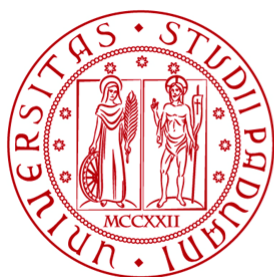


UNIVERSITÀ DEGLI STUDI DI PADOVA

FACOLTÀ DI INGEGNERIA



DEPARTMENT OF  
INFORMATION  
ENGINEERING

UNIVERSITY OF PADOVA



---

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE  
Corso di Laurea Triennale in Ingegneria Elettronica

**Sviluppo di un sistema di misura  
basato su LabVIEW  
per l'analisi di diagrammi di radiazione**

**Relatore:**

*Prof.ssa Giada Giorgi*

**Laureando:**

*Michael Corazza*

Anno Accademico 2009-2010



## Sommario

Il progetto ha lo scopo di realizzare un sistema di misura automatico per l'analisi delle radiazioni emesse da una sorgente.

Parte integrante del sistema di misura è una piattaforma realizzata in legno e plastica, entrambi materiali trasparenti alle radiazioni elettromagnetiche, sulla quale viene posizionato il circuito da analizzare. La piattaforma in legno inoltre viene pilotata mediante due motori di tipo passo-passo, gestiti entrambi dallo stesso sistema di misura automatizzato. In fig. 3.7 è riportata una foto della piattaforma in legno che, come si può osservare, è in grado di ruotare su sè stessa e lungo un asse trasversale, mediante l'azione dei due motori passo-passo.

Il progetto realizzato gestisce sia la rotazione della piattaforma, sia l'acquisizione delle radiazioni emesse dalla sorgente sotto esame.

È importante osservare che l'acquisizione delle forme d'onda è stata opportunamente sincronizzata con il movimento stesso della piattaforma.

Il programma di gestione dell'intero sistema di misura, di cui si fornirà una dettagliata descrizione in seguito, fornisce all'utente la possibilità di selezionare la risoluzione delle misure effettuate. In questo modo si potrà dare priorità ad aspetti relativi l'accuratezza delle misure di radiazione o, viceversa, al tempo impiegato per effettuarle.

Il programma realizzato inoltre fornisce all'utente la possibilità di interrompere l'esecuzione della misura, visualizzare il diagramma di radiazione per le diverse frequenze analizzate ed eventualmente salvare il risultato della misura in un report.





# Indice

<b>1</b>	<b>Introduzione</b>	<b>7</b>
1.1	Presentazione del sistema di misura . . . . .	7
1.2	Descrizione dei motori passo-passo . . . . .	9
1.2.1	Driver . . . . .	13
1.3	Descrizione scheda DAQ . . . . .	14
1.3.1	Gli ingressi analogici . . . . .	17
1.3.2	Uscite analogiche . . . . .	18
1.3.3	I/O digitali . . . . .	19
1.3.4	Collegamento tra DAQ e driver . . . . .	19
1.3.5	Convertitore ADC ad approssimazioni successive . . . . .	20
1.4	Analizzatore di spettro . . . . .	22
1.5	Funzionamento del sistema . . . . .	22
<b>2</b>	<b>Analizzatore di spettro</b>	<b>25</b>
2.1	Introduzione teorica sui segnali . . . . .	25
2.1.1	Trasformata di Fourier . . . . .	25
2.1.2	Basi teoriche sulle onde trasmesse . . . . .	26
2.2	Introduzione . . . . .	26
2.2.1	Tempo e frequenza a confronto . . . . .	26
2.2.2	Tipi di misurazioni . . . . .	27
2.2.3	Schermo . . . . .	27
2.3	Architettura e funzionamento base . . . . .	28
2.3.1	RF attenuator . . . . .	29
2.3.2	Low-pass filter or preselector . . . . .	29
2.3.3	Sintonizzare l'analizzatore . . . . .	29
2.3.4	IF gain . . . . .	29
2.4	Filtri analogici . . . . .	30
2.4.1	Risoluzione . . . . .	30
2.4.2	Sweep time . . . . .	30
<b>3</b>	<b>Programma LabVIEW</b>	<b>33</b>
3.1	Introduzione a LabVIEW . . . . .	33
3.2	Il programma di controllo del sistema . . . . .	34
3.2.1	Il front panel . . . . .	34
3.2.2	Schema a blocchi . . . . .	41
3.2.2.1	Controlli e inizializzazioni . . . . .	42
3.2.2.2	Il ciclo while principale . . . . .	43
3.2.2.3	Diagramma polare . . . . .	55
3.2.3	Il subVI del diagramma polare . . . . .	57

3.2.4	Generatore di report . . . . .	61
<b>4</b>	<b>Utilizzo dello strumento virtuale</b>	<b>65</b>
<b>5</b>	<b>Conclusioni</b>	<b>69</b>
5.1	Piattaforma mobile attraverso un sistema adatto . . . . .	69
5.2	Modalità simulazione avanzata . . . . .	70
5.3	Traccia spettrale selezionabile . . . . .	70

# Capitolo 1

## Introduzione

### 1.1 Presentazione del sistema di misura

Il sistema di misura realizzato in questo lavoro di tesi è stato schematizzato in fig. 1.1.

