I/O [19]

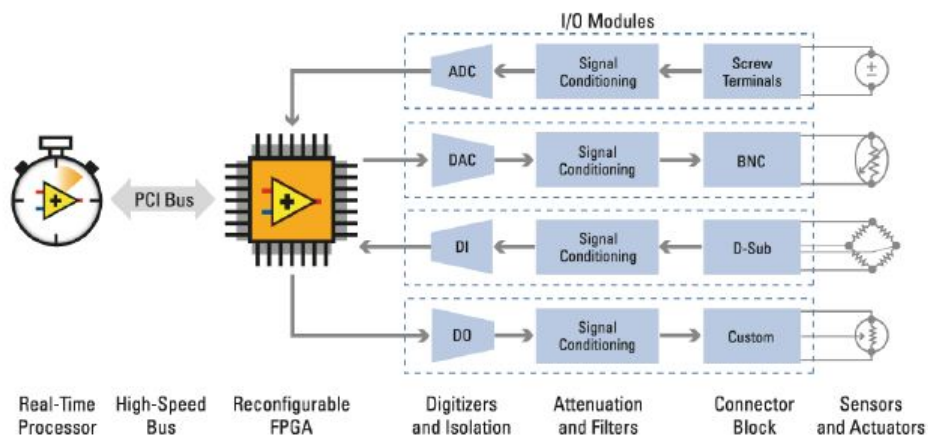


Figura 3.7: Architettura completa del CompactRIO[19]

3.2 NI cRIO-9082

Il CompactRIO utilizzato per il progetto, visibile in figura 3.8, è il "NI cRIO-9082".



Figura 3.8: CompactRIO 9082 [16]

Il quale possiede le seguenti caratteristiche:

- Processore 1.33 GHz dual-core Intel Core i7, memoria non volatile 32 GB, 2 GB DDR3 800 MHz RAM.
- 1 MXI-Express, 4 USB Hi-Speed, 2 Gigabit Ethernet, 2 porte seriali, per connettività ed espansione.
- Chassis 8-slot Spartan-6 LX150 FPGA per funzioni personalizzate di temporizzazione, controllo ed elaborazione I/O.
- Temperatura operativa da 0 a 55 °C.

3.2.1 Moduli I/O.

Di seguito si elencano i moduli I/O scelti per il cRIO-9082.

Il modulo NI 9215 possiede le seguenti caratteristiche [31]:

- 4 canali di ingresso analogici a campionamento simultaneo da 100kS/s.
- Calibrazione NIST-traceable.
- Risoluzione a 16 bit.
- Funzionamento hot-swapping.
- Sistema operativo Real-time, Windows.
- Massima tensione analogica in ingresso risulta pari a 10 V ed il massimo intervallo di tensione tra i -10V ed i 10V.
- Massima accuratezza è di 0.003 V.
- Temperatura di immagazzinamento rientra nell'intervallo da -40 a 85 °C.
- Temperatura di funzionamento rientra nell'intervallo da -40 a 70 °C.



Figura 3.9: NI 9215 [31]

Il modulo NI 9263 possiede le seguenti caratteristiche [28]:

- 4 canali di uscita analogici con aggiornamento simultaneo da 100kS/s.
- Calibrazione NIST-traceable.
- Risoluzione a 16 bit.
- Funzionamento hot-swapping.
- Sistema operativo Real-time, Windows.
- Massima tensione analogica in ingresso risulta pari a 10 V ed il massimo intervallo di tensione tra i -10V ed i 10V.
- Massima accuratezza è di 0.11V.
- Temperatura di immagazzinamento rientra nell'intervallo da -40 a 85 °C.
- Temperatura di funzionamento rientra nell'intervallo da -40 a 70 °C.



Figura 3.10: NI 9263 [28]

Il modulo NI 9264 possiede le seguenti caratteristiche [29]:

- 16 canali di uscita analogici a campionamento simultaneo da 25 kS/s.
- Connettività D-Sub o con terminali a molle.
- Isolamento 250Vrms (nella versione terminale a molle) oppure 60 VDC (nella versione D-Sub).
- Risoluzione a 16 bits.
- Funzionamento hot-swapping.

- Sistema operativo Real-time.
- Massima tensione analogica in ingresso risulta pari a 10 V ed il massimo intervallo di tensione tra i -10V ed i 10V.
- Massima accuratezza è di 0.01 V.
- Temperatura di immagazzinamento rientra nell'intervallo da -40 a 85 °C.
- Temperatura di funzionamento rientra nell'intervallo da -40 a 70 °C.



Figura 3.11: NI 9264 [29]

Il modulo NI 9401 possiede le seguenti caratteristiche [30]:

- 8 canali digitali I/O bidirezionali (configurabili in base al nibble di 4bit) alla velocità di 100ns, sinking/sourcing a 5V/TTL.
- Connettore D-Sub a 25pin standard.
- Funzionamento hot-swapping.
- Sistema operativo Real-time, Windows.
- Massima frequenza di clock 10MHz.
- Intervallo input e output massimo pari a 0V -5.25 V.
- Temperatura di immagazzinamento rientra nell'intervallo da -40 a 85 °C.
- Temperatura di funzionamento rientra nell'intervallo da -40 a 70 °C.



Figura 3.12: NI 9401 [30]

3.3 FPGA

Con FPGA (field-programmable gate array) si definisce un chip di silicio riconfigurabile attraverso software per diverse applicazioni. Tutte le schede FPGA Xilinx contengono le stesse risorse base:

- Slices (raggruppate in CLB) che contengono logica combinatoria e registri;
- I/O Blocks che sono le interfacce tra l'FPGA e il mondo esterno;
- interconnessioni programmabili;
- altre risorse quali memoria, moltiplicatori, buffer di clock globali e logica per Boundaryscan [18].

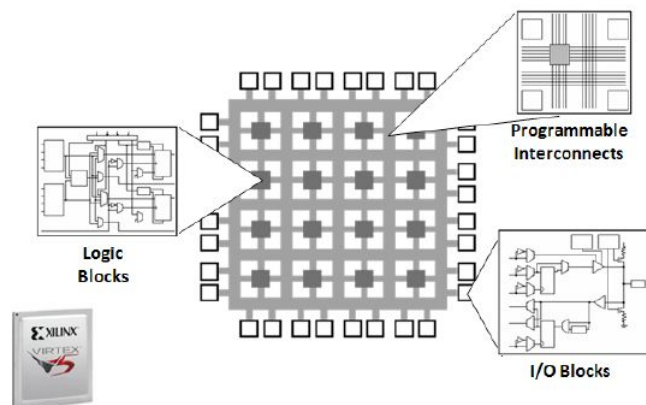


Figura 3.13: Struttura di un chip FPGA [19]

Nell'utilizzo tradizionale delle FPGA si richiede necessariamente la presenza di un programmatore VHDL. Come si vede in figura 3.14 tale scelta di programmazione richiede un elevato tempo di sviluppo ed una discreta complessità.

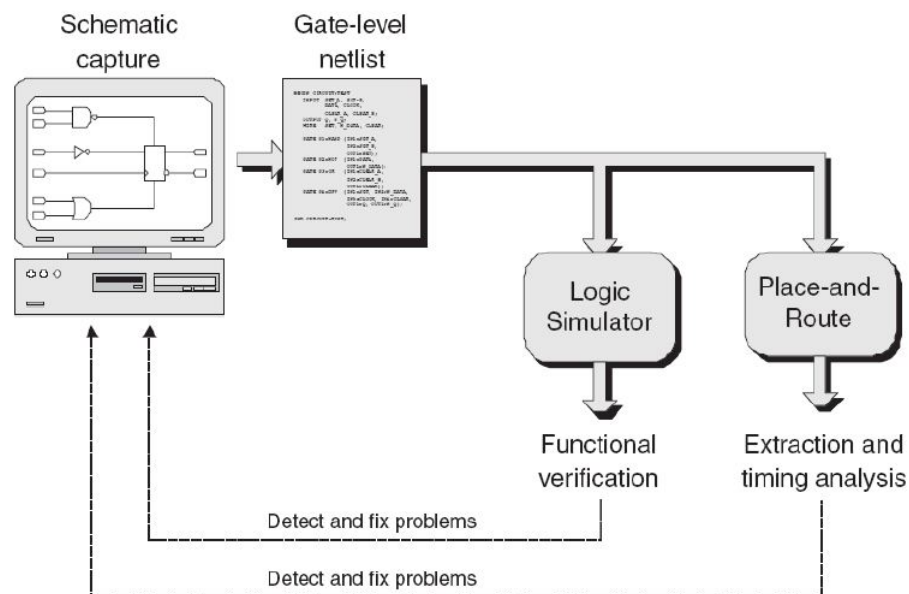


Figura 3.14: Fasi di progetto[18]

Grazie all'utilizzo di strumenti "NP", quali il cRIO, la programmazione del chip FPGA risulta estremamente semplificata in quanto il linguaggio VHDL viene sostituito con il LabVIEW. Il compito di convertire il linguaggio grafico del "VI" nel codice VHDL, al fine di programmare il chip FPGA è assegnato a LabVIEW ed al suo modulo FPGA.

In LabVIEW esistono meccanismi integrati per il trasferimento dati, infatti essi vengono trasferiti dai moduli di I/O al chip FPGA e dal chip FPGA al processore real-time embedded per analisi real-time, post processing, data-logging o comunicazioni via rete verso il PC host. Quindi il progetto, indicato come un insieme di files LabVIEW, possiede una diretta corrispondenza con il sistema CompactRIO, cioè si può dire che l'architettura software rispecchia l'architettura del progetto.

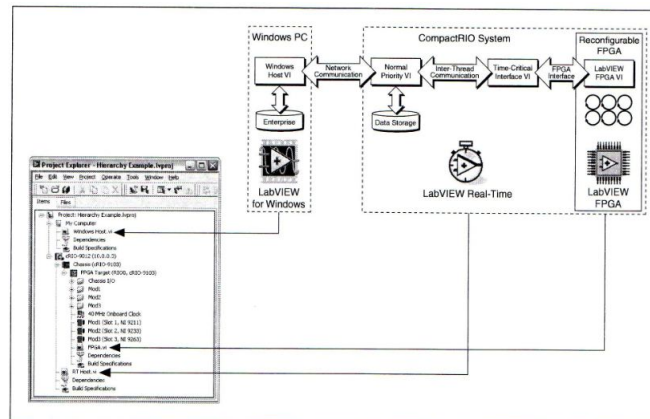


Figura 3.15: Relazione tra architettura e VIs [15]

3.4 CompactRIO Scan Mode

Utilizzando LabVIEW si può dunque programmare il CompactRIO e quindi il chip FPGA, senza l'utilizzo del VHDL. Per comunicare con i moduli senza il bisogno di programmare e compilare FPGA si utilizza la funzione CompactRIO Scan Mode, adatta per applicazioni che richiedono aggiornamenti di I/O sincroni alla frequenza desiderata fino a 1kHz. CompactRIO Scan Mode rileva in automatico i moduli di I/O e li aggiunge al LabVIEW Project. In questo modo, diventa possibile trascinare le variabili degli I/O sul proprio diagramma a blocchi del VI in LabVIEW Real-Time e nel PC host, in modo da leggere e scrivere istantaneamente i dati degli I/O [16].

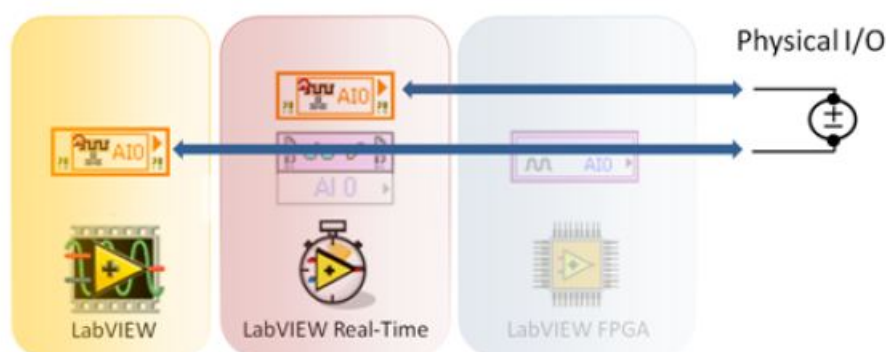


Figura 3.16: Accesso diretto agli I/O [17]

Caratteristiche specifiche del contatore:

- 1 MHz continuo su otto canali;
- count register a 32 bit, con terminal count programmabile;
- terminal count configurabile e monitoraggio del comportamento;
- fronte di segnale configurabile (salita, discesa o entrambi), sorgente e gate;
- misura di periodo, PWM e frequenza con time base configurabile.

Caratteristiche specifiche del PWM:

- 8 canali di output;
- configurazione per canale del duty-cycle e del periodo;
- periodo di frequenza configurabile (1 Hz, 50 Hz, 250 Hz, 500 Hz, 1 kHz, 5 kHz, 10 kHz, o 20 kHz) [17].

Il CompactRIO utilizza due tecnologie:

- NI Scan Engine

Trattasi di un componente di LabVIEW Real-Time, il cui compito è quello di aggiungere le variabili degli I/O ad una mappa di memoria e aggiornare i valori degli I/O. Il tutto avviene contemporaneamente a ogni termine di scansione degli I/O eseguita dal RIO Scan Interface. Inoltre, NI Scan Engine pubblica le variabili I/O sulla rete in modo tale che esse possano essere utilizzate per eventuali operazioni.

- RIO Scan Interface

Essa è responsabile delle funzioni di riconoscimento, temporizzazione, sincronizzazione e comunicazione dei moduli I/O tra FPGA ed il sistema operativo real-time.

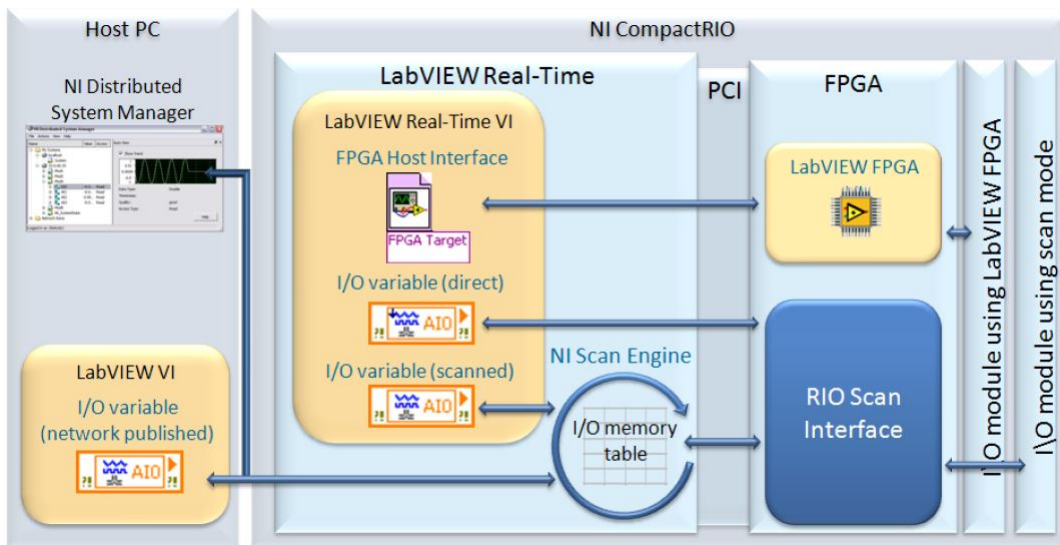


Figura 3.17: Funzionamento dello Scan Mode [17]

Per applicazioni che richiedono prestazioni più elevate, quali la modulazione digitale PWM per inverter, la funzione Scan Mode non può essere presa in considerazione; risulta invece conveniente utilizzare LabVIEW FPGA Module, descritta nella successiva sezione.

3.5 Architettura di controllo

L'ambiente di sviluppo software scelto per il progetto è LabVIEW per Windows a cui vengono aggiunti LabVIEW FPGA Module e LabVIEW Real-Time. Il controllo del sistema è quindi sviluppato in tre parti come si vede in figura 3.18:

- LabVIEW FPGA.
- LabVIEW Real-Time.
- LabVIEW per Windows.

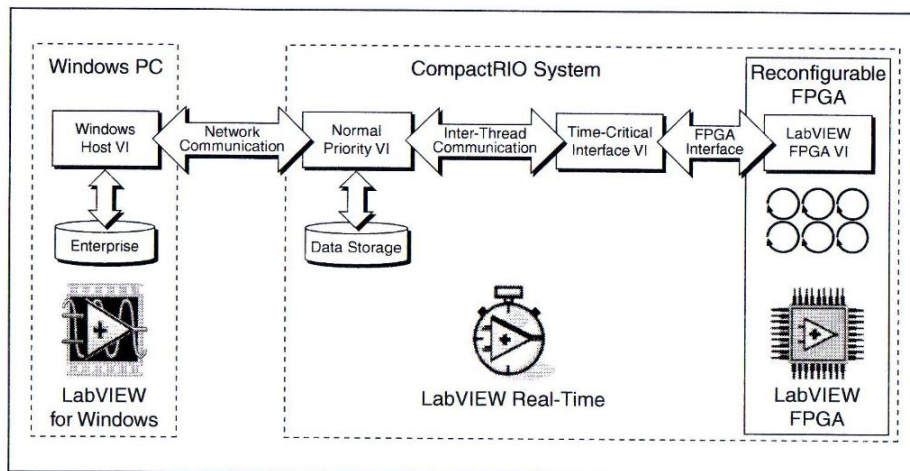


Figura 3.18: Architettura [15]

3.5.1 LabVIEW FPGA

La realizzazione di un VI per FPGA risulta esser molto simile a quella generica per ambiente Windows. Come si vede in figura 3.19 usando LabVIEW FPGA Module si definisce la logica combinatoria del chip FPGA tramite l'esecuzione dei VI.

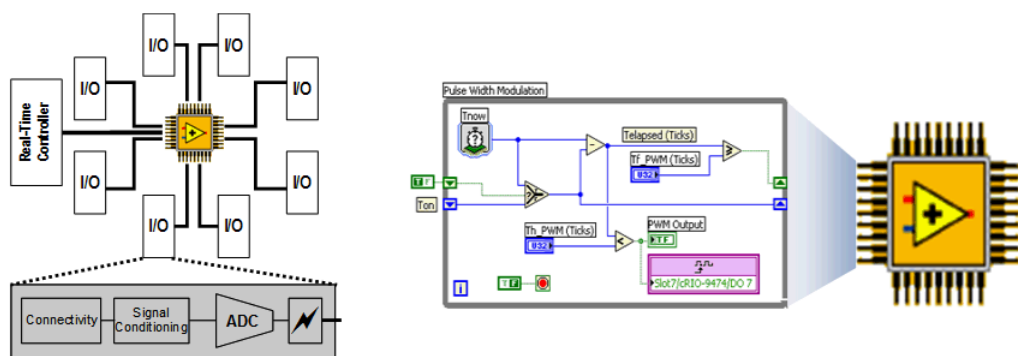


Figura 3.19: Definizione della logica combinatoria FPGA grazie a LabVIEW FPGA Module [16]

Utilizzando NI LabVIEW FPGA Module, il quale mette a disposizione vie di accesso dirette ai dispositivi collegati, è possibile acquisire i segnali provenienti dai dispositivi o generare segnali di controllo.

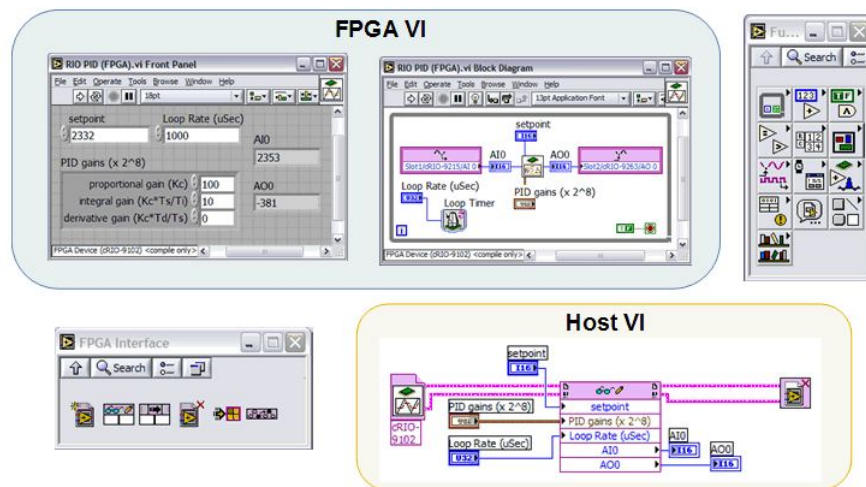


Figura 3.20: Esempio di utilizzo di LabVIEW FPGA Module [16]

Il processo di compilazione viene avviato con la pressione del pulsante RUN. In questo modo viene attivata la conversione del linguaggio LabVIEW Block Diagram in file intermediari.