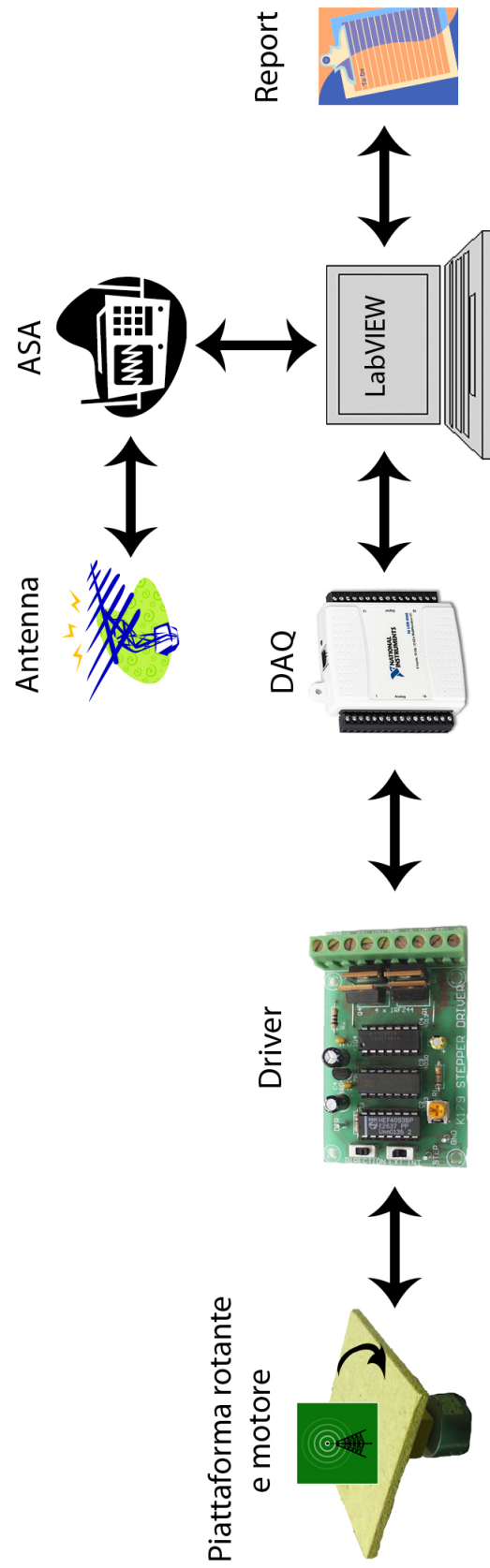


Figura 1.1: Presentazione del sistema di misura.

Dalla fig. 1.1 si nota come l'intero sistema venga gestito per mezzo di un Virtual Instrument realizzato in LabVIEW. Come si vede seguendo le frecce di fig. 1.1, il programma gestisce:

1. una scheda DAQ ( data acquisition ) la quale a sua volta comanda il driver che gestisce la rotazione dei motori passo-passo;
2. un analizzatore di spettro analogico ( ASA, analogic spectrum analyzer ) che attraverso un'antenna acquisisce il segnale emesso dal dispositivo in esame, posto sopra alla piattaforma rotante;
3. un generatore automatico di report che permette il salvataggio delle misure effettuate.

Il pannello frontale dello strumento virtuale che gestisce tutto il sistema è mostrato in fig. 1.2. Nel seguito verranno brevemente introdotti i vari componenti del sistema di misura ed infine verrà descritto il codice realizzato per la gestione del tutto.

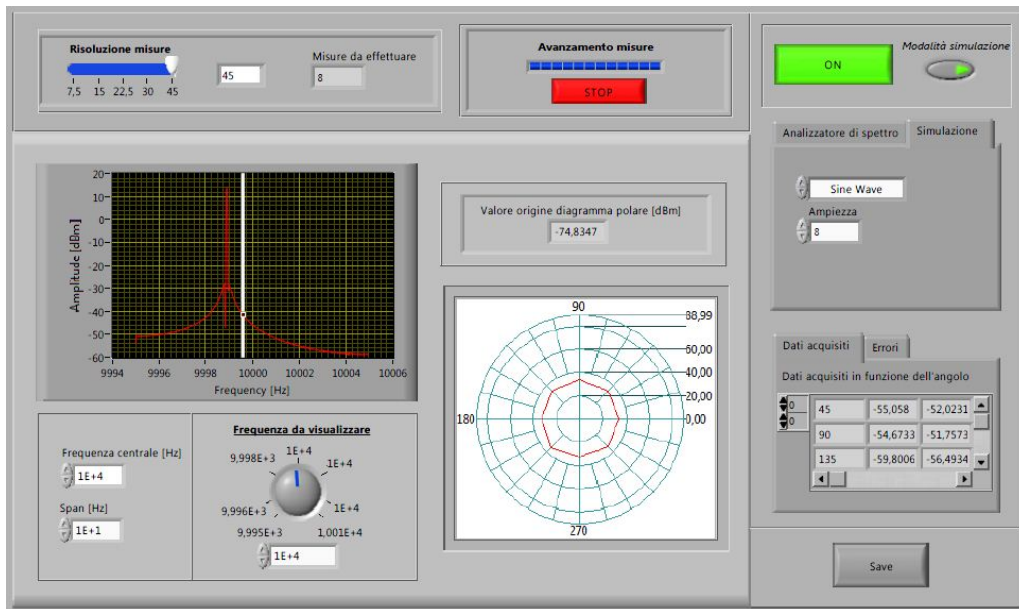


Figura 1.2: Il pannello frontale del VI che controlla il sistema di misura.

## 1.2 Descrizione dei motori passo-passo

I motori passo-passo ( o *stepper motors* ) sono normalmente utilizzati in applicazioni che richiedono una buona precisione e non molta coppia.

Esistono 3 tipi di motori passo-passo:

1. a magnete permanente;
2. a riluttanza variabile;
3. ibridi.

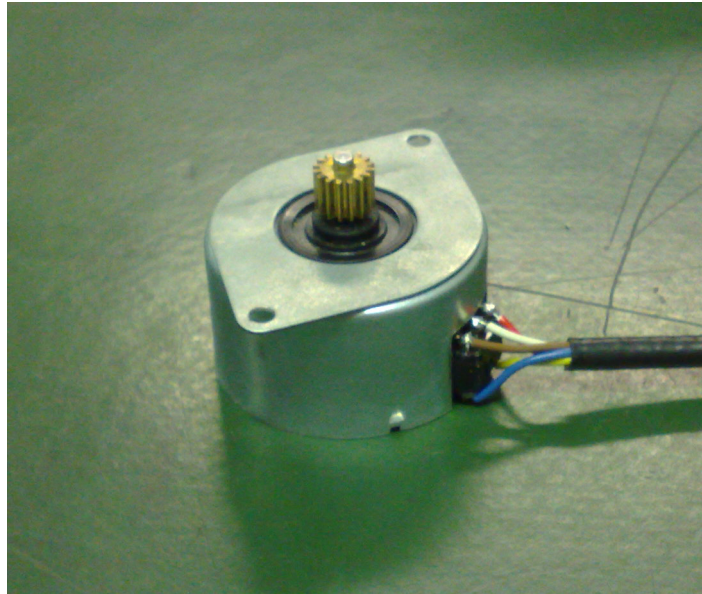


Figura 1.3: Il motore passo-passo utilizzato.

Il motore passo-passo più diffuso risulta essere quello a **magnete permanente**. In esso il magnete svolge la funzione del rotore, ovvero esso è direttamente collegato all'albero che ruota e che sviluppa la forza all'esterno.

Il rotore si muove grazie allo sviluppo di una forza magnetica generata dal passaggio di corrente all'interno di alcuni avvolgimenti situati intorno ad esso<sup>1</sup>. In particolare il passaggio di corrente, generato normalmente applicando un'onda rettangolare, genera uno *step*, ovvero il motore si muove di un "passo". Grazie al fatto che il motore esegue un passo per ogni "impulso" ricevuto, è possibile il controllo della posizione: sapendo di quanti gradi si sposta il motore dopo uno step, è possibile controllare il numero di impulsi da dare affinché l'albero del motore si sposti della quantità desiderata.

Nel caso specifico sull'etichetta è riportato un valore di step pari a 7,5 gradi come si vede dal dettaglio indicato in fig. 1.4; ciò significa che per far ruotare l'albero del motore di un dato angolo  $\alpha$  si dovrà applicare un numero di impulsi pari a:

$$n^o = \frac{\alpha}{step} \quad (1.1)$$

Ad esempio, per  $\alpha = 30$  gradi il numero di step sarà pari a  $30/7,5 = 4$ .

Si è detto che il motore si muove di uno step per ogni impulso di onda rettangolare che viene fornito all'avvolgimento dello statore. Pertanto una serie opportuna di impulsi permette la corretta rotazione continua del motore.

---

<sup>1</sup>Campo magnetico generato da una solenoide:  $B = \mu_0 n i$ , dove B è il campo magnetico, n è il numero di avvolgimenti, i è la corrente circolante e  $\mu_0$  la permeabilità magnetica nel vuoto.

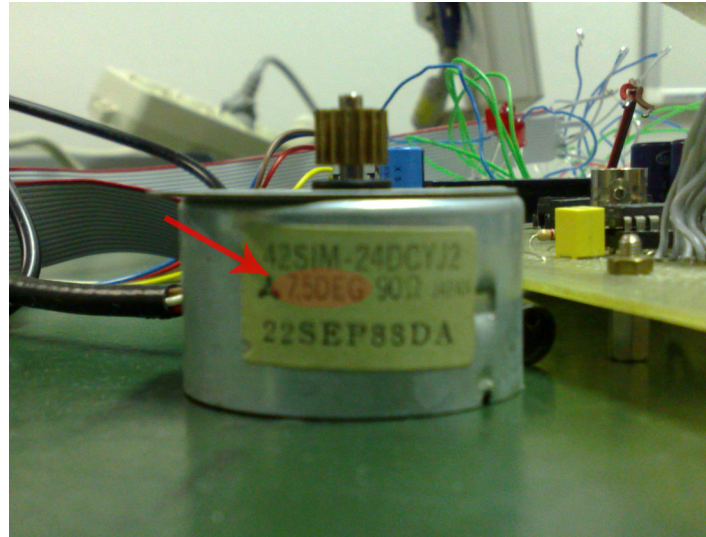


Figura 1.4: Dettaglio dell'etichetta presente sul motore passo-passo.

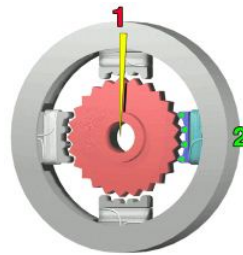


Figura 1.5: Immagine rappresentativa di uno stepper motor.

In fig. 1.5 viene illustrata la modalità per mezzo della quale avviene lo spostamento del rotore. Quando viene data corrente alla spira 1, i denti del rotore si allineano con il campo magnetico generato da tale spira. I denti allineati in 1, non sono però contemporaneamente allineati con 2, pertanto quando verrà generato un campo magnetico in 2, il motore si sposterà ulteriormente per allinearsi con i denti in 2. Il ripetersi in successione di questo ciclo comporta la rotazione.

Pertanto ciascun avvolgimento deve ricevere corrente e generare un campo magnetico in modo sequenziale: le onde che generano tali correnti su ciascuna spira devono dunque essere opportunamente sfasate. Inoltre variando la frequenza dell'onda rettangolare è possibile regolare la velocità di rotazione del motore.

Nel caso dei motori a magnete permanente i "denti" di cui si è parlato sono in realtà dei poli magnetici opportunamente realizzati sulla superficie del rotore. La quantità di poli presenti sulla superficie del rotore influenza direttamente il valore dello step. Infatti gli allineamenti tra gli avvolgimenti sullo statore e i poli magnetici saranno maggiori con un maggior numero di poli, pertanto i "denti" aumenteranno e si potrà avere una maggiore precisione nel movimento. Un'immagine illustrativa è mostrata in fig. 1.6.