In figura 3.21 si vede il passaggio dal linguaggio di programmazione LabVIEW allo sviluppo grafico dei chip FPGA nell'hardware "NI" cRIO.

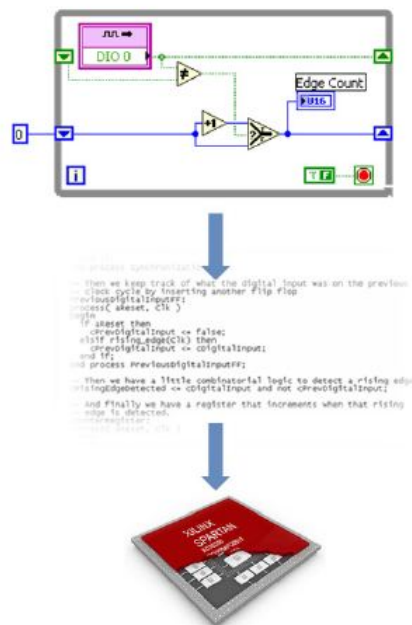


Figura 3.21: LabVIEW FPGA [19]

Successivamente il codice VHDL generato utilizzando il codice VI, viene trasmesso al compilatore Xilinx ISE, il quale sintetizza il codice VHDL dando come risultato il file bitstream che contiene le informazioni di configurazione del gate array. Durante la compilazione vengono inseriti dei vincoli temporali per ottimizzare l'utilizzo delle risorse del chip FPGA. La sequenza è quindi:

- compilazione dei VIs per FPGA;
- ritorno del file bitstream FPGA a LabVIEW;
- il bitstream viene memorizzato in VI.

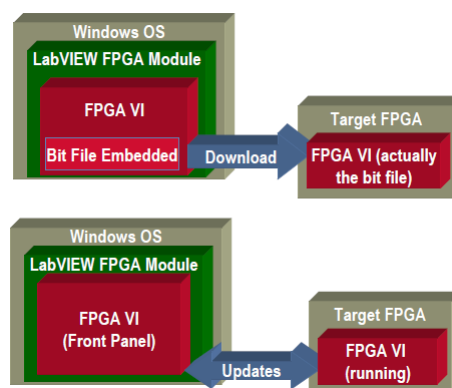


Figura 3.22: LabVIEW FPGA [26]

Risulta possibile implementare sistemi di controllo analogico multi loop PID con velocità oltre i 100 kS/s e sistemi digitali multi loop a velocità di oltre 1 MS/s.

Inoltre il sistema permette di caricare e far girare solo un FPGA VI alla volta, in quanto il successivo VI sovrascrive il precedente.

Non essendoci nessun sistema visivo che possa essere connesso al chip FPGA per controllarne lo stato di funzionamento, risulta necessaria la presenza di un'interfaccia che consenta questa funzione. A tale scopo si usa l'Interactive Front Panel Communication, il quale permette all'host PC di visualizzare la finestra del front panel FPGA VI e al FPGA target di eseguire il FPGA VI block diagram.

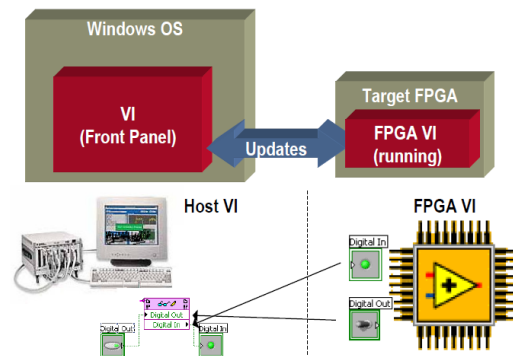


Figura 3.23: Interazione con FPGA tramite host PC based Front Panel [26]

Il front panel ed il block diagram comunicano tra loro, in questo modo risulta semplice e veloce modificare lo stato degli indicatori e dei sistemi di controllo in quanto agendo su uno dei due si va ad intervenire anche sull'altro come si vede in figura 3.24.

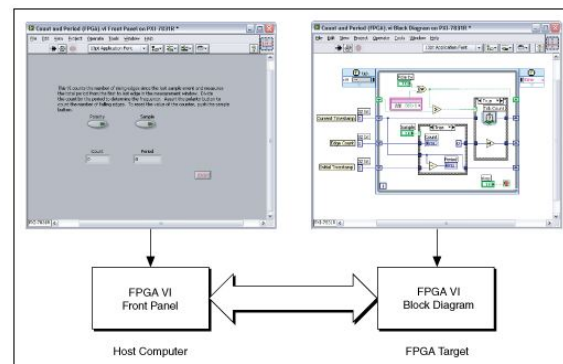


Figura 3.24: Interactive Front Panel Communication [15]

3.5.2 LabVIEW Real-Time

Il controllore Real-Time è un sistema presente nel CompactRIO utilizzabile come un computer industriale per l'esecuzione di VI. L'elaboratore Real-Time comunica con il mondo esterno sia grazie alle schede di input-output sia tramite un'interfaccia di rete. Il controllore Real-Time e il dispositivo FPGA vengono indicati anche come Target della programmazione.

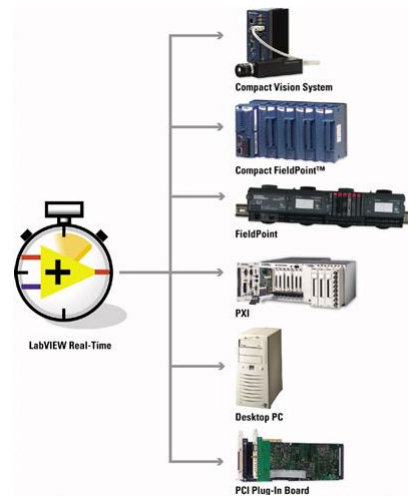


Figura 3.25: LabVIEW Real-Time [16]

Il controllore Real-Time può svolgere i seguenti compiti:

- permette di eseguire operazioni in virgola mobile, agevolando l'implementazione di algoritmi di controllo e di analisi più sofisticati di quelli implementabili su FPGA;
- controlla la tempistica e la sequenza del trasferimento dati;
- esegue più VI.

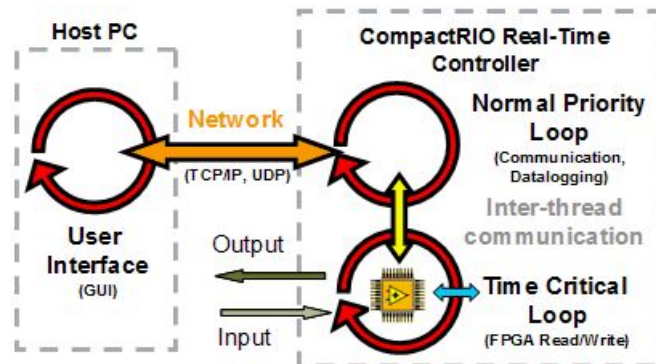


Figura 3.26: Controller Real-Time [16]

3.5.3 LabVIEW per Windows

LabVIEW offre meccanismi integrati per il trasferimento dati dai moduli I/O al chip FPGA e dal chip FPGA al processore Real-Time embedded per analisi,

elaborazioni, data-logging o comunicazioni via rete verso un computer Host (il PC con il quale si programma il CompactRIO, al quale è collegato tramite un cavo Ethernet, come si vede in figura 3.27).

Il PC fornisce l'Interactive Front Panel Communication, grazie al quale LabVIEW e LabVIEW Real-Time attivano l'esecuzione delle differenti sezioni dello stesso VI. Il pannello di controllo serve per mettere nel Front Panel una serie di controlli visibili all'utente ed ad ogni controllo corrisponde un nodo nel block diagram.

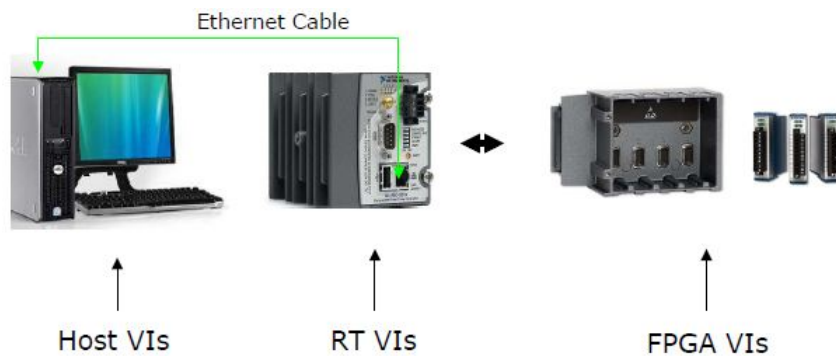


Figura 3.27: Architettura [27]

3.6 Sincronizzazione

Non andando ad intervenire nel sistema si nota che la FPGA, il processore Real-Time ed il PC Windows funzionano in modo asincrono, infatti:

- il clock del PC lavora al millisecondo;
- il clock del processore Real-Time lavora al microsecondo;
- il clock della FPGA lavora al nanosecondo.

Se non si agisce sincronizzando tra loro i tre livelli di architettura si può presentare il rischio di una perdita di informazione e si può incorrere nelle race conditions (situazione in cui la temporizzazione e la sequenza con cui vengono eseguiti i processi produce un esito non desiderato), per evitare ciò risulta conveniente seguire le seguenti condizioni:

- controllare le risorse comuni;
- seguire la sequenza di istruzioni;
- identificare e proteggere le sezioni critiche;

- ridurre al minimo il numero di variabili.

Il controller Real-Time può recuperare i dati da ogni controllo o indicatore posizionato sul pannello frontale dell'applicazione FPGA RIO tramite una funzione di lettura/scrittura FPGA. Comunemente, nella programmazione del CompactRIO vengono utilizzati dei loop multipli. Quando il loop "writer" funziona ad una velocità maggiore di quella del loop "reader" si può incorrere in una perdita dati, per evitare ciò risulta necessario sincronizzare il target e l'Host.

Vengono di seguito introdotti alcuni metodi di bufferizzazione e sincronizzazione:

- traditional handshaking.

Trattasi di un processo che valuta quando il dispositivo è in condizione di acquisire o trasmettere dati. L'applicazione host usa controlli ed indicatori booleani a livello del target per coordinare tra loro le applicazioni host e target.

- Interrupts.

Il RIO FPGA permette la generazione di IRQ (richieste di interrupt) per la sincronizzazione dell'esecuzione Real-Time del software con il RIO FPGA. Gli interrupts possono inviare un segnale di trigger dal target all'applicazione host, notificano l'inizio della lettura, la presenza di un errore ed il termine dell'esecuzione. In media, una singola lettura del registro FPGA richiede circa $10\mu s$ per il controllore cRIO-902/9004, mentre un'interruzione di processo richiede circa $250\mu s$. La massima velocità teorica per l'interruzione risulta quindi approssimata a $4kHz$, quindi è bene produrre impulsi di frequenza inferiore ad $1kHz$ [15].

- Direct memory access (FIFO).

Tecnica di buffering basata su un buffer di tipo FIFO (first in first out). LabVIEW FPGA Module propone diverse tipologie di FIFO:

- Target-Scoped: il trasferimento dati avviene da e verso sezioni della coda in vari VI dello stesso target con una singola FIFO.
- VI-Scoped: il trasferimento dati avviene da e verso molteplici loops senza l'utilizzo di singoli VI.
- DMA: si basa sull'accesso diretto alla memoria per il trasferimento dati.

Capitolo 4

GPIC

La NI Single-Board RIO General-Purpose Inverter Controller (GPIC) è un sistema di acquisizione e controllo.

Il sistema comprende il controller embedded NI sbRIO-9606, un Xilinx Spartan-6 LX45 FPGA e la scheda mezzanine NI 9683, che integra una serie di I/O analogico e digitale. NI 9606 e NI 9683 sono connesse tramite un connettore RIO Mezzanine Card.



Figura 4.1: GPIC [37]

Gli I/O presenti sono i seguenti:

- 16 canali di ingresso Simultaneous analog;
- 8 canali di ingresso Scanned analog;
- 28 canali di ingresso Sourcing digital;
- 14 canali di uscita Half-Bridge digital;

- 24 canali di uscita Sinking digital;
- 4 canali di uscita Relay Control digital;
- 32 canali LVTTL digital I/O.

Al fine di applicare il controllo di corrente all'emulatore dell'inverter, utilizzando come sistema di acquisizione ed elaborazione la GPIC, risulterà sufficiente l'utilizzo di:

- due canali di ingresso Simultaneous analog di figura 4.2;
- due canali di uscita Half-Bridge digital di figura 4.3.

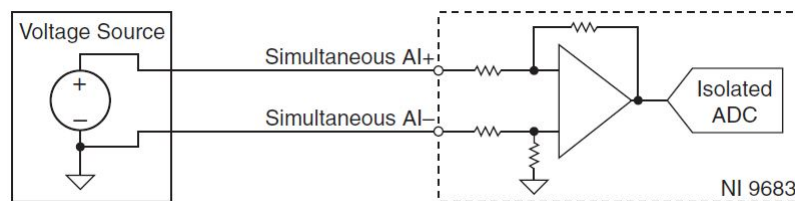


Figura 4.2: Simultaneous [37]

Le specifiche comprendono:

- risoluzione ADC di 12 bits;
- range d'ingresso tipico di $\pm 5V$, $\pm 10V$;
- range di modo comune di $\pm 10V$;

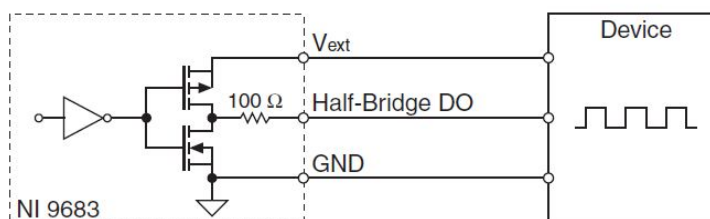


Figura 4.3: Half-Bridge [37]

Le specifiche comprendono:

- impedenza di uscita 100Ω ;

- range di potenza esterna tra i 5 V ed i 30V;
- valori logici:
 - $V_{OH} \approx V_{ext}$;
 - $V_{OL} \approx 0V$;
- minimo periodo di larghezza di impulso di 500ns.

Capitolo 5

Realizzazione della PWM digitale in LabVIEW

La ricerca bibliografica indirizzata allo studio delle diverse tipologie di PWM digitale ha fornito le basi per lo sviluppo in LabVIEW del modulatore stesso.

Il lavoro svolto per la realizzazione della PWM digitale si suddivide nei seguenti punti:

- per prima cosa si sceglie di creare varie tipologie di contatori e di paragonarne la velocità di esecuzione. Per far ciò si sono svolti test sia utilizzando cicli while temporizzati sia non temporizzati. Si è deciso inoltre di svolgere delle prove anche su singole operazioni, per constatare quanto pesassero sulla velocità totale del programma;
- il secondo passo è stato quello di verificare la funzionalità del modulatore contenente i contatori precedentemente realizzati;
- verificata la funzionalità dei programmi, si sono realizzati i subVI contenenti il modulatore.

Di seguito si illustrano i programmi di realizzazione dei PWM di tipo leading-edge e double-edge.

5.1 PWM digitale con contatore leading-edge

Il subVI che realizza il modulatore è composto da un contatore e da un comparatore.

Si realizza il contatore leading-edge implementando un semplice algoritmo che si basa sull'incremento del valore di conteggio ad ogni ciclo fino a raggiungimento del valore massimo scelto pari a 1000. Raggiunto tale valore, il contatore viene reinizializzato a 0.

Il comparatore confronta ad ogni ciclo il valore del duty-cycle con il valore di conteggio.

Su tale subVI si è scelto anche di realizzare un confronto tra il valore di conteggio ed il valore di duty-cycle dimezzato. Da tale confronto vengono emessi degli impulsi il cui ruolo è quello di indicare il luogo di campionamento. Tali impulsi avvengono ad ogni inizio di ciclo ed ogni qual volta il valore di conteggio eguaglia metà del valore del duty-cycle.

5.1.1 Descrizione del programma

In figura 5.1 si mostra il subVI che realizza la modulazione PWM counter-based con contatore leading-edge.

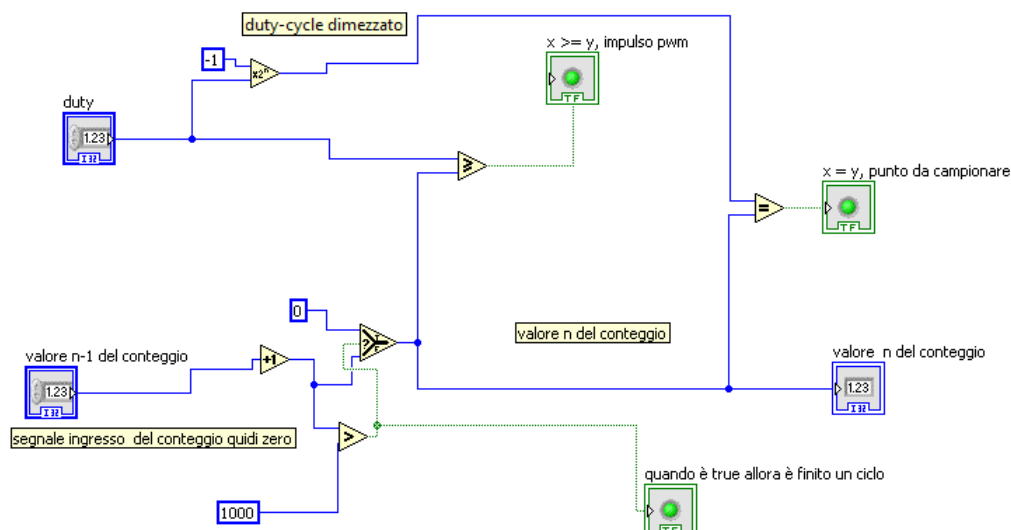


Figura 5.1: SubVI contatore singolo

Il contatore presenta come segnali di ingresso:

- il valore di conteggio prima dell'inizio di ogni ciclo;
- il valore di duty-cycle.

Entrambi questi segnali vengono acquisiti dal VI di figura 5.2 in cui il subVI è inglobato.

Il contatore presenta come segnali di uscita:

- il valore finale di conteggio dopo ogni ciclo;
- il segnale di avviso del termine di ogni ciclo;

- il segnale che indica il momento di campionamento;
- il segnale di impulso PWM.

Il ciclo while temporizzato è presente solo nel VI, in quanto se fosse stato inserito nel subVI l'esecuzione del programma resterebbe bloccata in esso fino alla terminazione del programma del subVI.

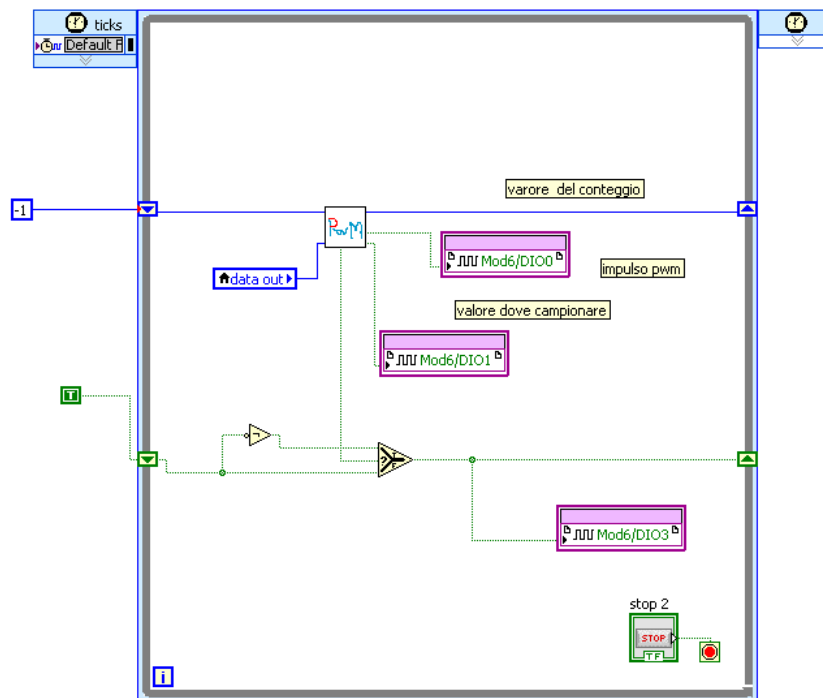


Figura 5.2: VI contatore singolo

I valori -1 e True presenti all'esterno del ciclo temporizzato vengono letti una sola volta all'avvio del programma in quanto il loro ruolo è unicamente quello di inizializzazione.