I motori passo-passo a **riluttanza variabile** sono costituiti da un rotore composto da un materiale ferro-magnetico. Quando viene applicata tensione allo statore, il campo magnetico che viene generato fa in modo che il rotore si muova verso la posizione con la

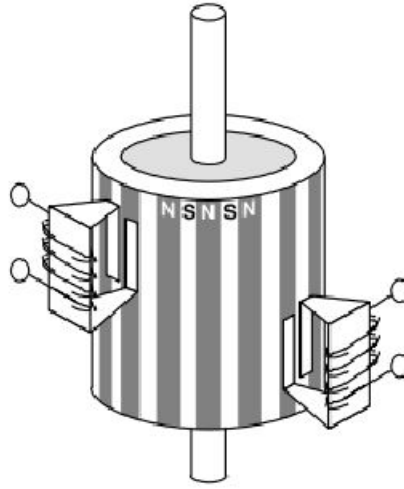


Figura 1.6: Il motore passo-passo a magnete permanente.

minima riluttanza.<sup>2</sup>

La riluttanza è definita nell'eq. 1.3: in tale equazione si vede come essa dipenda direttamente dalla distanza tra il rotore e lo statore. Tale distanza è chiamata *air gap*, letteralmente *buco o passaggio nell'aria*, ed è infatti la distanza che il campo magnetico, generato dallo statore, deve percorrere nell'aria prima di indurre un campo magnetico nel rotore.

$$R = \int_{l_{\text{air gap}}} \frac{dl}{\mu S} \quad (1.3)$$

Pertanto, considerando la sezione  $S$  e la permeabilità magnetica dell'aria  $\mu$  costante nell'area tra il rotore e lo statore, l'eq. 1.3 diventa:<sup>3</sup>

$$R = \frac{l_{\text{air gap}}}{\mu S} \quad (1.5)$$

Dall'eq. 1.5 si vede come la minima riluttanza si abbia quando  $l_{\text{air gap}}$  risulti minimo. Il principio di funzionamento è illustrato in fig. 1.7.

<sup>2</sup>La riluttanza è inversamente proporzionale alla permeanza: il campo magnetico indotto nel metallo risulta pertanto essere maggiore dove la riluttanza è minore, sviluppando in tale zona una maggiore forza di attrazione con lo statore.

$$B = \mu H \quad (1.2)$$

dove  $B$  è il campo magnetico indotto e  $H$  è il campo magnetico generato dagli avvolgimenti.

<sup>3</sup> $S$  è la sezione nella quale si considera che si sposti il campo magnetico. In particolare, introducendo il concetto di flusso:

$$\Phi = \int_S \vec{H} \vec{n} dS \quad (1.4)$$

dove  $\vec{H}$  è il vettore campo magnetico e  $\vec{n}$  è il versore normale alla superficie  $S$ .



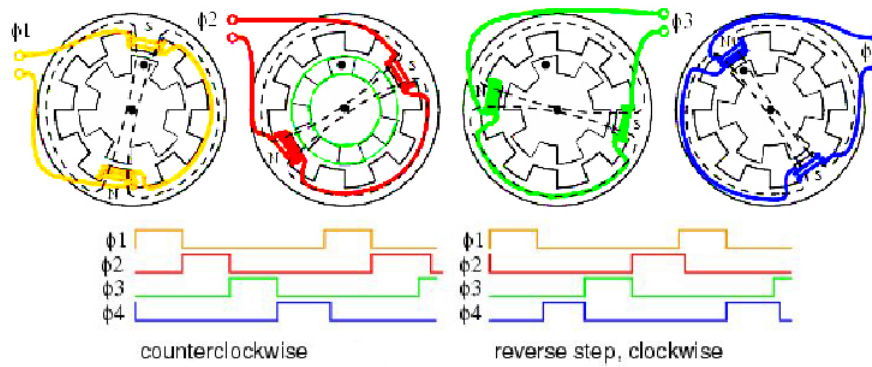


Figura 1.7: Il motore passo-passo a riluttanza variabile.

Risulta evidente che, quando una fase è attiva ( ovvero quando viene fatta passare corrente su una determinata coppia di avvolgimenti ) il rotore si sposta affinché la distanza tra gli avvolgimenti attivi e il rotore stesso sia minima. In questo modo il campo magnetico generato percorrerà il minor percorso possibile attraverso l'aria. Quando viene attivata la seconda fase il motore non sarà completamente allineato con gli avvolgimenti e pertanto si avrà un'ulteriore rotazione. Proseguendo con gli altri passi mostrati in fig. 1.7 e ripetendo il ciclo si avrà la rotazione.

I motori passo-passo **ibridi** sono caratterizzati da un rotore a magnete permanente con diverse espansioni ferromagnetiche. Sono i motori più performanti in termini di precisione, velocità e coppia, ma anche i più costosi. La struttura interna è riportata schematicamente in fig. 1.8.

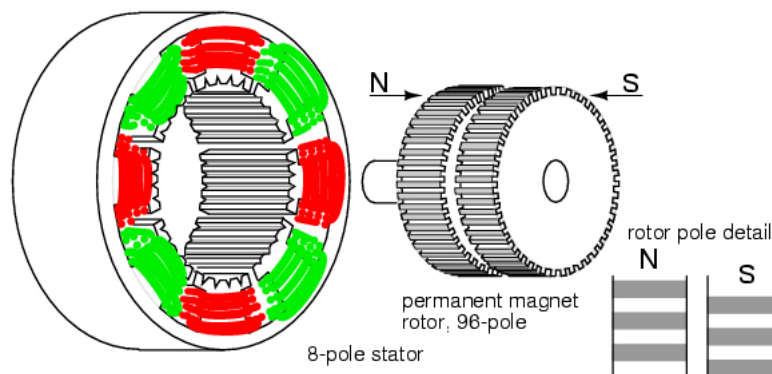


Figura 1.8: Il motore passo-passo ibrido.

### 1.2.1 Driver

Il driver è stato realizzato in modo tale da convertire i segnali generati dalla scheda DAQ in segnali adatti al pilotaggio dei motori.

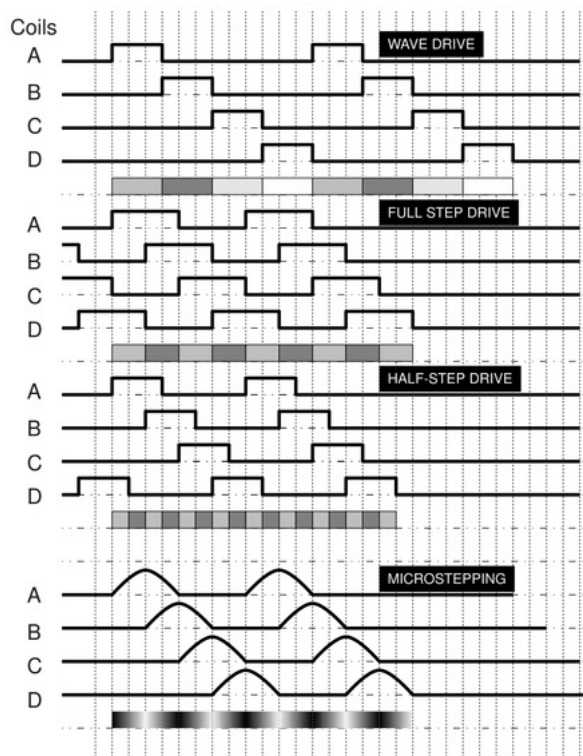


Figura 1.9: Diversi tipi di modalità di controllo di un motore passo-passo a 4 fasi.

Come si vede in fig. 1.9, a seconda di come le onde di ciascuna fase vengono generate dal driver, sono disponibili diversi tipi di controllo per pilotare un motore passo-passo. Si possono utilizzare le modalità:

1. wave drive;
2. full step drive;
3. half step drive;
4. microstepping.

La modalità wave drive, nella quale le due fasi consecutive non si intersecano, è quella utilizzata dal driver del sistema di misura impiegato. Il programma che pilota la scheda DAQ infatti non attiva mai due fasi contemporaneamente. Questa modalità è la peggiore per quanto riguarda la coppia sviluppata, infatti il motore tende a muoversi a scatti e questo danneggia lo sviluppo di coppia.

Con le altre modalità la difficoltà maggiore risiede nella costruzione del driver. Si ha in compenso un aumento della coppia e della risoluzione, soprattutto nella modalità half step e microstepping.

### 1.3 Descrizione scheda DAQ

La scheda utilizzata in laboratorio è la National Instruments 6008-USB. Tale dispositivo ha nel progetto lo scopo di interfacciarsi con il driver e di generare le fasi che controllano



Figura 1.10: Scheda DAQ utilizzata per il progetto.

il motore.

Essa può acquisire dati dall'esterno o generare segnali analogici/digitali. La scheda è dotata di:

- otto ingressi analogici;
- due uscite analogiche;
- dodici ingressi/uscite digitali;
- un contatore a 32 bit;
- un'interfaccia di collegamento USB.

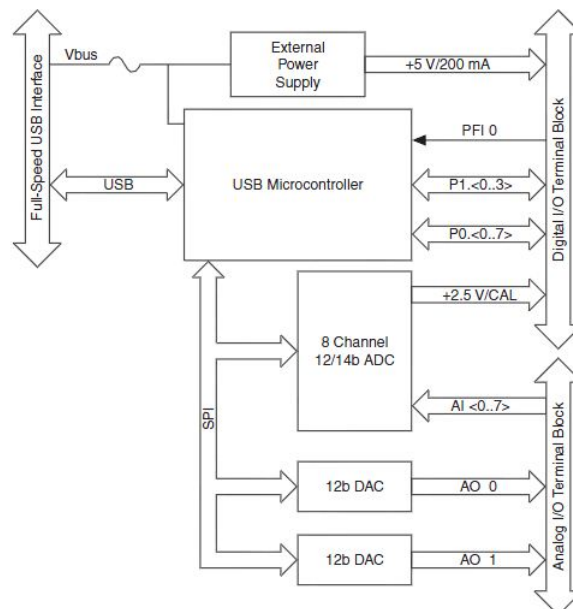


Figura 1.11: Lo schema a blocchi del dispositivo DAQ.

La fig. 1.11 rappresenta uno schema a blocchi del funzionamento del DAQ. In esso si vede come tutti i segnali in ingresso e in uscita passino attraverso un micro-controllore USB pertanto gli eventuali segnali analogici in ingresso saranno opportunamente convertiti da un ADC (*analogue to digital converter*) mentre i segnali analogici in uscita saranno convertiti da un DAC (*digital to analogue converter*).

Nella fig. 1.11 viene specificato l'utilizzo di DAC e di ADC a 12 bit.<sup>4</sup>

I pin di I/O presenti sono indicati in fig. 1.12 e fig. 1.13.

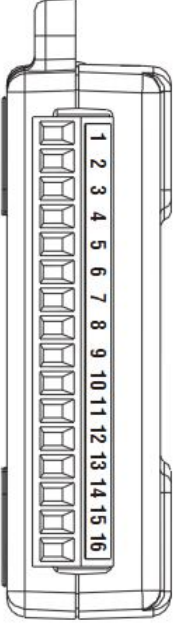
Module	Terminal	Signal, Single-Ended Mode	Signal, Differential Mode
	1	GND	GND
	2	AI 0	AI 0+
	3	AI 4	AI 0-
	4	GND	GND
	5	AI 1	AI 1+
	6	AI 5	AI 1-
	7	GND	GND
	8	AI 2	AI 2+
	9	AI 6	AI 2-
	10	GND	GND
	11	AI 3	AI 3+
	12	AI 7	AI 3-
	13	GND	GND
	14	AO 0	AO 0
	15	AO 1	AO 1
	16	GND	GND

Figura 1.12: I terminali analogici.

Facendo riferimento alla fig. 1.12, tutti i pin AI 0, ..., AI 7 possono essere utilizzati indifferentemente come pin d'ingresso analogici. Se si vuole utilizzare la modalità di acquisizione differenziale, la *coppia AI 0 differenziale* sarà data da AI 1 e AI 4 che saranno, rispettivamente, AI 0+ e AI 0-. Le altre coppie per acquisizioni differenziali sono AI 1 - AI 5, AI 2 - AI 6 e AI 3 - AI 7.

AO 0 e AO 1 sono invece le uscite analogiche.

In fig. 1.13 invece sono indicati i terminali di I/O digitali. Sia i terminali P1 0, ..., P1 3 che i terminali P0 0, ..., P0 7 sono configurabili singolarmente affinché fungano da input o da output.