I moduli digitali Mod6/DIO0, Mod6/DIO1, Mod6/DIO3 coincidono con le uscite digitali del compactRIO e permettono di visualizzare rispettivamente l'impulso PWM, il luogo di campionamento ed il periodo di 1000 conteggi. Tale periodo viene visualizzato al fine di verificare che l'effettivo tempo utilizzato dal sistema per compiere un ciclo, sia pari a quello impostato dal ciclo temporizzato di frequenza pari a 40MHz.

Il secondo ciclo while presente nel VI e mostrato in figura 5.3 contiene un generatore di segnali. Il suo compito è quello di generare un'onda quadra con duty-cycle variabile. Il segnale viene acquisito dal modulatore tramite l'ingresso "data out".

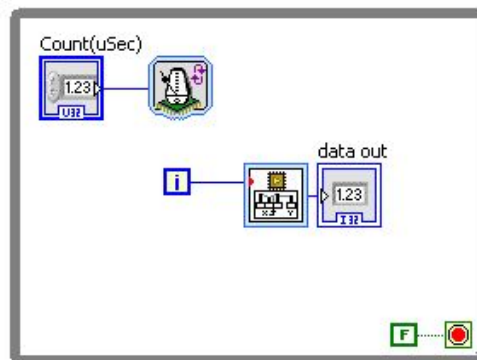


Figura 5.3: Generatore di onde quadre

Tale generatore di forme d'onda non poteva esser inserito direttamente nel ciclo temporizzato in quanto la frequenza di 40MHz risulta eccessiva per tale struttura. Risulta dunque conveniente utilizzare un secondo ciclo while dove inserire il generatore a cui si impone un periodo pari a $25\mu s$.

5.1.2 Test del programma

Il programma precedentemente esposto è stato testato al fine di verificarne la funzionalità e la durata di esecuzione. Si è scelto di far variare il duty-cycle del generatore di onde quadre tra il 30% ed il 50%.

L'esito del test in entrambe le casistiche è presentato nelle figure 5.4 e 5.5.



Figura 5.4: Simulazione con duty pari al 30%

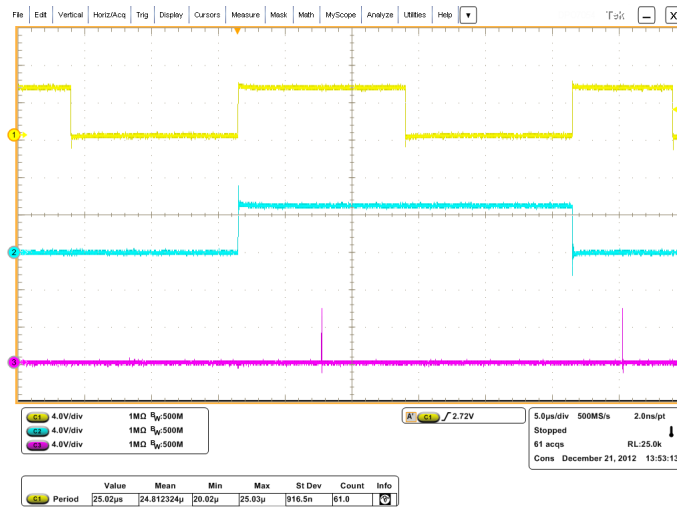


Figura 5.5: Simulazione con duty pari al 50%

L'onda quadra di colore giallo indica l'impulso PWM che inizia con lo start del conteggio e termina all'uguaglianza tra la scala di conteggio ed il duty-cycle.

L'onda quadra di colore azzurro indica il periodo che ci impiega il contatore a compiere 1000 conteggi che risulta pari a $25\mu s$. Si ricava che il periodo di un ciclo dato dal rapporto tra $25\mu s$ e 1000 risulta pari a $25ns$. Si conclude che viene perciò rispettata la tempistica desiderata.

L'impulso di colore fucsia indica invece l'istante di campionamento che per il contatore singolo risulta ottimale a metà dell'impulso PWM.

5.2 Modulazione PWM double-edge

Si è deciso di realizzare due modulatori PWM con contatore di tipo double-edge, in modo tale da verificare se la velocità di esecuzione cambi al variare del programma realizzato. I due modulatori si differenziano per il diverso algoritmo scelto per realizzare la scala di conteggio, il cui numero di scalini è pari a 1000 come nel caso leading-edge. Nel programma verrà però inserito come massimo valore 500 in quanto il contatore realizza 1000 conteggi suddivisi in due scale speculari da 500 scalini l'una.

5.2.1 Primo modulatore double-edge

Il primo modulatore realizzato genera contemporaneamente due scale di conteggio e ne visualizza per la prima metà del tempo la prima e per la seconda metà la successiva.

Un contatore, chiamato "a" incrementa da 0 a 1000 mentre l'altro contatore, chiamato "b", decrementa da tale valore allo 0. Una funzione di selezione consente la visualizzazione della scala "a" fino al raggiungimento del valore 500, per poi visualizzare nella seconda metà i 500 conteggi della scala "b". In questo modo è possibile realizzare il conteggio double-edge con le due scale perfettamente simmetriche.

5.2.1.1 Descrizione del programma

Il programma precedentemente descritto riceve tre ingressi:

- i due valori di conteggio pari a 1000 e 500;
- un valore costante pari a -1.

Il contatore "a" incrementa il valore iniziale di uno ad ogni ciclo fino al raggiungimento del valore 1000. Nel momento in cui il "select 4" riceve l'informazione che il conteggio è giunto a 1000 riavvia il conteggio dal valore 0.

Il contatore "b" decrementa il valore iniziale da 1001 fino a 0. Nel momento in cui il "select 1" viene informato che il conteggio è giunto a 0, trasmette in uscita al ciclo il valore in uscita al "select 2". Tale valore è pari a 1 nel caso in cui il valore del contatore "a" è inferiore a 1000 (in questo modo viene mantenuto il contatore "b" in uno stato di attesa), altrimenti viene restituito il valore 1000 e dunque il conteggio ricomincia.

La visualizzazione della scala simmetrica desiderata viene garantita dal "select 3". Quest'ultimo permette la visualizzazione della scala "a" nell'intervallo $[0 - 499]$ e della scala "b" nell'intervallo $[500 - 1]$.

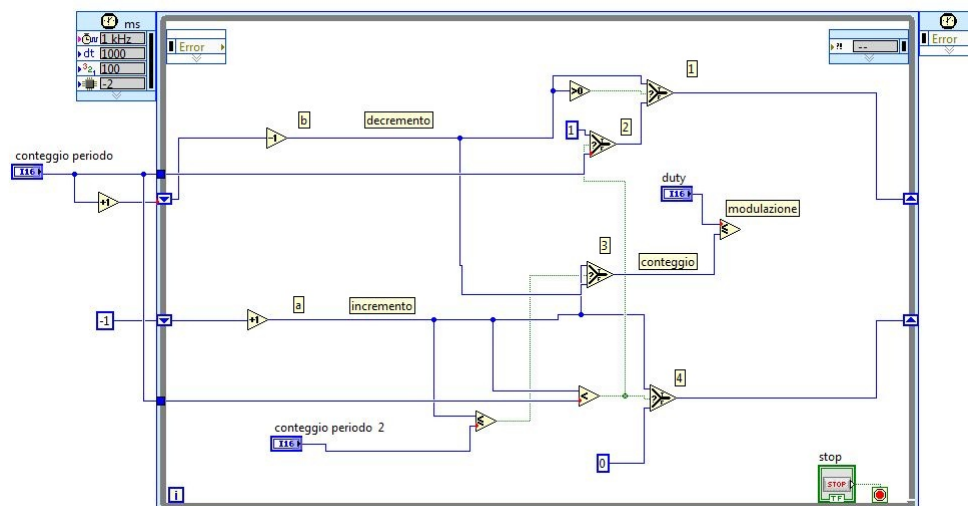


Figura 5.6: Contatore doppio

5.2.2 Secondo modulatore double-edge

Il secondo modulatore realizzato esegue la scelta di incrementare o decrementare il conteggio, basandosi sul valore che assume una linea booleana che viene aggiornata ad ogni ciclo assieme al conteggio.

5.2.2.1 Descrizione del programma

Il subVI che realizza il modulatore presenta come segnali di ingresso:

- il valore di conteggio all'inizio di ogni ciclo;
- il valore di duty-cycle;
- il valore booleano all'inizio di ogni ciclo.

Come nel caso del subVI del leading-edge, questi segnali vengono acquisiti dal VI in cui il subVI è inglobato.

Il subVI presenta come segnali di uscita:

- il valore di conteggio al termine di ogni ciclo;
- il segnale di avviso del termine di ogni semi-ciclo;
- il segnale che indica il momento di campionamento;
- il segnale di impulso PWM.

In figura 5.7 si mostra il programma di realizzazione del subVI del modulatore.

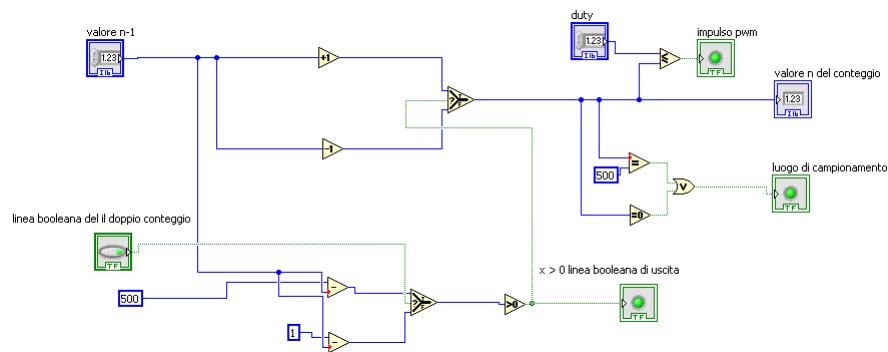


Figura 5.7: SubVI del contatore doppio

Il programma acquisisce il valore di conteggio iniziale e contemporaneamente lo incrementa e decrementa ad ogni ciclo. Il valore che viene posto come segnale di uscita del ciclo sarà pari a quello incrementato se la linea booleana presenta il True, viceversa se presenta il False.

L'informazione trasportata dalla linea booleana viene decisa dal valore in uscita al "select" (se maggiore di zero True altrimenti False). Tale "select" utilizza come discriminante il valore della linea booleana stessa. Emette il valore pari alla differenza tra 500 ed il valore iniziale di conteggio nel caso in cui sia True, ed il valore pari alla differenza tra 1 ed il valore iniziale di conteggio nel caso in cui sia False. In questo modo l'informazione della linea booleana si inverte ogni volta che il conteggio giunge a 0 oppure a 500 comandando quindi l'incremento o il decremento del contatore.

L'impulso PWM viene generato quando il valore del segnale duty-cycle è inferiore a quello di conteggio, mentre l'impulso che indica il luogo di campionamento avviene quando il valore di conteggio è pari a 0 e a 500.

In figura 5.8 si vede il VI che incorpora il SubVI precedentemente descritto. I moduli digitali Mod6/DIO0, Mod6/DIO1, Mod6/DIO3 permettono di visualizzare l'impulso PWM, il luogo di campionamento e l'indicazione di raggiungimento del valore 500 nella scala di conteggio.

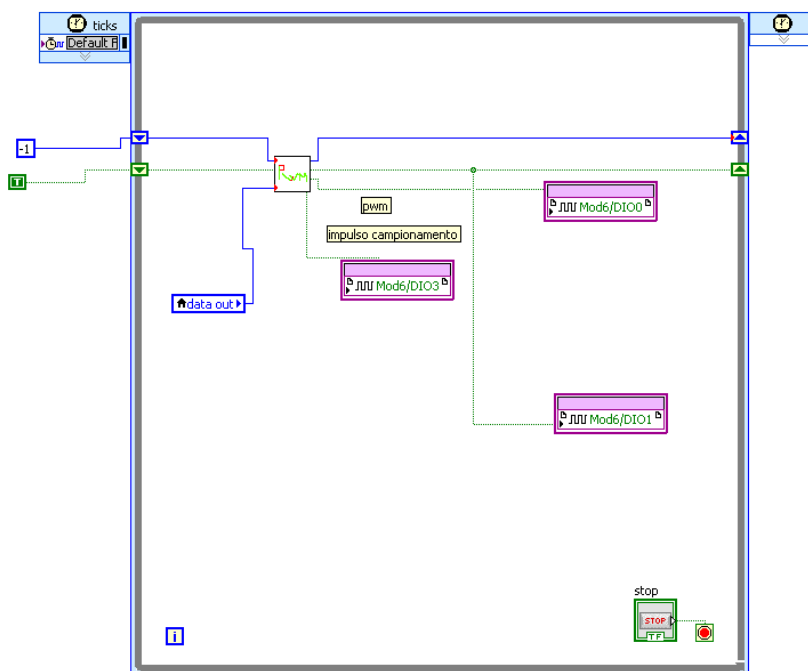


Figura 5.8: VI del contatore doppio

5.2.2.2 Test del programma

Il programma precedentemente descritto è stato testato con lo stesso generatore di onde quadre utilizzato per il modulatore con contatore single-Edge di figura 5.3.

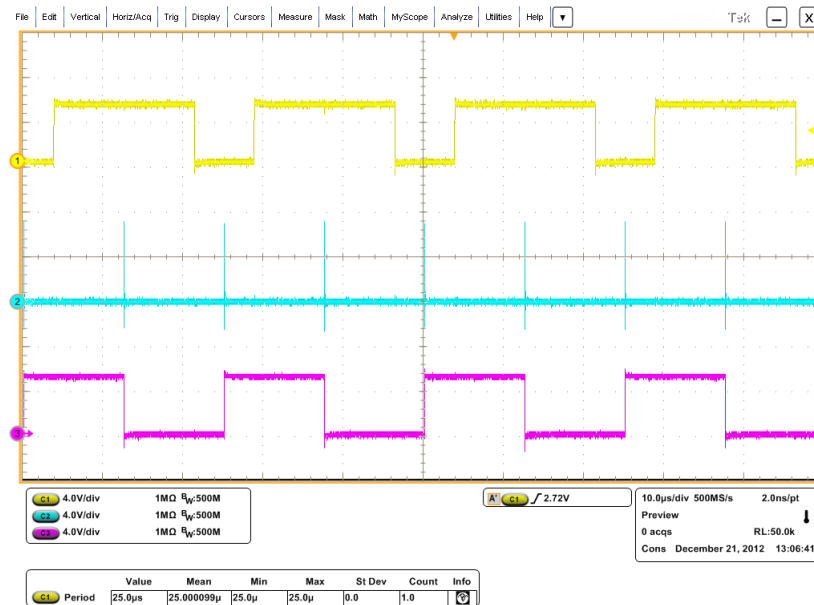


Figura 5.9: VI del contatore doppio

L'onda quadra di colore giallo indica l'impulso PWM .

L'onda quadra di colore fucsia indica un semi periodo, cioè il tempo che il contatore impiega a compiere 500 conteggi. Il periodo corrispondente a 1000 conteggi risulta pari a $25\mu s$ e dunque il periodo di un ciclo dato dal rapporto tra $25\mu s$ e 1000 vale 25ns.

L'impulso di colore azzurro indica invece l'istante di campionamento (che per il contatore doppio risulta ottimale a metà dell'impulso PWM) ed all'inizio di ogni periodo di conteggio.

5.2.3 Test sui contatori double-edge

Di seguito sono riportati i programmi creati per la realizzazione del contatore double-edge. Il fine è quello di rilevare la reale velocità del programma al variare delle funzioni utilizzate. Per compiere tali test dunque non viene utilizzato il ciclo temporizzato.

La tipologia di figura 5.10 realizza il contatore double-edge creando due scale di conteggio e visualizzandone solo una parte di entrambe. Nello specifico questa versione presenta l'utilizzo di due ingressi esterni riferiti al periodo di realizzazione di 1000 scalini ed al periodo di 500 scalini.

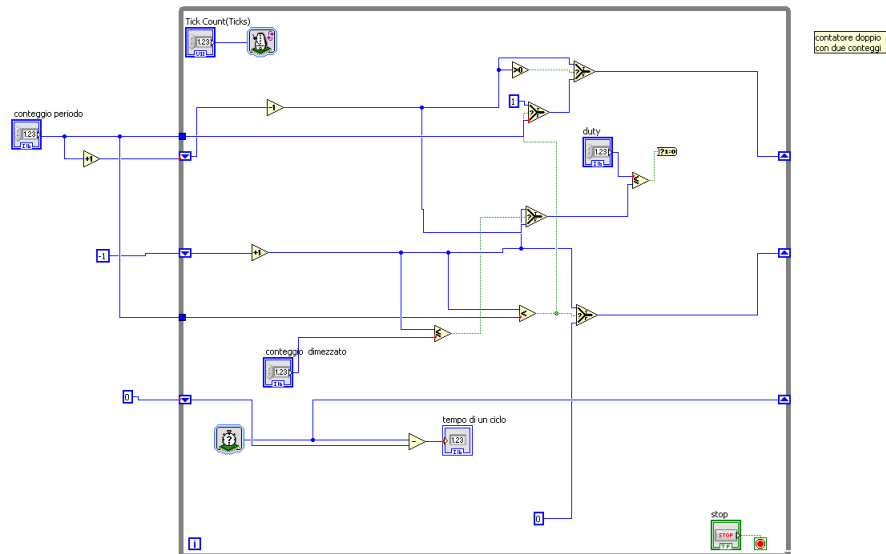


Figura 5.10: VI del contatore doppio

Si rileva che il tempo impiegato per un ciclo vale 351 ms.

In questa seconda versione si utilizza un solo ingresso esterno riferito al massimo valore di conteggio, mentre per realizzare il conteggio di 500 scalini si sceglie il blocco $x \cdot 2^n$.

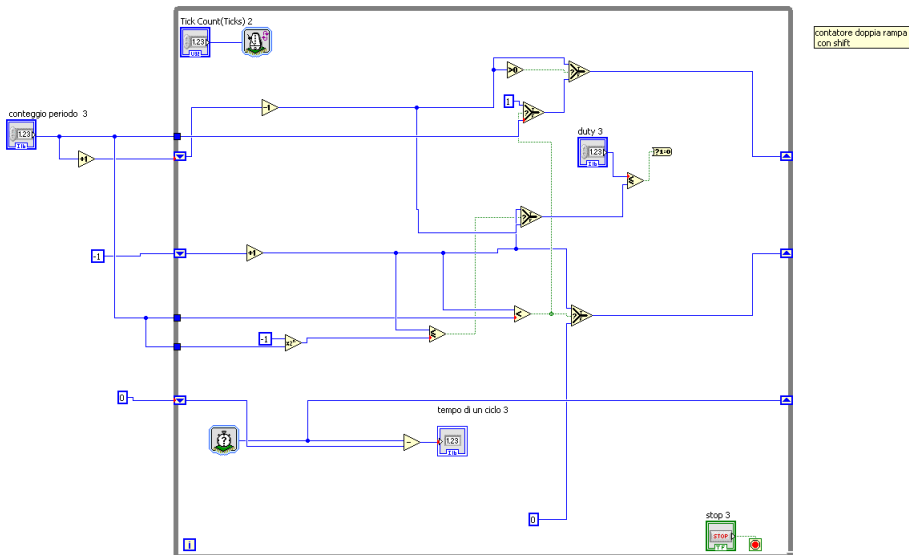


Figura 5.11: VI del contatore doppio

Si rileva che il tempo impiegato per un ciclo è pari a 351 ms.

Nella terza versione si impiega un divisore per realizzare il periodo di conteggio di 500 scalini.

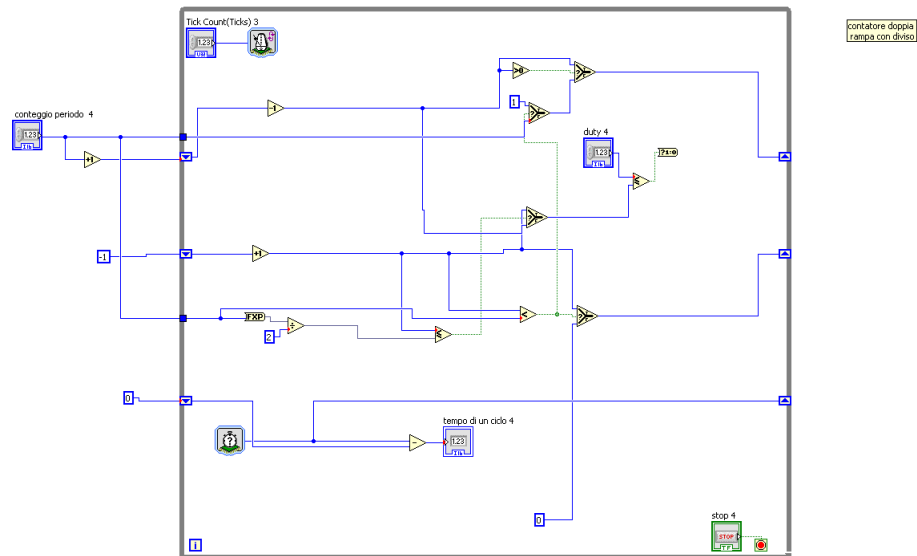


Figura 5.12: VI del contatore doppio

Si rileva che il tempo impiegato per un ciclo è pari a 351 ms. L'ultima tipologia è quella relativa al contatore con linea booleana.

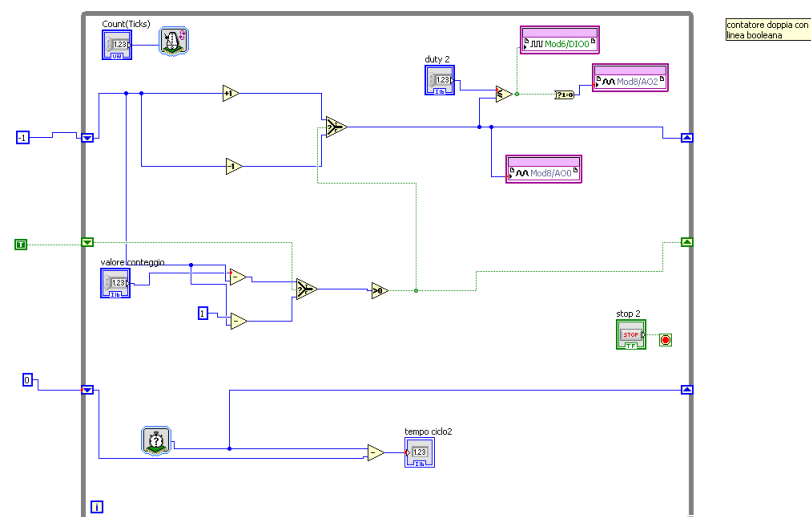


Figura 5.13: VI del contatore doppio

Si rileva che il tempo impiegato per un ciclo è pari a 243 ms.

Ciascuno dei cicli sopra citati, utilizzando il ciclo temporizzato a 40MHz, rispetta tale condizione di tempistica impiegando 25ns a ciclo.

5.3 Test sulla velocità dei contatori al variare del numero di conteggi

I contatori precedentemente descritti sono stati testati anche per un diverso numero di conteggi per verificare che il periodo di un ciclo restasse di 25ns per un ciclo temporizzato di 40MHz. In figura 5.14 viene riportato l'esempio con il numero di conteggi pari a 100.

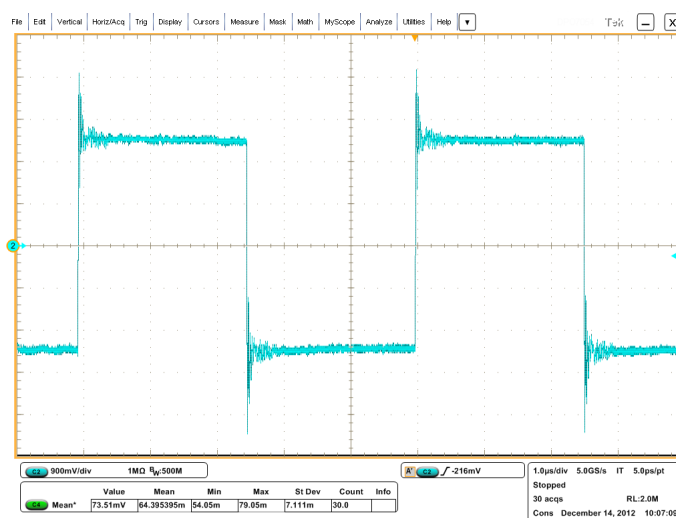


Figura 5.14: Periodo di 100 cicli

In figura 5.15 è mostrato il periodo di un singolo ciclo a 40MHz. Ovviamente l'andamento non è ben definito come in 5.14, ma si riesce ugualmente ad intuire che il periodo è di 25ns.

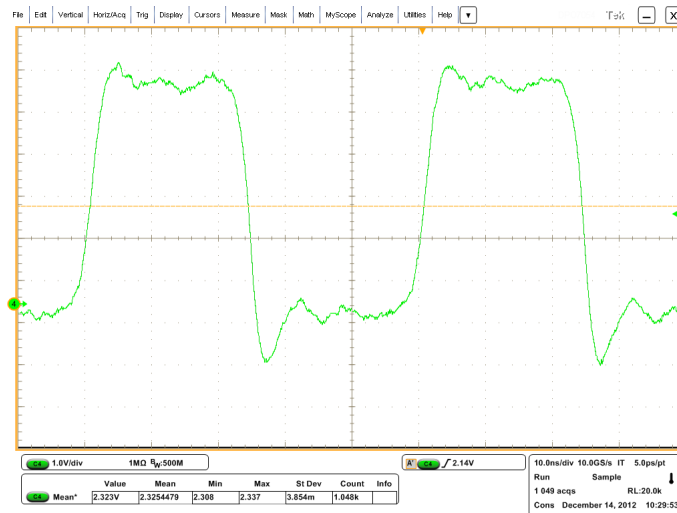


Figura 5.15: Periodo di 1 ciclo

5.4 Test generali

Dopo aver rilevato le tempistiche dei vari contatori con diversi divisori vengono ora analizzate le velocità di alcune operazioni.

Il tempo di un ciclo, ovviamente non temporizzato, per i tre programmi di figura 5.16, 5.17 e 5.18 riferiti alle tre tipologie di divisione, è pari a 75ms per il ciclo con "divisore 1", 6ms per il "divisore 2" e 40ms per il "divisore 3".

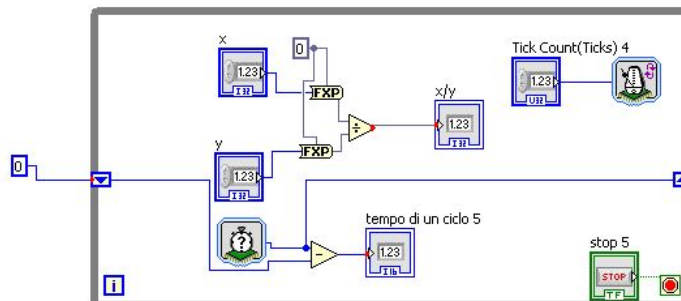


Figura 5.16: Divisore 1

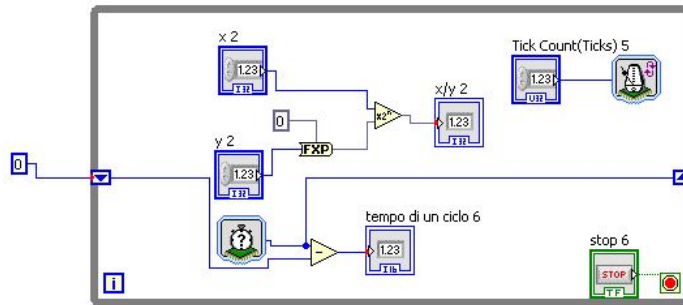


Figura 5.17: Divisore 2

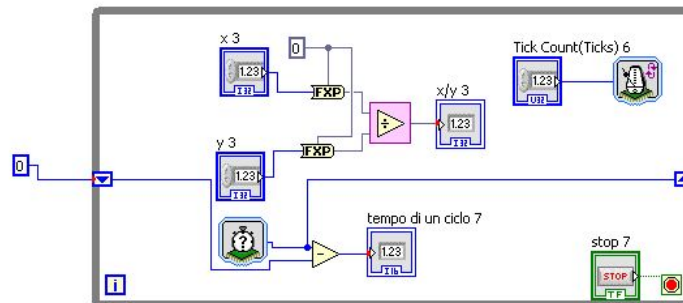


Figura 5.18: Divisore 3

Al fine di capire quanto incide un invertitore, sulla tempistica totale di un programma con ciclo non temporizzato, si esamina la velocità del programma di figura 5.19 in cui è presente una cascata di tre invertitori. Si rileva che il tempo che intercorre tra la visualizzazione dell'ingresso del primo invertitore e del terzo è pari a 75ns. Risulta dunque evidente che ogni singolo invertitore pesa 25ns sulla velocità complessiva.

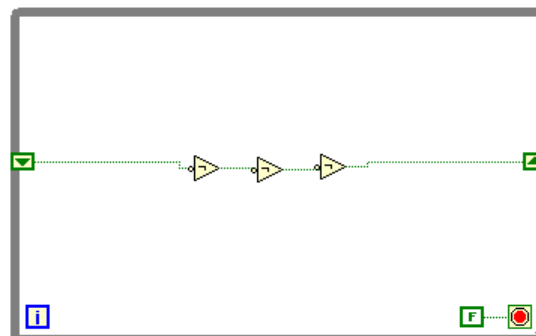


Figura 5.19: VI del contatore doppio

Capitolo 6

Realizzazione del regolatore PI

Il regolatore scelto per la realizzazione del controllo di corrente della scheda simulante il funzionamento dell'inverter è stato di tipo PI (proporzionale integrale).

Il regolatore, come si vede in figura 6.1, acquisisce il segnale di errore $e[k]$ dato dalla differenza tra il segnale di riferimento ed il segnale retroazionato ed emette il segnale modulante $m[k]$. Il segnale digitale $m[k]$ sarà quindi utilizzato dal modulatore PWM come segnale da confrontare con la portante.

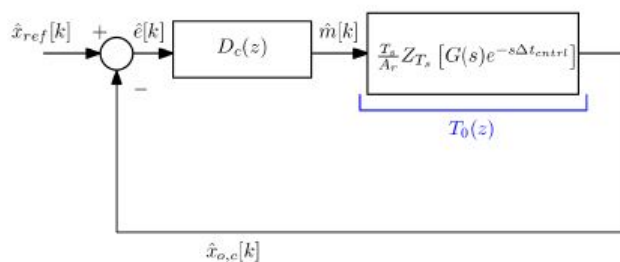


Figura 6.1: Regolatore [38]

La natura digitale del segnale $m[k]$ comporta del tempo di ritardo tra l'aggiornamento di $m[k]$ e del segnale in uscita al modulatore PWM. Tale ritardo, definito Δt_{PWM} varia a seconda della portante utilizzata come in tabella 6.2.

Modulazione	$A(f)$	Δt_{PWM}
Trailing Edge	1	$D \cdot T_s$
Leading Edge	1	$(1-D) \cdot T_s$
Triangolare	$\cos(\pi f D T_s)$	$T_s / 2$

Figura 6.2: Tabella [38]

La funzione di trasferimento del modulatore risulta dipendente da tale ritardo ed è definita come:

$$G_{PWM}(f) = \frac{1}{A} e^{-j2\pi f \Delta t_{PWM}} \quad (6.1)$$

Lo sfasamento risulta quindi dato da:

$$\Delta \varphi_{PWM} = \arg(G_{PWM}(f)) \quad (6.2)$$

Lo sfasamento $\Delta \varphi_{PWM}$ dovuto dal ritardo Δt_{PWM} incide sul margine di fase e quindi su una delle specifiche di progetto del regolatore. Sapendo che Δt_{PWM} a sua volta dipende dalla portante, si sceglie per il progetto la portante con contatore double-edge. In questo modo il punto medio della forma d'onda del segnale da campionare è fisso rispetto alla portante stessa ed è dunque possibile ottenere il valore medio del segnale indipendentemente dal duty-cycle.

6.1 Dimensionamento del K_I e del K_P

Il dimensionamento dei valori K_I e K_P è stato effettuato studiando la funzione di trasferimento del controllo di corrente dell'inverter.

$$G(s) = G_{PWM}(s) \cdot G_{INV}(s) \cdot G_{PI}(s) \quad (6.3)$$

Dove:

$$G_{PWM}(s) = \frac{1}{A} \quad (6.4)$$

Dove con A si indica l'ampiezza della portante.

$$G_{INV}(s) = \frac{2V_{AB}}{sRC} \quad (6.5)$$

$$G_{PI}(s) = K_P + \frac{K_I}{s} = K_I \frac{1 + s\tau}{s} \quad (6.6)$$

Dove:

$$\tau = \frac{K_P}{K_I} \quad (6.7)$$

Le specifiche riguardano:

- la banda di controllo scelta in un primo momento pari ad $\frac{f_{sw}}{10}$ e poi a $\frac{f_{sw}}{8}$;
- il margine di fase scelto pari ad 60° a cui si aggiungono 20° dovuti al ritardo dell'aggiornamento di m[k] e del segnale PWM;
- il valore di V_{AB} pari a 12V;
- l'ampiezza della portante A pari a 500;
- la frequenza f_{sw} pari a 40KHz;

Approssimando $K_I \ll \omega_{cros} \cdot K_P$, dove ω_{cros} è pari a $2\pi \frac{f_{sw}}{10}$ nel primo caso ed a $2\pi \frac{f_{sw}}{8}$ nel secondo, si ottiene:

$$|G(j\omega_{cros})| = \left| \frac{1}{A} \cdot \frac{2V_{AB}}{j\omega_{cros} \cdot RC} \cdot K_P \right| \quad (6.8)$$

Il cui modulo è pari ad 1 essendo ω_{cros} la frequenza di attraversamento del diagramma di Bode. Dunque si ottiene:

$$K_P = A \cdot \frac{\omega_{cros}}{2V_{AB}} \cdot RC \quad (6.9)$$

mentre K_I si ottiene dalla fase di $G(j\omega_{cros})$:

$$K_I = \frac{\omega_{cros} \cdot K_P}{\text{tg}(-90 + 80 + 90)} \quad (6.10)$$

I valori di K_I e K_P ottenuti con la banda pari a $\frac{f_{sw}}{10}$ e poi a $\frac{f_{sw}}{8}$ sono rispettivamente di:

- $K_I = 5 \cdot 10^5$; $K_P = 115, 2$.
- $K_I = 7, 98 \cdot 10^5$; $K_P = 144$.

Trattandosi di un regolatore digitale il valore di K_P nel discreto è lo stesso del caso continuo. Contrariamente, il valore di K_I nel discreto è ottenuto moltiplicando il valore di K_I nel continuo per il periodo T_K di campionamento pari ad 80kHz.

- $K_{I_{discreto}} = 6$.
- $K_{I_{discreto}} = 10$.

La correttezza di tali valori è stata verificata anche mediante simulazione Matlab.

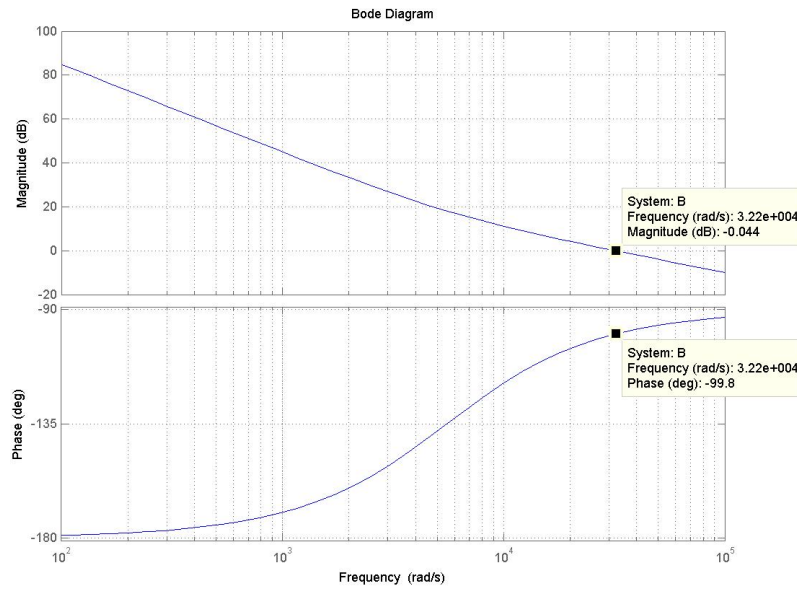


Figura 6.3: Diagramma di bode con regolatore e con la banda pari a $\frac{f_{sw}}{8}$

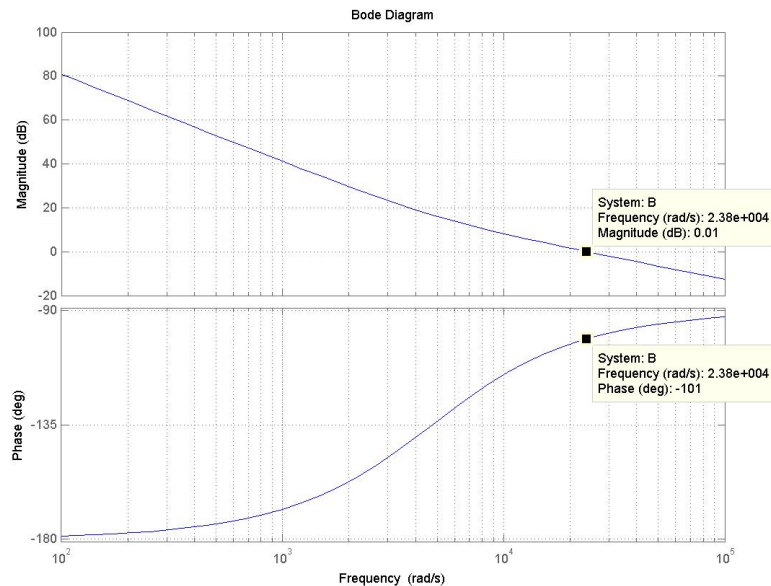


Figura 6.4: Diagramma di bode con regolatore e con la banda pari a $\frac{f_{sw}}{10}$

Senza il regolatore il diagramma di Bode si presentava nel seguente modo.

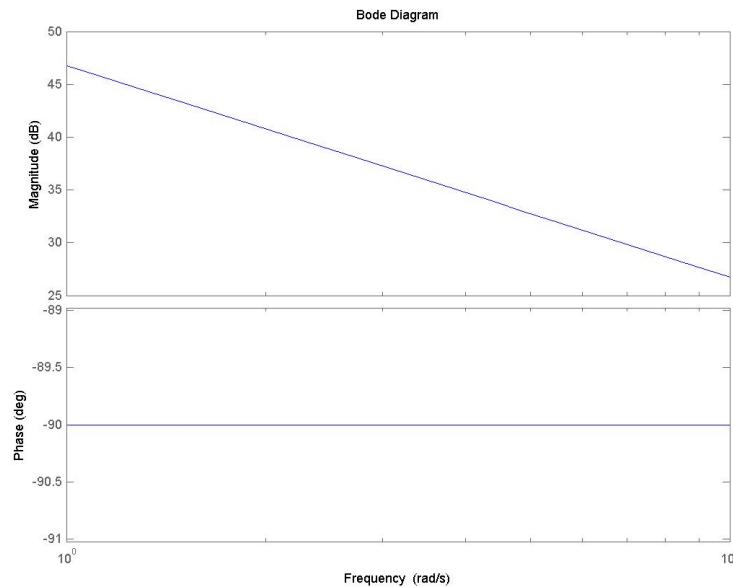


Figura 6.5: Diagramma di bode senza regolatore

6.2 Tecnica anti-wind-up

Utilizzando un regolatore PI possono verificarsi problemi legati alla saturazione delle uscite. Nel momento in cui l'errore mantiene lo stesso segno per un determinato periodo, il segnale in uscita al regolatore (in questo caso la modulante) cresce in modulo anche se ha raggiunto il valore massimo entro cui può variare. In questo caso l'intervallo di valori entro cui può variare è pari all'ampiezza della portante (-250 e 250). Quando la modulante raggiunge quindi un valore superiore a quello limite si osserva un suo ritardo nel rispondere alla variazione di segno del segnale di ingresso al regolatore. Tale ritardo è dovuto all'integratore del PI che continua ad integrare l'errore anche quando sta saturando. Per evitare tale problematica si può intervenire mediante tecniche anti-wind-up.