<sup>4</sup>L'immagine è presa dallo *user guide* dei modelli 6008/6009: il valore in fig. 1.11 per il DAC con 14 bit è relativo al modello 6009.

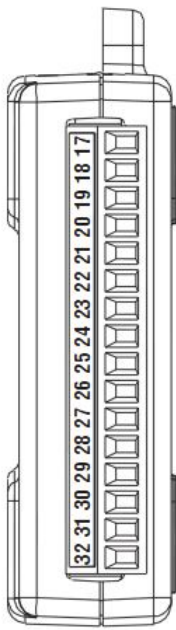
Module	Terminal	Signal
	17	P0.0
	18	P0.1
	19	P0.2
	20	P0.3
	21	P0.4
	22	P0.5
	23	P0.6
	24	P0.7
	25	P1.0
	26	P1.1
	27	P1.2
	28	P1.3
	29	PFI 0
	30	+2.5 V
	31	+5 V
	32	GND

Figura 1.13: I terminali digitali.

I pin 5 V e 2,5 V forniscono in uscita tale tensione, mentre il pin con etichetta GND fornisce un collegamento a terra.

### 1.3.1 Gli ingressi analogici

Il circuito degli ingressi analogici è mostrato in fig. 1.14.

AI ( *analogue input* ) è il pin collegato con l'ingresso in acquisizione. Esso passa attraverso un multiplexer, il quale fa in modo che venga processato un solo segnale analogico per volta.

Il PGA ( *programmable-gain amplifier* ) amplifica l'ingresso a seconda della modalità di funzionamento selezionata:

- è possibile scegliere un'amplificazione di un fattore 1, 2, 4, 5, 8, 10, 16, 20 se siamo in modalità differenziale, ovvero se stiamo considerando la differenza di tensione tra due segnali;
- è possibile scegliere un'amplificazione solo di un fattore 1 ( cioè il segnale resta lo stesso ) se siamo in modalità di acquisizione di un singolo segnale.

Il convertitore ADC è ad approssimazioni successive e verrà mostrato più nel dettaglio nella sezione "ADC ad approssimazioni successive".

Dopo la conversione il segnale digitale viene passato ad una AI FIFO ( *first in first out* ) che gestisce i dati in ingresso in modo tale che il primo dato acquisito sia il primo ad essere processato. Il buffer collegato a questa struttura assicura che non vi siano perdite

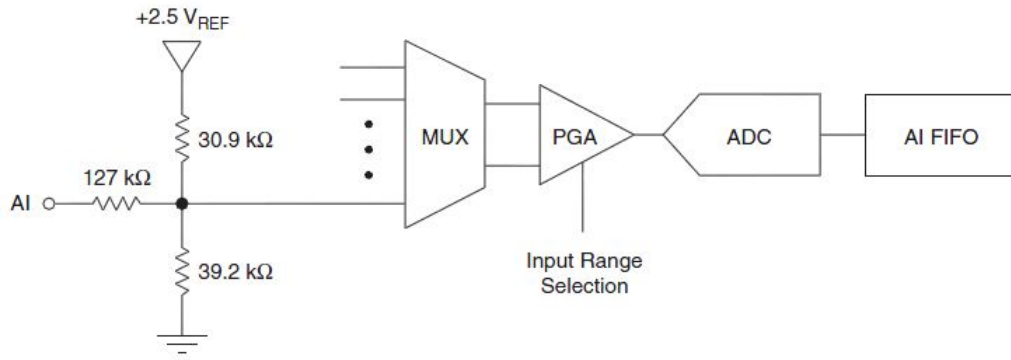


Figura 1.14: Circuito degli ingressi analogici.

di dati.

Se si utilizza la funzione di acquisizione di un segnale analogico, è possibile settare l'ingresso  $PFI\ 0$  come trigger digitale, in particolare vi è la possibilità di impostarlo sul fronte di salita o sul fronte di discesa di un'onda rettangolare.

Il limite in tensione per i canali d'ingresso analogici è di  $\pm 10\text{ V}$  ( riferiti a massa ) e l'impedenza equivalente d'ingresso, fornita dal documento [3], è pari a  $144\text{ k}\Omega$ . Pertanto la massima corrente entrante sarà:

$$\frac{10}{144 \cdot 10^3} = 69,4\ \mu\text{A} \quad (1.6)$$

### 1.3.2 Uscite analogiche

La scheda NI 6008 possiede due uscite analogiche indipendenti tra loro. Queste possono generare segnali nel range tra 0 e 5 V, come si vede dal circuito di uscita del segnale analogico presentato in fig. 1.15. Il convertitore DAC traduce un segnale digitale

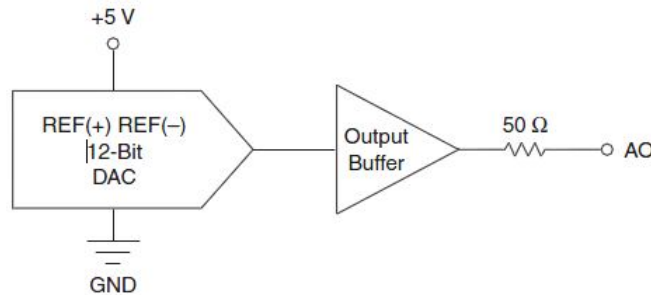


Figura 1.15: Circuito delle uscite analogiche.

definito da  $N$  bit in un segnale analogico  $\hat{V}$ .

La conversione viene effettuata qualitativamente attraverso l'eq:

$$\hat{V} = (b_{N-1} \cdot 2^{N-1} + b_{N-2} \cdot 2^{N-2} + \dots + b_1 \cdot 2^1 + b_0 \cdot 2^0) \cdot \frac{V_R}{2^N} \quad (1.7)$$

dove  $V_R$  ( tensione di riferimento ) è in questo dispositivo pari a  $5\text{ V}$  e  $N$  è il numero di bit del convertitore. Ciascun bit nell'eq. 1.7 è indicato con  $b_i$ , dove  $i$  è l'indice che si riferisce

alla posizione del bit (0=LSB, N-1=MSB). Il convertitore DAC è caratterizzato da una precisione di 12 bit, si potranno pertanto specificare i segnali analogici da generare fino ad una precisione di:

$$\frac{5}{2^{12}} = 0,0012V \quad (1.8)$$

La resistenza d'uscita, indicata in [3], è pari a  $50 \Omega$ , mentre la tensione del segnale generato può variare da 0 a 5 V.