Tale tecnica limita l'uscita del regolatore in modo tale che il valore della modulante sia minore del valore che può assumere la portante. Si è deciso di limitare la parte proporzionale del regolatore tra -250 e +250 (valore picco-picco della portante). Mentre il valore limite della parte integrale dipende dalla parte proporzionale come nell'equazione 6.11:

$$|L(k)| = Y_{max} - |k_p e(k)| \quad (6.11)$$

dove Y_{max} è pari a 250.

In figura 6.6 è mostrata la tecnica Anti-wind-up.

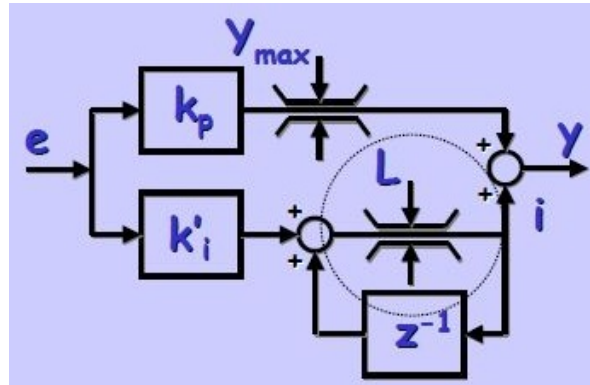


Figura 6.6: Anti-wind-up [10]

Per mostrare l'efficacia di tale tecnica, si mostra in figura 6.7 la risposta al gradino unitario di un sistema con regolatore PI con (in verde) e senza (in blu) anti-wind-up.

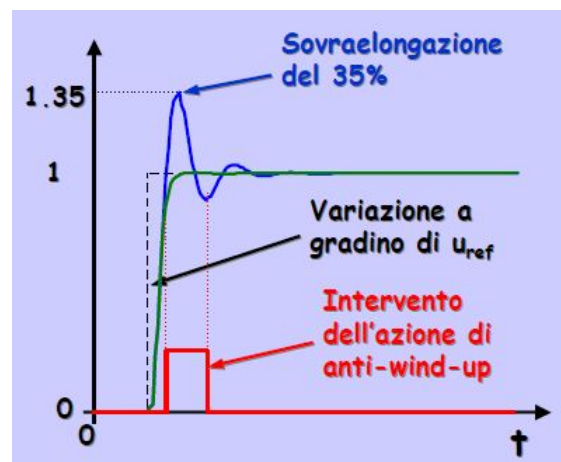


Figura 6.7: Anti-wind-up [10]

L'azione della tecnica anti-wind-up elimina dunque la sovralongazione.

6.3 Realizzazione del subVI del regolatore PI

Il subVI contenente il regolatore si basa sul calcolo del valore di $m[k]$, dove per k si definisce il k -esimo istante di campionamento. Il regolatore implementato è di tipo proporzionale integrale dunque il valore di $m[k]$ sarà:

$$m[k] = m_I[k] + m_P[k] \quad (6.12)$$

dove $m_I[k]$ è riferito all'integrale ed $m_P[k]$ al proporzionale.

$$m_P[k] = K_P \cdot e[k] \quad (6.13)$$

$$m_I[k] = T_K \cdot K_I \cdot e[k] + m_I[k - 1] \quad (6.14)$$

In figura 6.8 si mostra la struttura realizzata avente come segnali di ingresso:

- il valore di $e[k]$, dato dalla differenza tra il segnale di riferimento ed il segnale retroazionato;
- il valore $m_I[k - 1]$.

Mentre come segnali di uscita:

- il valore $m[k]$, la modulante;
- il valore $m_I[k]$.

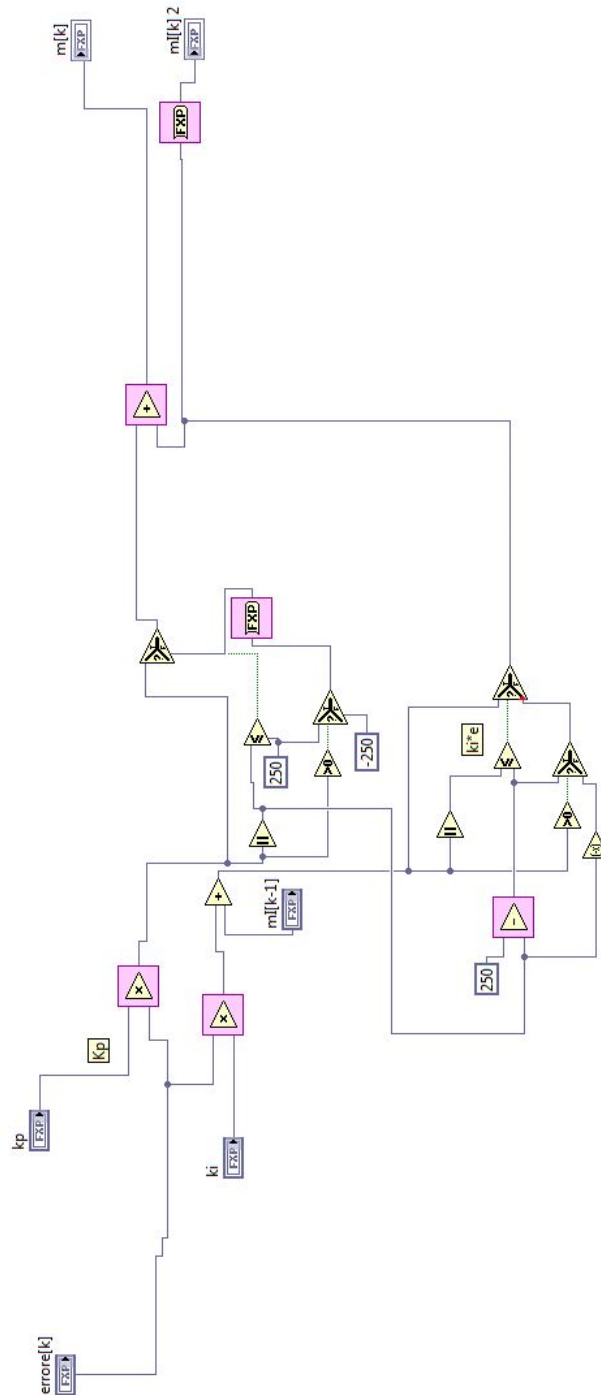


Figura 6.8: Regolatore PI con anti-wind-up

Per non appesantire troppo il VI, si è deciso di realizzare una tecnica di anti-wind-up più semplice di quella illustrata in precedenza e consistente nella semplice limitazione ad un livello predefinito della parte integrale del regolatore.

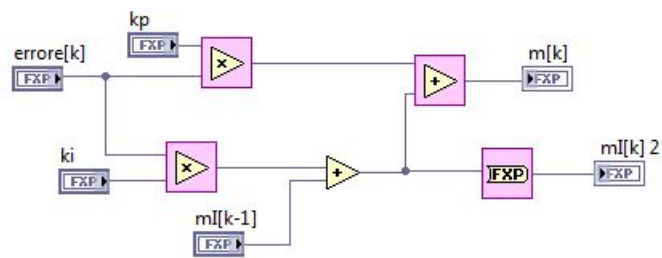


Figura 6.9: Regolatore PI utilizzato

Capitolo 7

Realizzazione del controllo di corrente

Conclusa la trattazione del modulatore PWM e del regolatore PI digitale, si illustra il controllo di corrente che si è deciso di effettuare.

Il controllo si effettua comparando il valore di corrente in uscita dall'inverter con un valore di riferimento. L'errore risultante da tale sottrazione viene processato dal regolatore PI così da ottenere la modulante che verrà paragonata alla portante. L'impulso di uscita al PWM ed il suo negato comandano quindi i driver che controllano a loro volta le gambe dell'inverter.

Nello specifico, nel controllo in questione, il segnale di riferimento viene prelevato dalla schedina simulante la tensione di rete ad una frequenza di 50Hz. Tale segnale viene in seguito moltiplicato per un valore variabile in modo tale da poterne modificare l'ampiezza. Il segnale ottenuto viene quindi paragonato con la corrente in uscita all'emulatore così da ottenere il valore $e[k]$ da immettere nel regolatore.

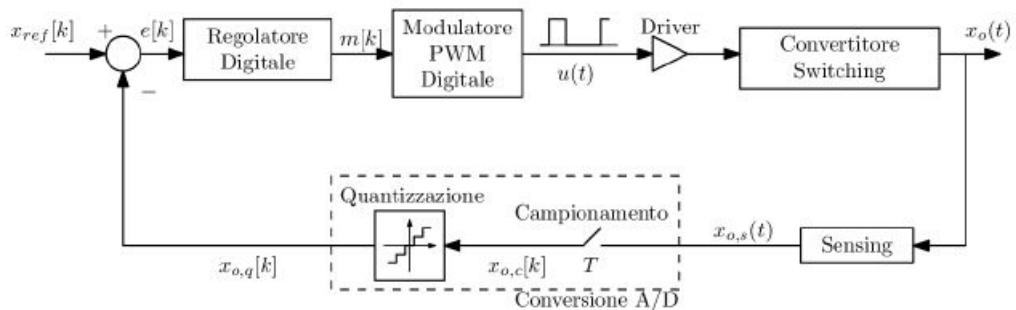


Figura 7.1: Controllo [38]

7.1 Realizzazione del sistema di acquisizione

L'acquisizione dati avviene interfacciando il convertitore A/D con il trasduttore.

Il sistema di acquisizione funziona in modo tale da garantire che l'acquisizione del segnale avvenga sempre nello stesso istante e cioè l'istante di campionamento.

Per far ciò si è deciso di utilizzare una "Flat sequence structure". Essa permette di comandare un ordine di priorità nell'esecuzione di una successione di comandi, in modo tale che fino a non avvenuta conclusione del comando precedente il successivo non può essere eseguito. In questo caso è di nostro interesse garantire che l'acquisizione del segnale avvenga sempre nell'istante in cui la portante raggiunge il suo valore minimo e massimo. Per far ciò, si utilizza la struttura precedentemente citata che comanda l'avvio dell'acquisizione solo al raggiungimento dell'istante di campionamento (che avviene ad una frequenza di 80kHz). Fino all'arrivo del successivo istante di campionamento l'acquisitore ha dunque l'ordine di restare in attesa.

Estratto e convertito il segnale, esso è ora pronto per essere trasmesso al regolatore. Si intuisce che come comando di acquisizione non possono essere utilizzati gli impulsi di campionamento generati nel ciclo temporizzato da 40MHz, in quanto essi non possono essere visibili nel ciclo di acquisizione. Si decide dunque di rilevare i fronti di salita e di discesa del segnale riferito al semi periodo della portante triangolare (il segnale di colore fucsia presente in figura 5.9), dato che anch'essi si riferiscono agli istanti di campionamento utili al fine di prelevare il valor medio del segnale.

Il "select" presente nel VI è utilizzato al fine di consentire all'utente di poter variare manualmente il valore di riferimento oppure di utilizzare un riferimento esterno acquisito da "RMC/Simultaneous/AI1". Il compito di "RMC/Simultaneous/AI0" è invece quello di acquisire la corrente in uscita all'emulatore.

In figura 7.3 si vede che durante l'esecuzione del programma è concesso all'utente di:

- variare i parametri di K_I e di K_P ;
- scegliere l'utilizzo di un riferimento che può essere variato manualmente dall'utente;
- scegliere di acquisire esternamente una sinusoide la cui ampiezza può essere variata agendo sul valore di $G_{equivalente}$;
- decidere di impostare un valore di modulante costante.

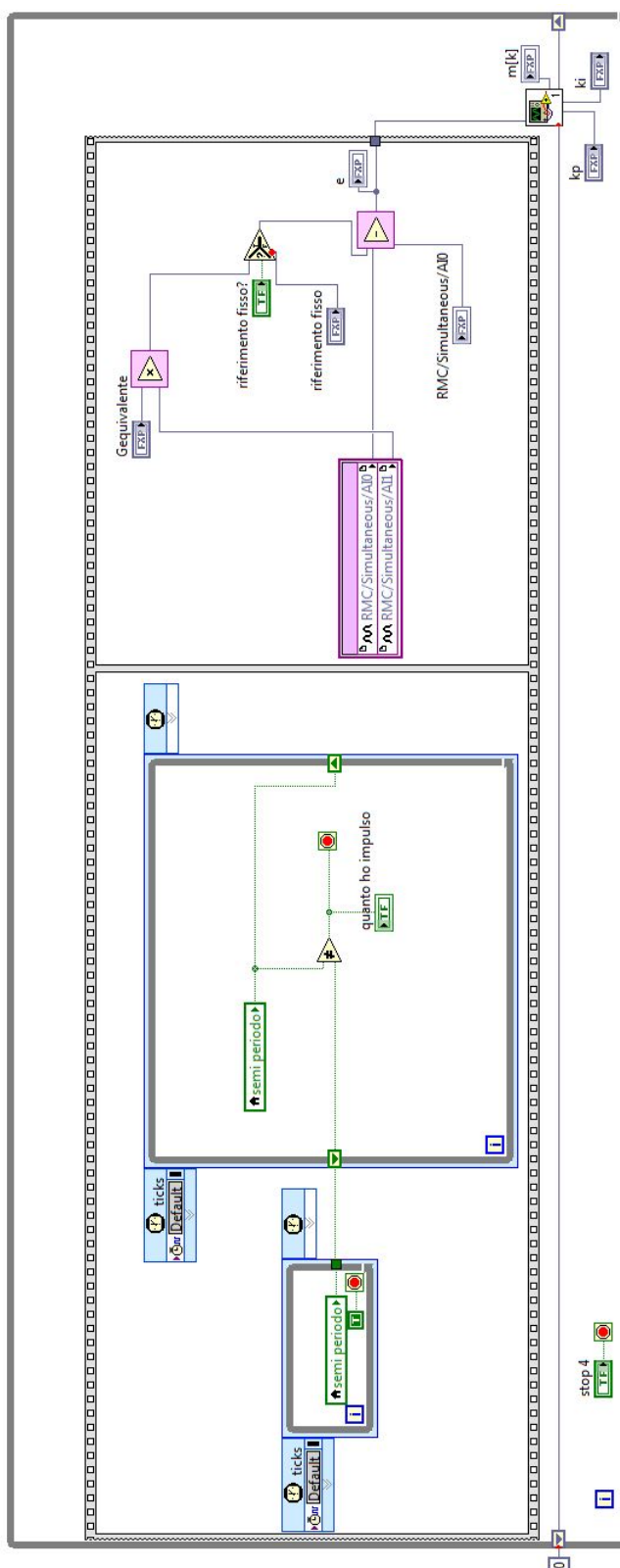


Figura 7.2: Acquisizione

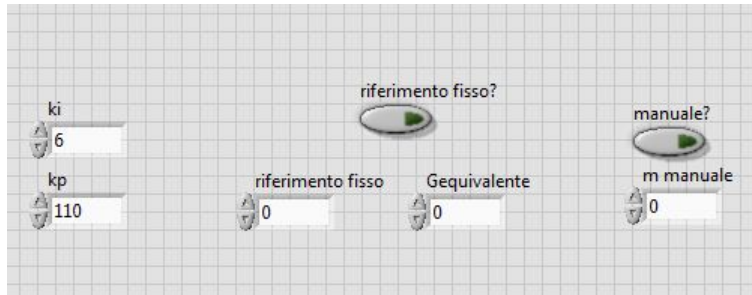


Figura 7.3: Front Panel

7.2 Realizzazione del sistema di controllo

Il controllo di corrente realizzato con LabView è esposto in figura 7.4.

Il VI acquisisce come dato esterno al suo ciclo la modulante e grazie al sub-VI del modulatore PWM realizza i segnali di comando delle gambe dell'inverter. Tali comandi sono trasferiti alla GPIC grazie a "RMC/Half-Bridge/DO0" e "RMC/Half-Bridge/DO1".

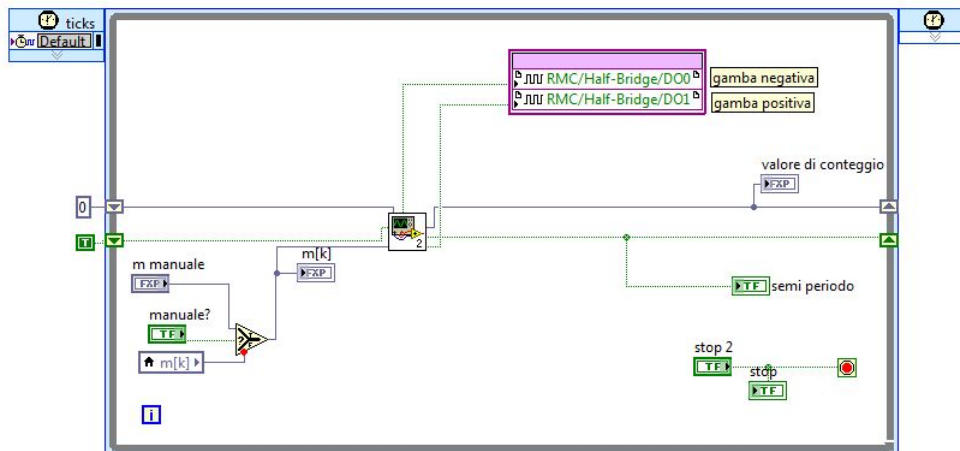


Figura 7.4: Controllo

Capitolo 8

Emulatore di inverter

L'emulatore di inverter ha il compito di simulare il comportamento di un vero inverter a ponte intero monofase connesso alla rete. In figura 8.1 si presenta lo schema a blocchi di un convertitore connesso alla rete per impianti fotovoltaici (un esempio di applicazione per gli inverter connessi alla rete).

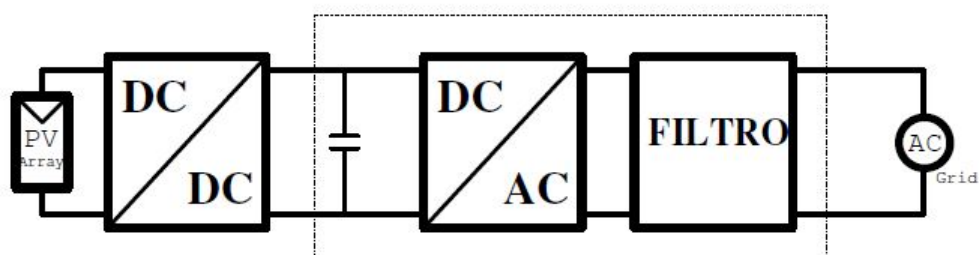


Figura 8.1: Schema a blocchi di un convertitore connesso alla rete [39]

Il capitolo è suddiviso in due parti:

- la prima spiega il funzionamento dell'inverter e la modulazione PWM;
- la seconda è incentrata sull'emulatore e sulle caratteristiche che dovrà avere l'inverter da emulare.

8.1 Funzionamento inverter

Gli inverter sono dispositivi che permettono la conversione da una tensione continua ad alternata, generalmente sinusoidale.

Lo schema di un inverter a mezzo ponte monofase con tensione e corrente di uscita sinusoidali che alimenta un carico di caratteristiche induttive è mostrato in figura 8.2.

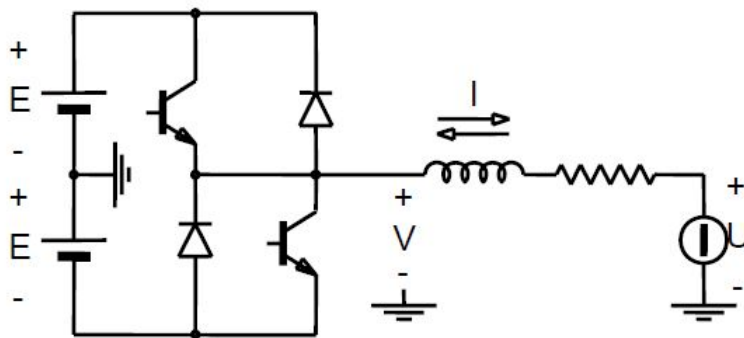


Figura 8.2: Inverter a mezzo ponte [11]

La struttura presenta due interruttori connessi in serie con il relativo diodo in antiparallelo e due generatori ideali uguali di tensione continua. Tale sistema permette due stati:

- se l'interruttore alto è acceso e quello in basso è spento l'uscita è positiva e pari ad E ;

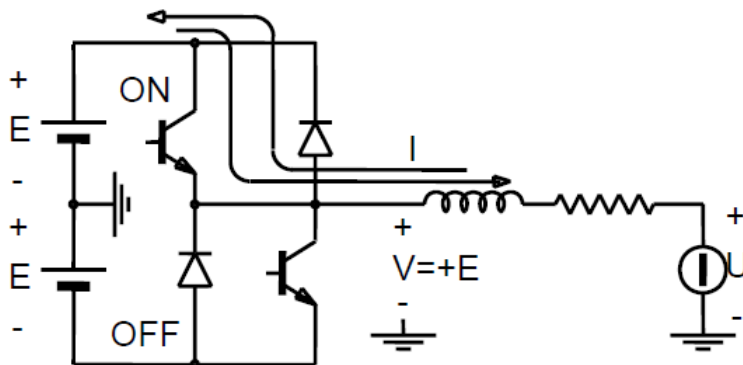


Figura 8.3: Inverter a mezzo ponte [11]

- se l'interruttore basso è acceso e quello in alto è spento l'uscita è negativa e pari ad $-E$;

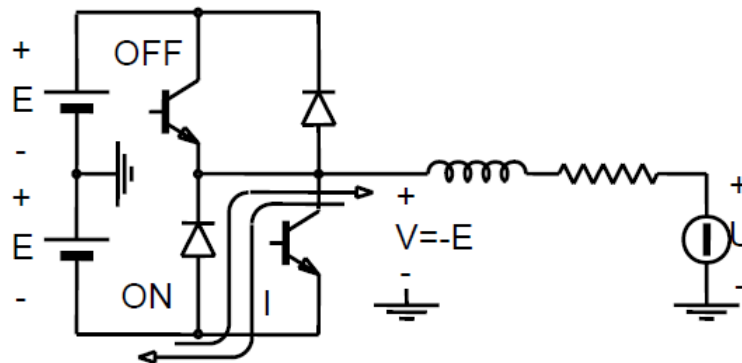


Figura 8.4: Inverter a mezzo ponte [11]

I due interruttori non sono mai entrambi accesi, in quanto in questo caso si avrebbe la presenza di un corto circuito tra le alimentazioni.

La presenza dei diodi in antiparallelo garantisce una via di richiusura alle correnti di carico, nei momenti in cui si vuole che la potenza istantanea fluisca dal carico verso l'alimentazione.

La presenza di una induttanza in serie all'uscita garantisce l'assenza di impulsi di corrente di valore elevato durante le commutazioni, dovute all'accumulo di tipo capacitivo in continua.

L'inverter a ponte intero consiste nella connessione in serie di due inverter a mezzo ponte. La tensione in uscita a tale inverter è data dalla differenza delle tensioni in uscita a ciascuno dei due inverter a mezzo ponte.

$$V_{full-bridge}(t) = V_{half-bridge1}(t) - V_{half-bridge2}(t) \quad (8.1)$$

La modulazione scelta per comandare l'inverter full-bridge è stata quella a due livelli. Tale tecnica consiste nel comandare le due gambe dell'inverter in modo tale che il segnale che comanda la gamba negativa sia il negato del segnale che comanda la gamba positiva, come mostrato in figure 8.5 e 8.6.

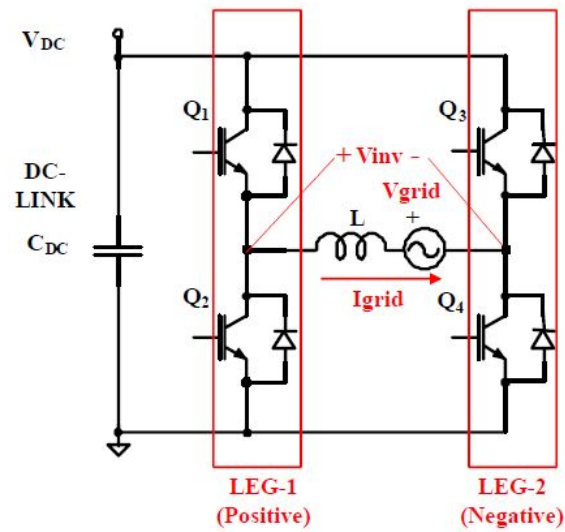


Figura 8.5: Inverter a ponte intero da emulare [39]

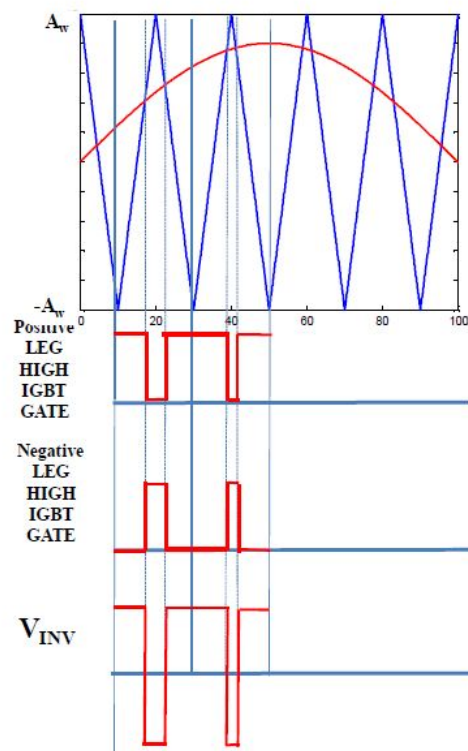


Figura 8.6: Modulazione a due livelli dell'inverter [39]

In figura 8.5 si vede che ogni gamba presenta due interruttori chiamati Q_1 e Q_2 per la gamba positiva e Q_3 e Q_4 per la gamba negativa.

Gli interruttori di una stessa gamba lavorano in modo complementare. Dunque si avrà che saranno contemporaneamente accesi o spenti gli interruttori Q_1 e Q_4 e Q_2 e Q_3 come in figura 8.7. Per fornire in uscita una tensione di $+V_{DC}$ è necessario che siano attivi gli interruttori Q_1 e Q_4 , viceversa per una tensione di $-V_{DC}$ si attiva Q_2 e Q_3 .

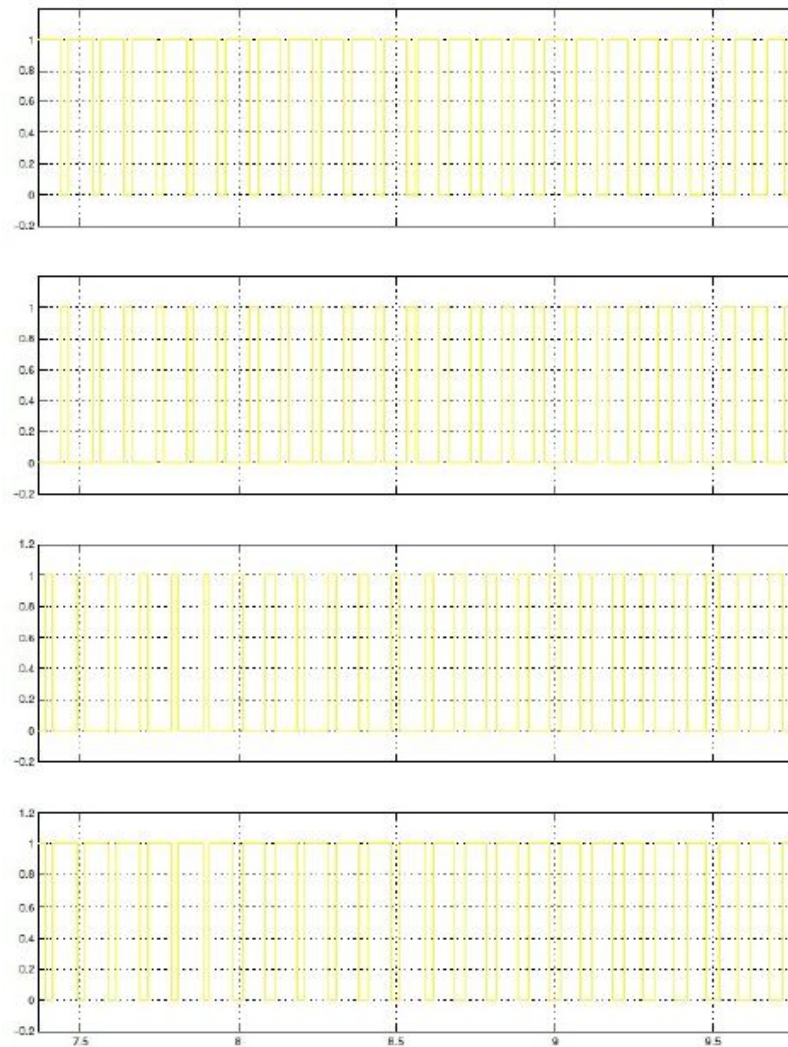


Figura 8.7: Comandi di Q_1 Q_2 Q_3 Q_4

8.2 Emulatore

Il programma descritto nei precedenti capitoli ha il compito di controllare la corrente in uscita di un emulatore di inverter, il quale simula il funzionamento di un inverter a ponte intero monofase collegato alla rete, come quello di figura 8.8.

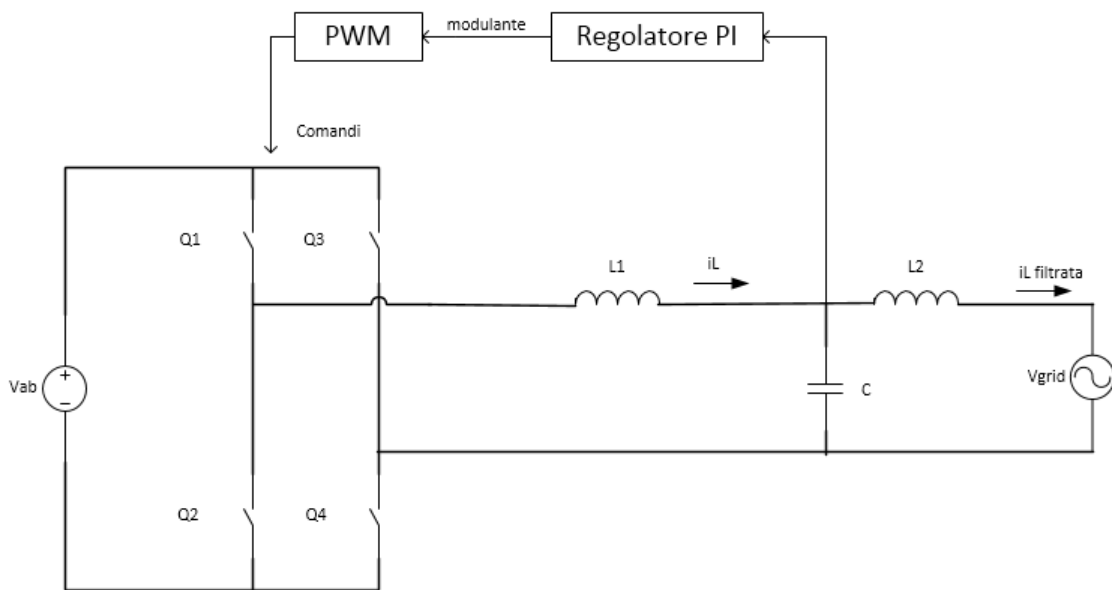


Figura 8.8: Modello da emulare

Il circuito che simula il funzionamento del modello di inverter presenta due stadi come si può vedere in figura 8.9.

Il primo stadio è un integratore non invertente, il cui compito è quello di emulare la dinamica della induttanza di fase che avviene sfruttando la capacità C_1 come integratore.

L'equazione che esprime l'uscita del primo stadio è espressa da:

$$V_{phase-current}(s) = \frac{V_{inv}(s) - V_{grid}(s)}{sR_1C_1} \quad (8.2)$$

L'uscita di questo primo stadio coincide con la corrente in uscita all'inverter emulato, cioè la I_L di figura 8.10 dello schema a blocchi del controllo di corrente.

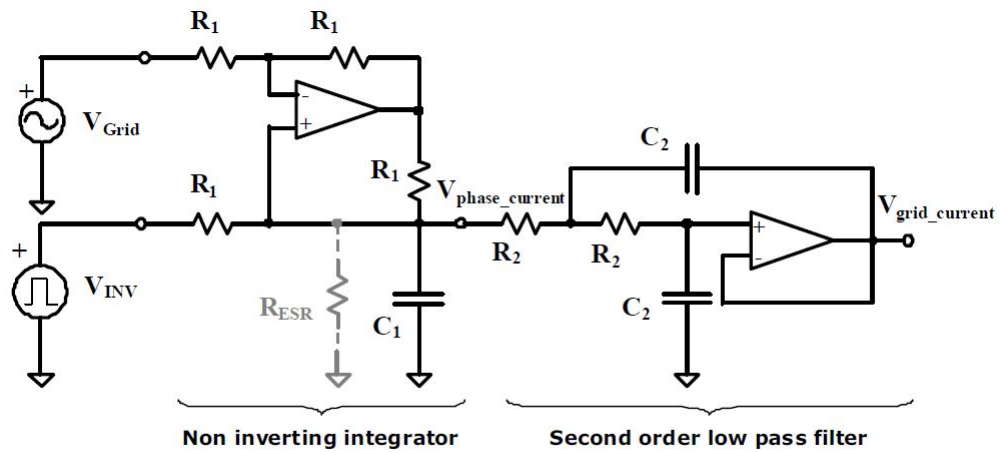


Figura 8.9: Circuito emulatore [40]

Il secondo stadio rappresenta invece un filtro del secondo ordine di tipo Sallen Key la cui funzione di trasferimento è data da:

$$\frac{V_{grid-current}(s)}{V_{phase-current}(s)} = \frac{1}{(1 + sR_2C_2)^2} \quad (8.3)$$

L'uscita del filtro Sallen Key è la corrente di rete, cioè la I_L priva di ripple.

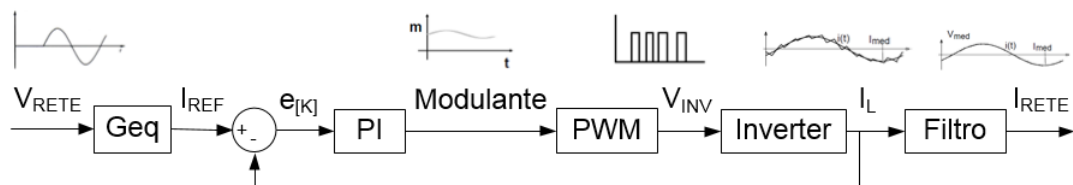


Figura 8.10: Controllo

Nel dettaglio lo schema elettrico del primo stadio dell'emulatore è presentato in figura 8.11.

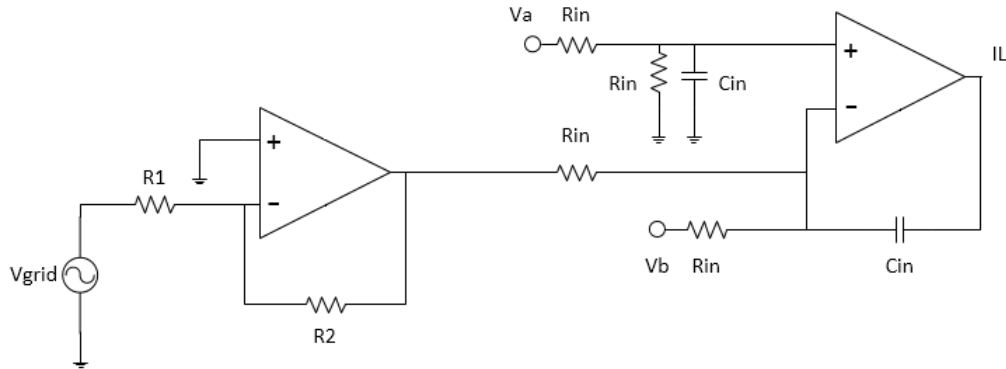


Figura 8.11: Schematico emulatore

L'uscita del primo operazionale, in configurazione invertente è data dall'equazione:

$$V_{invertente}(s) = -\frac{R_2}{R_1} \cdot V_{grid}(s) \quad (8.4)$$

L'uscita del secondo operazionale, dunque I_L è data da:

$$V_L(s) = \frac{1}{sR_{in}C_{in}}(V_a - V_b - V_{invertente}(s)) \quad (8.5)$$

Di seguito sono elencate le caratteristiche che si vuole abbia l'inverter da emulare e di conseguenza le caratteristiche dell'emulatore.

Caratteristiche dell'inverter da emulare:

- Potenza nominale $P_0 = 3kW$;
- Frequenza di commutazione $f_s = 40kHz$;
- Tensione di ingresso $V_{dc} = 400V$;
- Massimo indice di modulazione $m_{max} = \frac{V_r \cdot 1,2 \cdot \sqrt{2}}{V_{dc}} = 0,976$;
- Indice di modulazione nominale $m_{nom} = \frac{m_{max}}{1,2} = 0,813$;
- Ripple massimo di corrente picco-picco in rete $\Delta I_{rel} = 0,02$;
- Corrente di picco $I_{rpk} = \frac{P_0 \cdot \sqrt{2}}{V_r} = 18,446$.

Inoltre la schedina in cui è presente l'emulatore dell'inverter è provvista di un oscillatore che simula la tensione di rete:

- Tensione di rete $V_r = 230V$;
- Frequenza di rete $f_r = 50Hz$.

Il valore dell'induttanza di fase è ricavato da:

$$L_f = \frac{V_{dc}}{2f_s \Delta I_{fpp}} \quad (8.6)$$

dove:

$$\Delta I_{fpp} = \frac{P_0}{V_r} \cdot \sqrt{2} \cdot \Delta I_{frel} = 0,369A \quad (8.7)$$

Il valore dell'induttanza di fase con ipotesi di singolo induttore risulta pari a $L_f = 0,014H$. Tale valore è piuttosto elevato e ciò può comportare una riduzione della velocità di risposta del sistema di controllo durante i transitori. Per questo si preferisce utilizzare un filtro L-C-L.

Fattori di scala e parametri dell'emulatore:

- Tensione dell'inverter $V_{AB} = \pm 12V$;
- Fattori di scala per le tensioni $K_V = \frac{V_{AB}}{V_{dc}} = 0,03$;
- Fattori di scala per le correnti $K_A = 0,18 \frac{V}{A}$;
- Ripple emulato $\Delta I_{emul} = \Delta I_{fpp} \cdot K_A = 0,664V$;
- Corrente di picco emulata $I_{emup} = \frac{P_0}{V_r} \cdot \sqrt{2} \cdot K_A = 3,32V$;
- Resistenza di ingresso $R_{in} = 10k\Omega$;
- Capacità integratore $C_{emul} = 22nH$;
- Costante di tempo integratore $\tau_{Lemul} = 2,2 \cdot 10^{-4}s$;
- Resistenza $R_1 = 16,8k\Omega$;
- Resistenza $R_2 = 300k\Omega$.

L'emulatore simula il comportamento dell'induttanza di fase grazie all'utilizzo della resistenza R_{in} e del condensatore C_{in} precedentemente definiti.

Il rapporto tra l'induttanza di fase e l'induttanza di fase emulata è data da:

$$L_{emul} = \frac{V_{AB}}{2f_s \Delta I_{emul}} = \frac{K_V \cdot V_{dc}}{2f_s \Delta I_{emul}} = \frac{K_V}{K_A} \cdot \frac{V_{dc}}{2f_s \Delta I_{fpp}} = \frac{K_V}{K_A} \cdot L_f \quad (8.8)$$

Affinché sia garantita la corretta corrispondenza tra l'inverter e l'emulatore deve esser verificata l'equazione 8.8.