Rappresentando la scheda DAQ con una rappresentazione equivalente, l'impedenza d'ingresso equivalente e la resistenza d'uscita per i segnali analogici possono essere rappresentate attraverso un doppio bipolo, si veda fig. 1.16.

- $R_{eq} = 144 \text{ k}\Omega$
- $R_{out} = 50 \Omega$

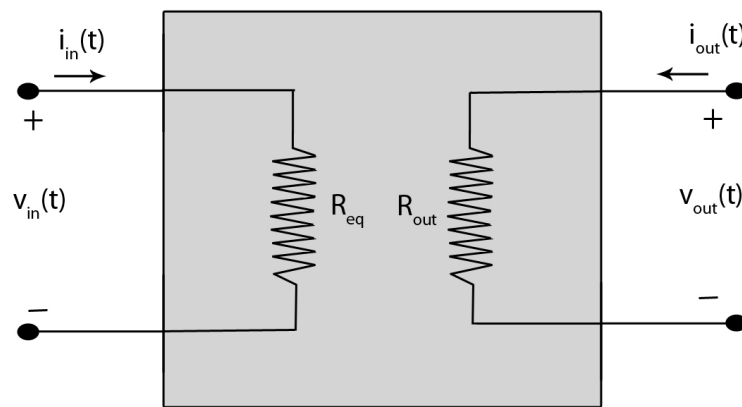


Figura 1.16: Doppio bipolo equivalente.

Le tensioni  $v_{in}$  e  $v_{out}$  sono rispettivamente i segnali analogici entranti e uscenti dal dispositivo. Mentre le correnti  $i_{in}$  e  $i_{out}$  sono le correnti rispettivamente entranti ed uscenti.

### 1.3.3 I/O digitali

I segnali di input/output digitali sono quelli utilizzati nel progetto. In particolare nel programma sono configurati come output in quanto hanno la funzione di comandare un dispositivo esterno.

La configurazione di default di queste connessioni è di tipo *open collector*, avente una resistenza di pull-up di  $4,7 \text{ k}\Omega$  collegata ad una tensione di 5 V. Pertanto i pin dopo l'inizializzazione della scheda sono a livello logico alto.

### 1.3.4 Collegamento tra DAQ e driver

Il collegamento tra DAQ e driver è stato effettuato utilizzando lo schema in fig. 1.17. Si è collegata una resistenza aggiuntiva tra il pin che fornisce la tensione di 5 V e quello dell'uscita digitale per la quale si richiede una maggiore corrente.

Tale schema infatti è riportato in [3] e permette una maggiore corrente in uscita, necessaria per pilotare correttamente i componenti presenti nel driver.

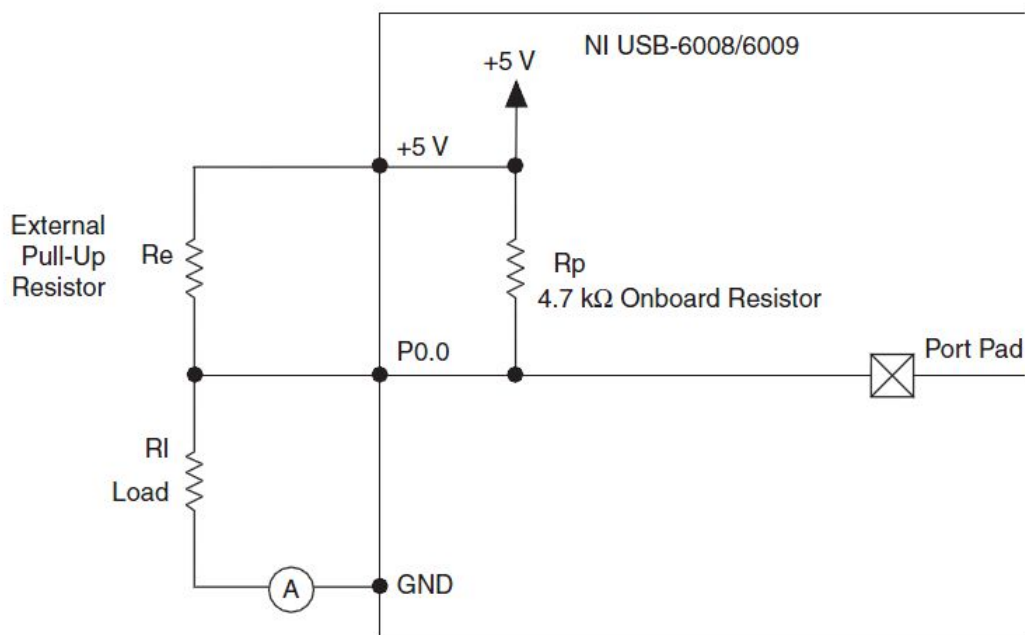


Figura 1.17: Circuito che permette un incremento della corrente in uscita.

### 1.3.5 Convertitore ADC ad approssimazioni successive

Un convertitore ad approssimazioni successive converte un segnale analogico  $V_X$  in un segnale digitale; tale operazione viene eseguita attraverso un ciclo che rende, ogni nuova iterazione, il segnale convertito sempre più vicino all'originale.

La conversione del valore  $V_X$  avviene in un numero di passi pari al numero di bit del convertitore.

Partendo dalla scelta del bit più significativo, verranno processati durante i vari passi del funzionamento i bit meno significativi, fino al LSB. Ad ogni bit verrà assegnato uno 0 o un 1 logico utilizzando un comparatore che confronta un valore di riferimento  $V_T$ , funzione dei bit che stiamo convertendo, con il valore da convertire  $V_X$ . Un'immagine rappresentante il funzionamento del convertitore è mostrata in fig. 1.18.