Capitolo 9

Risultati sperimentali

Per effettuare i test sul programma di controllo di corrente si è utilizzata la "GPIC" a cui è stato collegato l'emulatore. La scelta della "GPIC" e non del "CRIO" è legata al fatto che il "CRIO" possiede dei limiti troppo restrittivi in riferimento al massimo valore di tensione che può erogare.

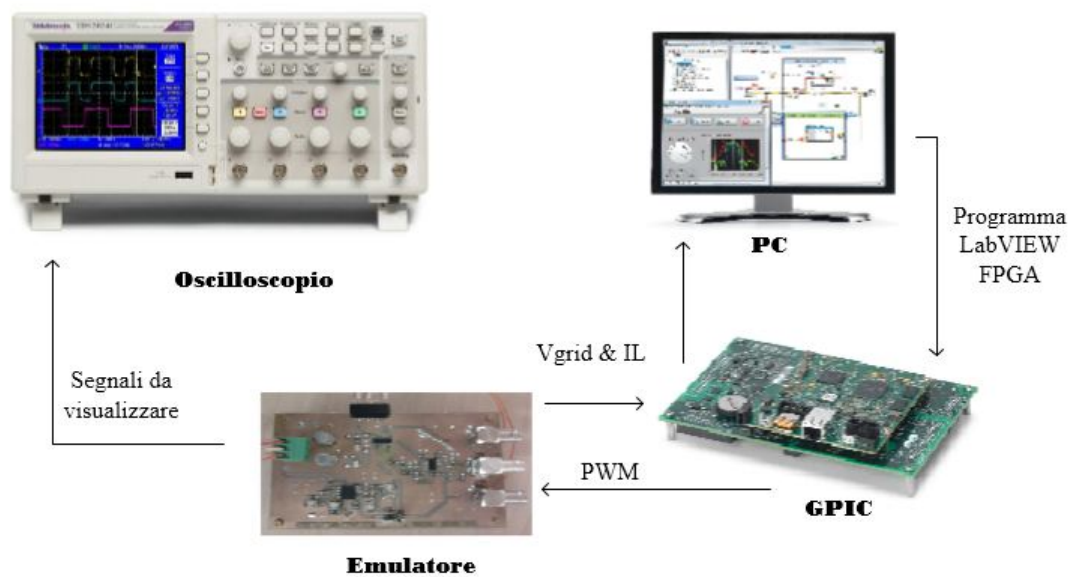


Figura 9.1: Banco di lavoro

I test effettuati per verificare il corretto funzionamento del programma di controllo sono stati fatti per gradi:

- impostando la corrente di riferimento ad un valore costante e verificando che in tal modo la corrente in uscita all'emulatore fosse dello stesso valore

di quella di riferimento. Tale verifica si è compiuta per entrambi i valori di banda passante desiderati pari a $\frac{f_{sw}}{10}$ ed a $\frac{f_{sw}}{8}$.

- Verificato il corretto funzionamento nel caso di riferimento costante si sceglie ora di utilizzare come riferimento la tensione di rete, opportunamente scalata, emulata dall'oscillatore presente nella scheda.

Il programma realizzato in LabVIEW effettua il controllo di corrente acquisendo tramite la "GPIC" la tensione emulata di rete e la corrente in uscita dall'emulatore. La "GPIC" a sua volta emette i segnali PWM di comando di gate garantendo così la tensione dell'inverter V_{inv} desiderata. La corrente in uscita all'emulatore di inverter è controllata dai segnali di gate provenienti dalla "GPIC" e dunque dai segnali PWM.

9.1 Risultati con riferimento fisso

Impostando un riferimento costante si nota come la corrente in uscita all'emulatore segua tale valore. In figura 9.2 viene mostrato in blu l'andamento della corrente ed in giallo e fucsia i due segnali PWM.

Con entrambi i valori di K_I e K_P ottenuti con le bande rispettivamente di $\frac{f_{sw}}{10}$ e di $\frac{f_{sw}}{8}$ si ottiene lo stesso risultato di figura 9.2.

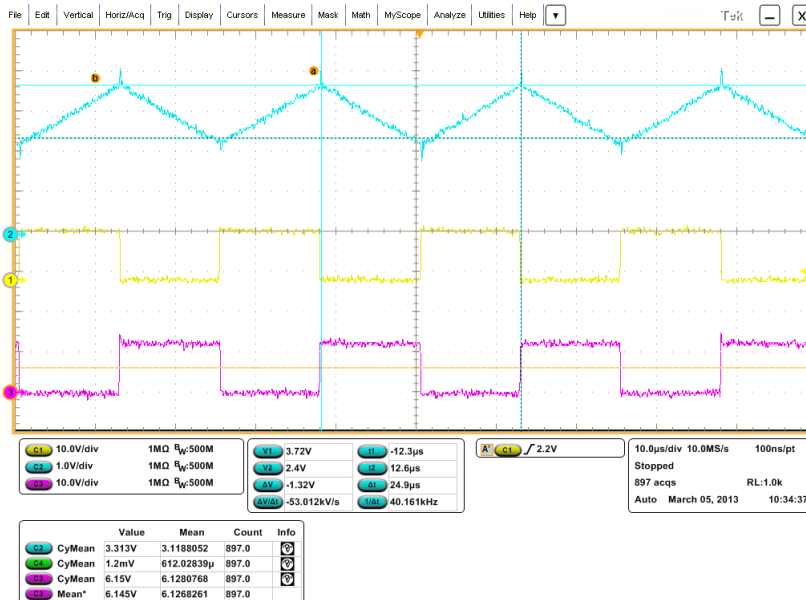


Figura 9.2: TEST con riferimento $I^* = 3$

Si è deciso di esaminare anche il transitorio al fine di rilevare il tempo di salita ed il tempo di assestamento. Nella figure 9.3, 9.4 e 9.5 si effettua rispettivamente una variazione tra i riferimenti -7 e 7, -4 e 4 e -2 e 2. Il tempo di salita rilevato per tali transistori è rispettivamente pari a $185\mu s$, $110\mu s$ e $60\mu s$. Il tempo di assestamento invece risulta rispettivamente pari a $680\mu s$, $635\mu s$ e $175\mu s$.

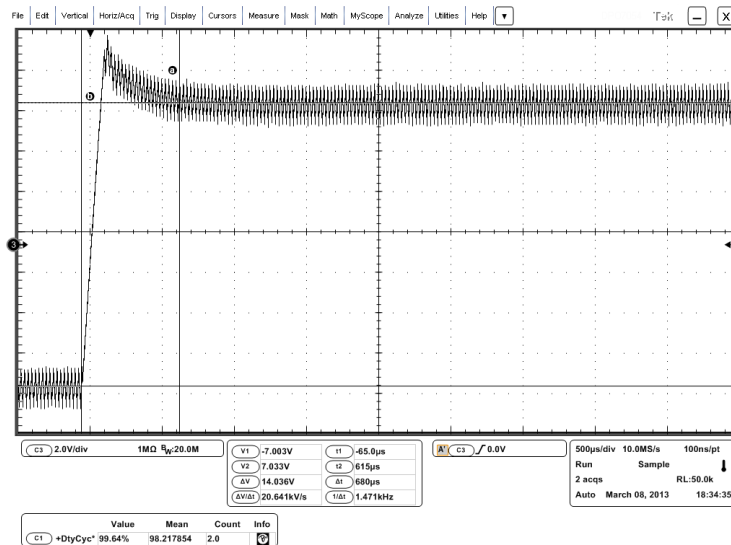


Figura 9.3: Transitorio della variazione della I^* tra -7 e 7

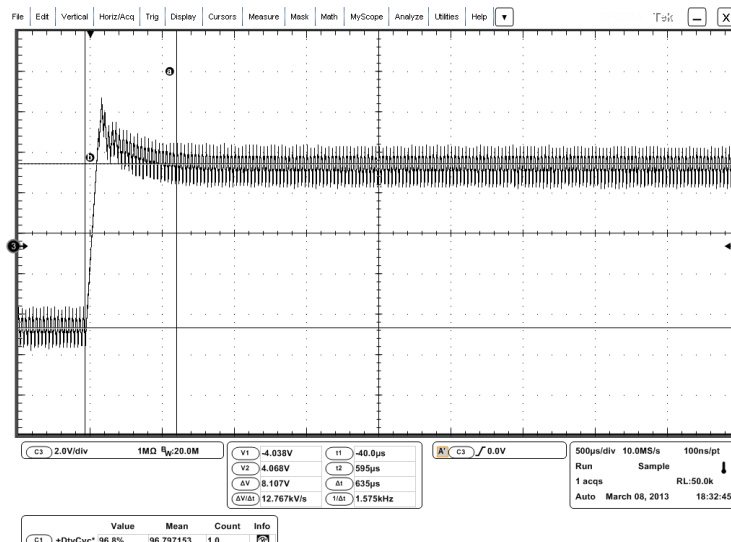


Figura 9.4: Transitorio della variazione della I^* tra -4 e 4

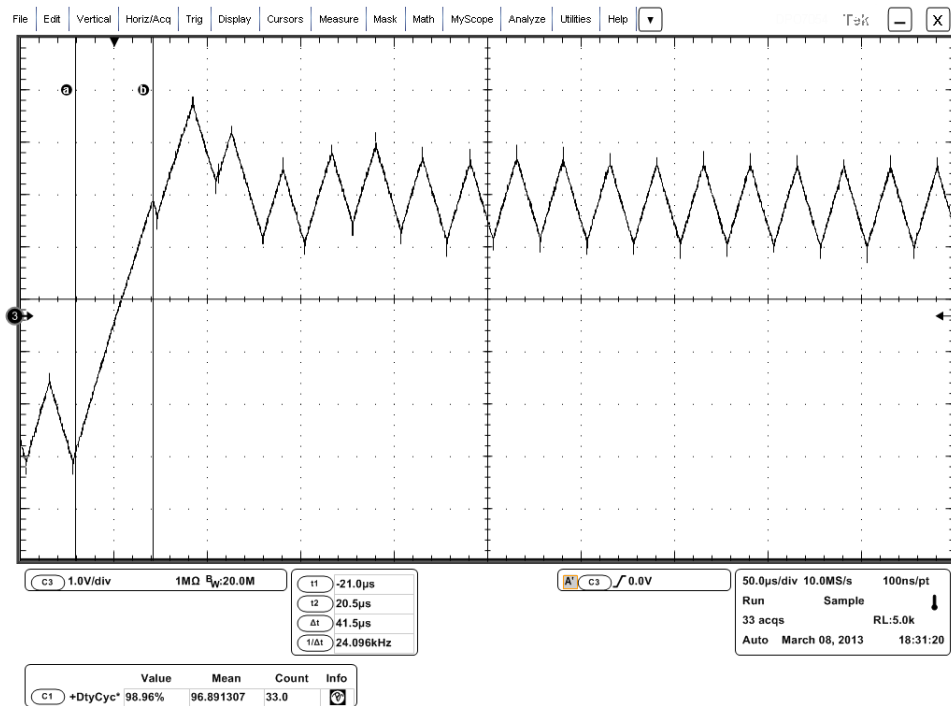


Figura 9.5: Transitorio della variazione della I^* tra -2 e 2

Delle tre immagini si nota che solo durante il transitorio di figura 9.5 avviene un tentativo di commutazione durante il tempo di salita. Mentre le altre immagini mostrano l'assenza di tale fenomeno.

Questo avviene per effetto della saturazione del regolatore. Infatti variando i valori di K_I e K_P riducendoli entrambi di un fattore 10, diminuendo la banda di controllo, si nota la presenza della commutazione durante il tempo di salita.

In figura 9.6 si vede il transitorio di corrente tra il riferimento -5 e 5, avendo utilizzato come valori di K_I e K_P 0,6 e 10.

Il tempo di salita risulta pari a $500\mu s$ mentre il periodo di assestamento è pari a $1,82ms$.

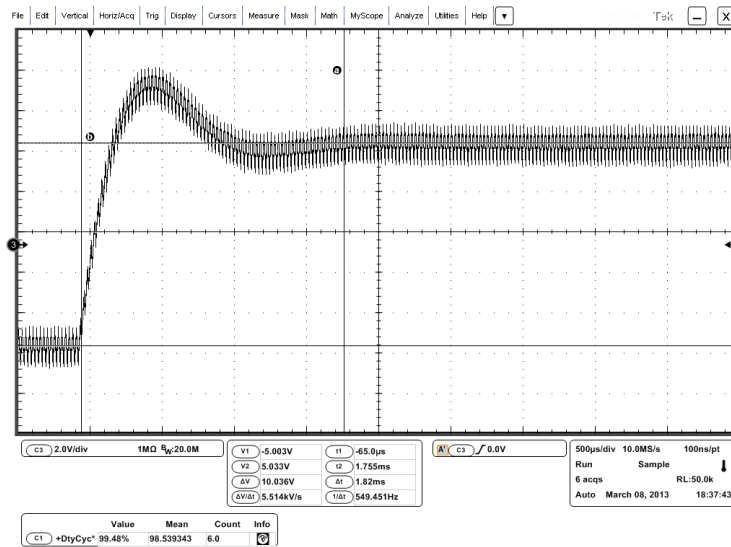


Figura 9.6: Transitorio della variazione della I^* tra -5 e 5 con k_i e k_p variati

L'overshoot in percentuale risulta dato da:

$$Overshoot = 100 \cdot \frac{(V_{max} - V)}{V} \quad (9.1)$$

Dove V_{max} è il massimo punto di sovralongazione e V il valore a regime.

L'overshoot di in figura 9.3 risulta circa del 36% mentre quello di figura 9.6 del 48%.

9.2 Risultati con riferimento la tensione di rete

Verificato il corretto funzionamento con l'utilizzo di un riferimento costante si è passati ad acquisire dall'emulatore e usare come riferimento la tensione di rete. Si è verificato il corretto funzionamento per entrambe le bande di $\frac{f_{sw}}{10}$ e di $\frac{f_{sw}}{8}$.

Nelle successive figure si vede in verde l'andamento della tensione di rete ed in fucsia l'andamento della corrente I_L .

In figura 9.7 sono visibili la V_{grid} e la I_L non filtrate ed in figura 9.8 si può vedere come esse siano perfettamente in fase.

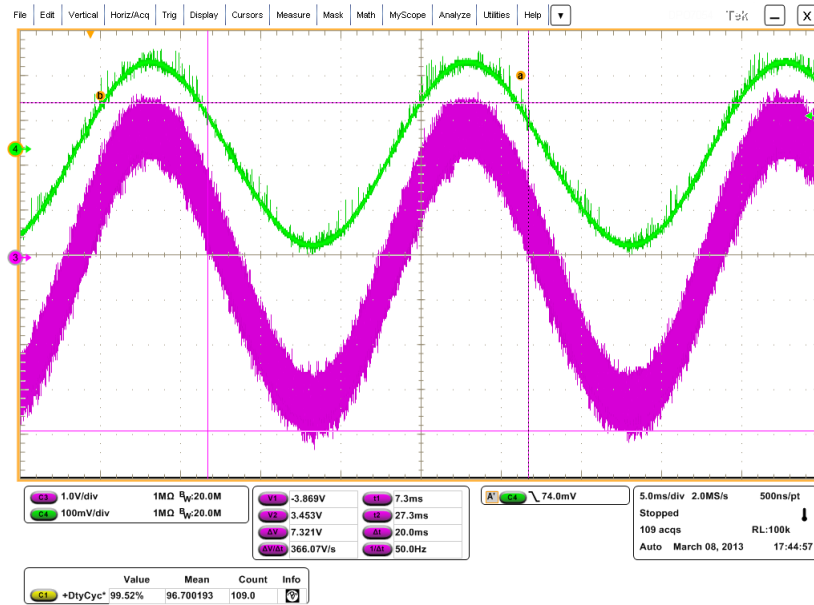


Figura 9.7: Corrente dell'emulatore e tensione di rete non filtrati

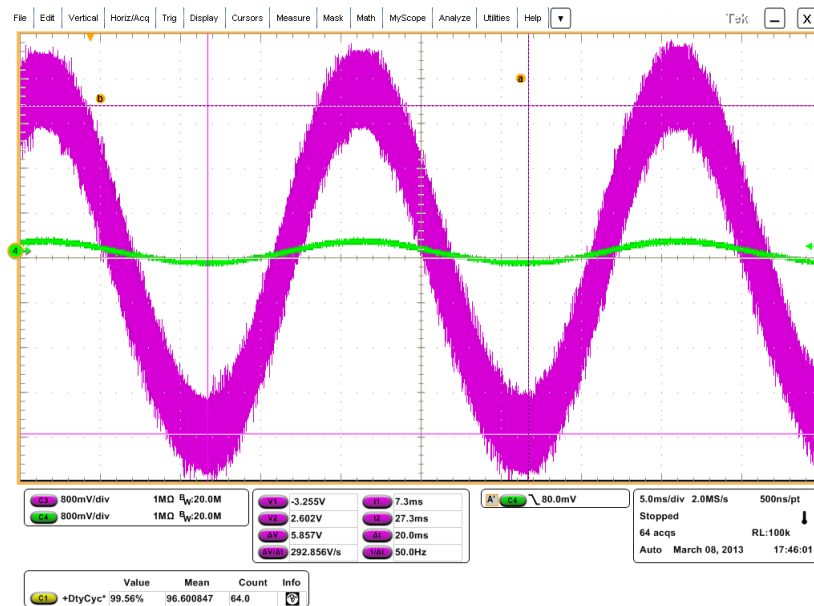


Figura 9.8: Corrente dell'emulatore e tensione di rete in fase

Nelle figure 9.9, 9.10 e 9.11 sono mostrate la tensione di rete e la corrente I_L filtrate dall'oscilloscopio. La tensione di rete è stata moltiplicata per valori di $G_{equivalente}$ pari rispettivamente a 3, 9 e 12.

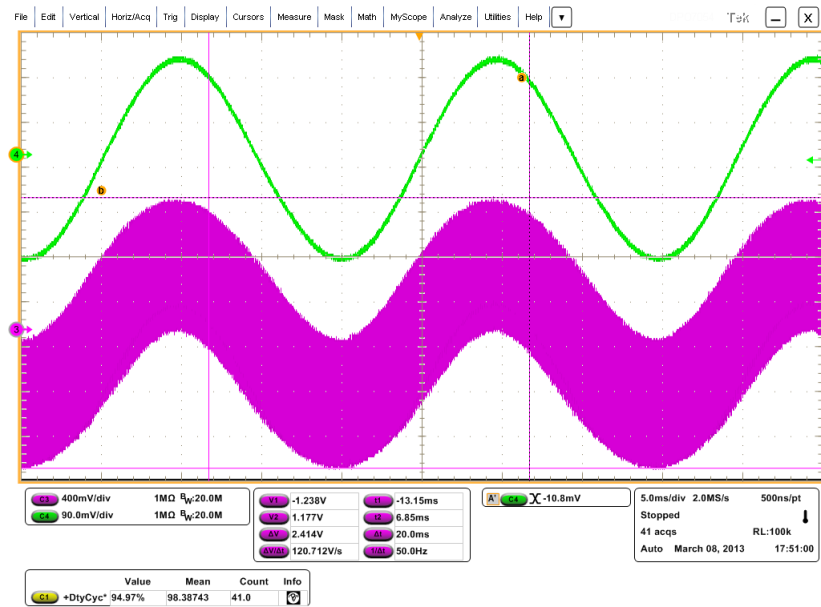


Figura 9.9: TEST con riferimento la tensione di rete filtrata e $G_{equivalente} = 3$

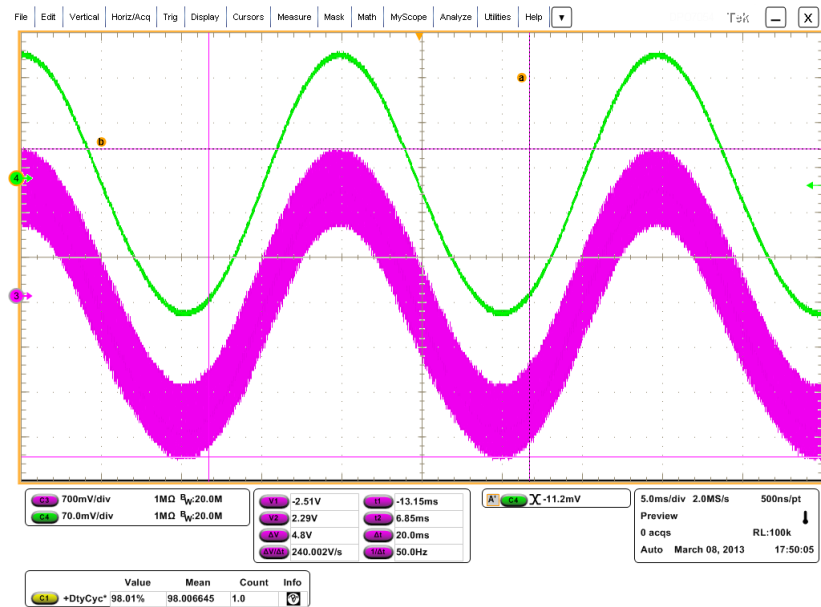


Figura 9.10: TEST con riferimento la tensione di rete filtrata e $G_{equivalente} = 9$

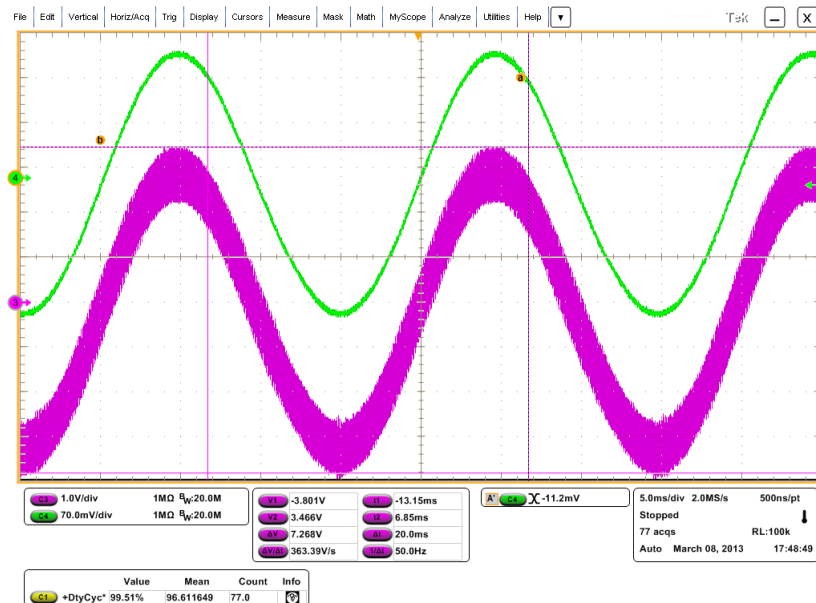


Figura 9.11: TEST con riferimento la tensione di rete filtrata e $G_{equivalente} = 15$

Per poter ottenere i valori di ripple emulato $\Delta I_{emul} = 0,662V$ e corrente di picco emulata $I_{emul-pp} = 3,32V$ è necessario porre $G_{equivalente} = 15$ come si vede in figura 9.11.

Le seguenti figure riportano l'impulso PWM in azzurro e l'andamento di I_L in fucsia.

In figura 9.12 si vede come al variare dell'impulso PWM ci sia la conseguente variazione dell'andamento della corrente.

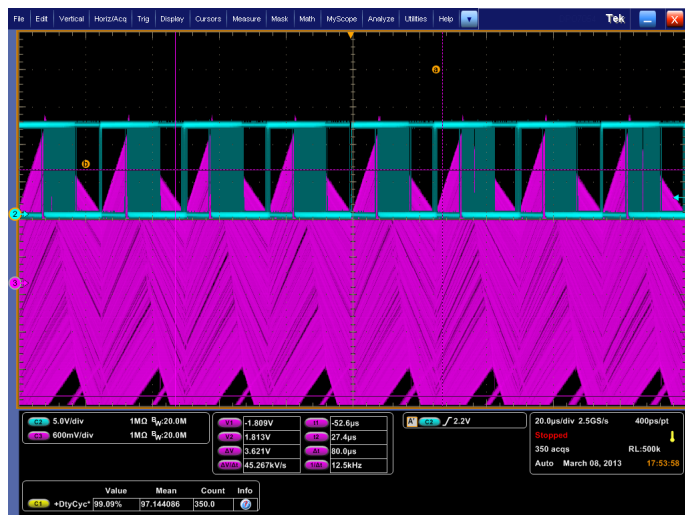


Figura 9.12: Come la corrente segue la variazione dell'impulso

In figura 9.13 si nota che il duty-cycle varia tra lo 32% e lo 69% e dunque si ha una $\Delta\delta = 37\%$.

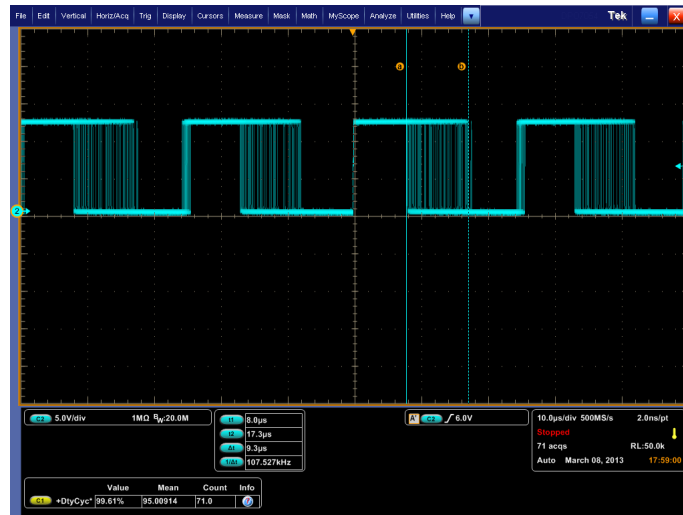


Figura 9.13: Range di variazione del duty-cycle

Capitolo 10

Conclusioni

Il lavoro svolto fa parte di un progetto universitario del settore di Elettronica di Potenza di ambito Smart Grids. Scopo di tale progetto è quello di poter simulare il funzionamento di un'intera rete e poterne conoscere e controllare la tensione ai nodi. Basandosi su tale informazione risulta infatti possibile decidere quanta potenza erogare, a quale nodo ed in quale momento.

Il controllo di corrente effettuato in ambiente di sviluppo LabVIEW FPGA può esser applicato non solo in ambito Smart Grids per inverter fotovoltaici, ma in tutti quegli ambienti in cui è presente un inverter connesso alla rete con controllore "National Instruments".

Il programma di controllo realizzato è stato sviluppato in modo tale da esser composto da un insieme di subVI singolarmente utilizzabili. Questo permette un riutilizzo di non solo l'intero programma, ma anche di singole parti di codice, quali quelle realizzanti il modulatore PWM, il regolatore PI e l'acquisitore. La versatilità del programma è garantita anche dallo stesso "Front Panel" che si è realizzato, che contempla la variazione di alcuni parametri quali il segnale di riferimento ed i valori di K_I e K_P . Tali parametri possiedono la caratteristica di poter esser variati anche durante l'esecuzione del programma, evitando ulteriori compilazioni.

La funzionalità e la correttezza del codice è stata verificata mediante test, i cui esiti sono stati presentati nel capitolo 9. Il programma possiede una buona velocità di risposta alle variazioni di riferimento ed utilizza una frequenza di modulazione di $40kHz$. Allo stesso modo dei parametri precedentemente citati è possibile variare anche tale frequenza andando ad intervenire sulla scelta del ciclo temporizzato contenente il subVI del modulatore PWM.

Ringraziamenti

Il primo ringraziamento va ai miei genitori Graziella e Franco per il sostegno e l'aiuto che mi hanno sempre dato.

Grazie a Renzo per essermi stato accanto in ogni momento, per l'avermi spronata a dare il meglio e per aver reso le mie giornate più belle e complete.

Grazie ai miei compagni/colleghi di università, in particolare grazie a Luca, Tommaso (il quale può esser definito il mio correlatore non ufficiale), Fabio e Marco. Il vostro aiuto è stato indispensabile per raggiungere questo traguardo.

Ringrazio tutti i miei amici, i quali hanno reso questi anni speciali. Grazie in particolare a Elisa, Federica e Nicoletta.

Grazie al mio fratellone Paolo per esserci sempre stato.

Un ringraziamento particolare è dovuto ai professori Paolo Tenti ed Elena Gaio per la passione che mi hanno trasmesso.

Ringrazio infine il mio relatore Simone Buso per i consigli ricevuti per realizzare questa tesi e per la grande professionalità che ha dimostrato.

Bibliografia

- [1] Hsin-Chuan Chen, "A High-Resolution Digital PWM Controller Using Capacitor Integration", IEEE, pp. 166-169, 2010.
- [2] S. Saggini, P. Mattavelli, M. Zigliotto, D. Trevisan, "Limited-Pool Random Carrier-Frequency PWM for Digitally Controlled dc-dc Converters", IEEE, pp. 4929-4934, 2006.
- [3] Asif Syed, Ershad Ahmed and Dragan Maksimovic', Eduard Alarcon, "Digital Pulse Width Modulator Architectures", IEEE, Conf., pp. 4689-4695, 2001.
- [4] Yu-Cherng Hung, "Delay-Line Sharing Based: A New CMOS Digital PWM Circuit", IEEE, International Symposium on Computer, Consumer and Control, pp 305-308, 2012.
- [5] Abram P., Dancy and Anantha P. Chandrakasan, "Ultra low power control circuits for PWM converters", IEEE, pp. 21-27, 1997.
- [6] Benjamin J. Patella, Aleksandar Prodic, Art Zirger, Dragan Maksimovic, "High-Frequency Digital PWM Controller IC for DC-DC Converters", IEEE TRANSACTIONS ON POWER ELECTRONICS, VOL. 18, NO. 1, JANUARY 2003.
- [7] Vahid Yousefzadeh, Toru Takayama, Dragan Maksimovic, "Hybrid DPWM with Digital Delay-Locked Loop", IEEE COMPEL Workshop, Rensselaer Polytechnic Institute, Troy, NY, USA, July 16-19, pp. 142-148, 2006.
- [8] Xuzhen Shen, Xiaobo Wu, and Jing Lu, Lin Qin, "Hybrid DPWM with Analog Delay Locked Loop", in Proc. IEEE. IMECS, 2010.
- [9] Simone Buso, "Digital Control of switching mode power supplies" UNICAMP, Agosto 2011.
- [10] Simone Buso, Slide del corso "Microcontrollori e DSP".

- [11] Paolo Tenti, slide del corso di "elettronica industriale"
- [12] Jian Li, "Current-Mode Control: Modeling and its Digital Application", Blacksburg, Virginia, April 14th, 2009.
- [13] Simone Buso, Paolo Mattavelli, Digital Control in Power Electronics, Morgan & Claypool, 2006.
- [14] R8C/3x Timers for Power Conversion February 2010.
- [15] Course Manual, "CompactRIO and LabVIEW Development Fundamentals" NATIONAL INSTRUMENTS.
- [16] <http://www.ni.com/compactrio>.
- [17] Tutorial, National instruments, pubblicazione dic 19, 2011.
- [18] Vogrig Daniele, slide del corso di "L.E.D".
- [19] CompactRIO Developers Guide, "Recommended LabVIEW Architectures and Development Practices for Machine Control Applications", National Instruments, Maggio 2009.
- [20] <http://calogerolombardo.altervista.org/index.php/tutorial/36-tutorial/49-microcontrollori-pic>.
- [21] Angelo Monfroglio "I MICROCONTROLLORI PIC (MICROCHIP)" PARTE I.
- [22] MICROCHIP, Section 15. Motor Control PWM, 2007
- [23] Renesas, APPLICATION NOTE R01AN1050EJ0100 Rev.RX630 Group, R01AN1050EJ0100 Rev. 1.00 Page 1 of 8 Oct. 5, 2012.
- [24] Renesas, R8C/35A Group, April 1st, 2010 Renesas Electronics Corporation.
- [25] Renesas, RX630 Group, DATASHEET.
- [26] LABVIEW FPGA how to program FPGAs without any VHDL knowledge, NI.
- [27] LabVIEW User Group Meeting, Long Island Chapter September 11 2008.
- [28] DATASHEET NI, 2012, NI 9263.

- [29] DATASHEET NI, 2012, NI 9264.
- [30] DATASHEET NI, 2012, NI 9401.
- [31] DATASHEET NI, 2012, NI 9215.
- [32] Introduzione ai microcontrollori PIC.
- [33] A.P. Dancy, R. Amirtharajah, A.P. Chandrakasan, "High-Efficiency Multiple-Output DC-DC Conversion," IEEE Trans. VLSI Syst., vol. 8, no. 3, pp. 252-263, June 2000.
- [34] A. Dancy, A. Chandrakasan, "Ultra Low Power Control Circuits for PWM Converters," IEEE PESC 1997, vol. 1, 1997, pp. 21-27.
- [35] A. Dancy, A. Chandrakasan, "A Reconfigurable Dual Output Low Power Digital PWM Power Converter," IEEE International Symposium on Low Power Electronics and Design, pp. 191-196, 1998.
- [36] DIGITAL PULSE WIDTH MODULATION TECHNIQUES FOR POWER CONVERTERS by THANUKAMALAM ARUNACHALAM.
- [37] USER GUIDE AND SPECIFICATIONS NI 9683 General Purpose Inverter Controller RIO Mezzanine Card.
- [38] Luca Corradini, slide del corso di "energia 2".
- [39] Alessandro Costabeber, report.
- [40] Simone Buso, report emulatore.

Elenco delle figure

1	LabVIEW & LabVIEW FPGA	iii
2	CompactRIO & GPIC	1
1.1	Controllo di un convertitore buck con modulazione PWM analogica [11]	3
1.2	Principio di funzionamento della modulazione analogica [11]	4
1.3	Modulazione analogica con modulante costante [10]	5
1.4	Andamento della modulante e della tensione media [11]	5
1.5	Andamento della modulante e della tensione media [10]	6
1.6	Segnale demodulato [10]	7
1.7	Leading-edge [36]	8
1.8	Tipologie di DPWM counter-based [12]	8
1.9	Leading-edge [10]	9
1.10	Dual-edge [10]	10
1.11	Campionamento con e senza sincronizzazione [9]	11
1.12	Tipologie di DPWM counter-based	12
1.13	Double-edge [12]	12
1.14	Delay-line [3]	13
1.15	Cella di ritardo [3]	13
1.16	a) delay-line open-loop b)delay-line ring-oscillator [3]	14
1.17	delay-line test [6]	15
1.18	Hybrid [36]	15
1.19	Hybrid test [6]	16
1.20	Hybrid test con assegnazione esterna del segnale di clock [7]	17
1.21	Potenza dissipata con il valore di conteggio pari a 1024 e 256 con la tipologia hybrid [35]	17
1.22	SCI DPWM [1]	19
1.23	Tabella di confronto [1]	19
1.24	Delay-line ed hybrid random pulse width modulator [2]	20
1.25	Analisi spettrale priva di modulazione random ad $f_{sw} = 833kHz$ [2]	21
1.26	Analisi spettrale con modulazione random a $f_{sw1} = 833kHz$ e $f_{sw2} = 1.041MHz$ [2]	21

1.27	Analisi spettrale con modulazione random a $f_{sw1} = 500kHz$, $f_{sw2} = 890kHz, f_{sw3} = 1.200MHz$ [2]	22
1.28	Metodo che garantisce gli impulsi centrati [2]	22
1.29	Modulazione analogica [12]	23
1.30	Modulazione digitale [12]	24
2.1	Modulazione digitale a due livelli con microcontrollore [11]	27
2.2	Pic 32[32]	29
2.3	Caratteristiche dsPIC30F[22]	29
2.4	PWM Leading-Edge[22]	30
2.5	PWM Dual-Edge[22]	30
2.6	Modalità free running[22]	31
2.7	Up/down counting mode [22]	31
2.8	Realizzazione della modulazione PWM[23]	32
2.9	Funzionamento di un motore [14]	32
2.10	Controllo di un motore DC [14]	33
2.11	Controllo di un motore brushless DC [14]	33
3.1	CompactRIO [16]	35
3.2	Struttura del compactRIO [15]	36
3.3	CHASSIS da 8 slot[16]	36
3.4	NI cRIO-9014 Real-Time Controller [19]	37
3.5	Descrizione del controllore [15]	37
3.6	Modulo I/O [19]	38
3.7	Architettura completa del CompactRIO[19]	38
3.8	CompactRIO 9082 [16]	38
3.9	NI 9215 [31]	39
3.10	NI 9263 [28]	40
3.11	NI 9264 [29]	41
3.12	NI 9401 [30]	42
3.13	Struttura di un chip FPGA [19]	42
3.14	Fasi di progetto[18]	43
3.15	Relazione tra architettura e VIs [15]	44
3.16	Accesso diretto agli I/O [17]	44
3.17	Funzionamento dello Scan Mode [17]	46
3.18	Architettura [15]	47
3.19	Definizione della logica combinatoria FPGA grazie a LabVIEW FPGA Module [16]	47
3.20	Esempio di utilizzo di LabVIEW FPGA Module [16]	48
3.21	LabVIEW FPGA [19]	48
3.22	LabVIEW FPGA [26]	49
3.23	Interazione con FPGA tramite host PC based Front Panel [26]	50

3.24	Interactive Front Panel Communication [15]	50
3.25	LabVIEW Real-Time [16]	51
3.26	Controller Real-Time [16]	51
3.27	Architettura [27]	52
4.1	GPIC [37]	55
4.2	Simultaneous [37]	56
4.3	Half-Bridge [37]	56
5.1	SubVI contatore singolo	60
5.2	VI contatore singolo	61
5.3	Generatore di onde quadre	62
5.4	Simulazione con duty pari al 30%	62
5.5	Simulazione con duty pari al 50%	63
5.6	Contatore doppio	64
5.7	SubVI del contatore doppio	65
5.8	VI del contatore doppio	66
5.9	VI del contatore doppio	67
5.10	VI del contatore doppio	68
5.11	VI del contatore doppio	68
5.12	VI del contatore doppio	69
5.13	VI del contatore doppio	69
5.14	Periodo di 100 cicli	70
5.15	Periodo di 1 ciclo	71
5.16	Divisore 1	71
5.17	Divisore 2	72
5.18	Divisore 3	72
5.19	VI del contatore doppio	72
6.1	Regolatore [38]	73
6.2	Tabella [38]	74
6.3	Diagramma di bode con regolatore e con la banda pari a $\frac{f_{sw}}{8}$	76
6.4	Diagramma di bode con regolatore e con la banda pari a $\frac{f_{sw}}{10}$	76
6.5	Diagramma di bode senza regolatore	77
6.6	Anti-wind-up [10]	78
6.7	Anti-wind-up [10]	78
6.8	Regolatore PI con anti-wind-up	80
6.9	Regolatore PI utilizzato	81
7.1	Controllo [38]	83
7.2	Acquisizione	85
7.3	Front Panel	86

7.4	Controllo	86
8.1	Schema a blocchi di un convertitore connesso alla rete [39]	87
8.2	Inverter a mezzo ponte [11]	88
8.3	Inverter a mezzo ponte [11]	88
8.4	Inverter a mezzo ponte [11]	89
8.5	Inverter a ponte intero da emulare [39]	90
8.6	Modulazione a due livelli dell'inverter [39]	90
8.7	Comandi di Q_1 Q_2 Q_3 Q_4	91
8.8	Modello da emulare	92
8.9	Circuito emulatore [40]	93
8.10	Controllo	93
8.11	Schematico emulatore	94
9.1	Banco di lavoro	97
9.2	TEST con riferimento $I^* = 3$	98
9.3	Transitorio della variazione della I^* tra -7 e 7	99
9.4	Transitorio della variazione della I^* tra -4 e 4	99
9.5	Transitorio della variazione della I^* tra -2 e 2	100
9.6	Transitorio della variazione della I^* tra -5 e 5 con k_i e k_p variati .	101
9.7	Corrente dell'emulatore e tensione di rete non filtrati	102
9.8	Corrente dell'emulatore e tensione di rete in fase	102
9.9	TEST con riferimento la tensione di rete filtrata e $G_{equivalente} = 3$	103
9.10	TEST con riferimento la tensione di rete filtrata e $G_{equivalente} = 9$	103
9.11	TEST con riferimento la tensione di rete filtrata e $G_{equivalente} = 15$	104
9.12	Come la corrente segue la variazione dell'impulso	104
9.13	Range di variazione del duty-cycle	105