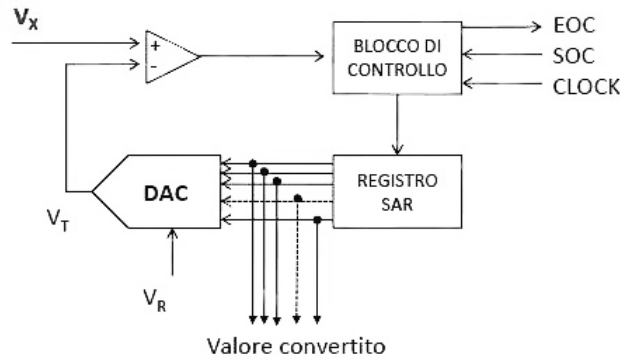


Figura 1.18: Schema del funzionamento di un convertitore ADC ad approssimazioni successive.

Il processo di conversione può essere schematizzato nei seguenti punti ( per la notazione si faccia riferimento alla fig. 1.18 ).

1. Il segnale SOC ( *start of conversion* ) inizializza il sistema: il registro SAR ( *successive approximation register* ) si trova nello stato 1000...00 ( MSB=1 e tutti gli altri bit a zero );
2. Il DAC converte il valore binario presente nel SAR, fornendo un primo livello di soglia logica con cui confrontare  $V_X$ .

$$V_T = (b_{N-1} \cdot 2^{N-1} + b_{N-2} \cdot 2^{N-2} + \dots + b_1 \cdot 2^1 + b_0 \cdot 2^0) \cdot \frac{V_R}{2^N} \quad (1.9)$$

dove N è il numero di bit di cui è dotato il convertitore.

Nella prima iterazione, ponendo  $b_{N-1} = 1$  e tutti gli altri coefficienti a zero, ottengo dall'eq. 1.9, un valore  $V_T$  pari a metà fondo scala, infatti:

$$2^{N-1} \cdot \frac{V_R}{2^N} = \frac{V_R}{2} \quad (1.10)$$

3. Il comparatore esegue il confronto tra  $V_X$  e  $V_T$ :
  - se  $V_X > V_T$  il primo bit del SAR rimane a 1;
  - altrimenti il bit viene azzerato.
4. Si considera quindi il secondo bit del SAR che viene portato a 1 e si procede come nei punti 2) e 3) fino al LSB.
5. La fine della conversione è indicata dal segnale EOF ( *end of conversion* )

Al termine della conversione il valore convertito sarà formato da tutti i coefficienti  $b_i$  ricavati, in particolare il valore approssimato sarà:

$$\hat{V}_X = (b_{N-1} \cdot 2^{N-1} + b_{N-2} \cdot 2^{N-2} + \dots + b_1 \cdot 2^1 + b_0 \cdot 2^0) \cdot \Delta_q \quad (1.11)$$

dove  $\Delta_q = \frac{V_R}{2^N}$  è il quanto e rappresenta l'incertezza nella conversione.

Il tempo necessario per la conversione del valore analogico è indipendente dal valore da convertire ma dipende solo dal numero N di bit del convertitore.

$$T_{\text{conv}} = N \cdot T_f \quad (1.12)$$

dove  $T_{\text{conv}}$  è il tempo necessario per effettuare l'intera conversione, mentre  $T_f$  il tempo per eseguire un confronto.

## 1.4 Analizzatore di spettro

L'analizzatore di spettro utilizzato in laboratorio è il modello Agilent E4407B, mostrato in fig. 2.1. Tale strumento ha lo scopo di acquisire i valori delle radiazioni emesse da una sorgente, in funzione della frequenza. Lo schermo avrà lungo l'asse delle ascisse i valori delle frequenze e lungo le ordinate le rispettive ampiezze. Il funzionamento di questo strumento visto con maggior dettaglio verrà mostrato nel capitolo 1.

## 1.5 Funzionamento del sistema

Il programma deve gestire e sincronizzare l'acquisizione delle radiazioni e la rotazione dei motori.

La gestione dei motori viene effettuata attraverso un programma che, dati il numero di passi da effettuare e la pressione del tasto *avvia*, genera i segnali per la rotazione.

Per semplicità, nella prima versione del progetto, la piattaforma è stata posizionata direttamente sopra al motore, pertanto una rotazione del motore comporta la rotazione della piattaforma e della sorgente sopra essa in rapporto diretto 1:1.

Il programma che gestisce i motori è composto da un ciclo che viene ripetuto un numero di volte pari al numero di step da compiere.

Il motore utilizzato ha 5 fili: uno è la massa comune e gli altri comandano ciascuno il passaggio di corrente su un avvolgimento attorno al rotore. Il programma che lo gestisce fa in modo che venga attivata una fase alla volta, in modo sequenziale.

Il procedimento del programma schematizzato è il seguente.

1. Inizializzazione del motore: una fase alta e tutte le altre basse. Nel programma LabVIEW i segnali che gestiscono i 4 fili sono di tipo booleano, pertanto uno sarà posto a *true* mentre gli altri saranno posti *false*. Schematicamente si avrà:

	1	2	3	4
segnali	T	F	F	F

Tabella 1.1

2. Il movimento del motore si avrà portando a *true* la fase successiva e le altre a *false*, questo comporterà uno step. Si veda tab. 1.2

	1	2	3	4
segnali	F	T	F	F

Tabella 1.2

3. Allo step successivo si avrà la situazione presentata in tab. 1.3.

	1	2	3	4
segnali	F	F	T	F

Tabella 1.3

4. Il quarto passo porterà alla tab. 1.4

	1	2	3	4
segnali	F	F	F	T

Tabella 1.4

	1	2	3	4
segnali	T	F	F	F

Tabella 1.5

5. Infine si otterrà la situazione di partenza, ripresentata nella tab. 1.5

Ripetendo ciclicamente i passi 2), 3), 4) e 5) si avrà la rotazione del motore.

Quando i passi richiesti dal programma che gestisce il motore sono stati completati, il programma di gestione del sistema può proseguire la sua esecuzione andando ad acquisire le radiazioni emesse tramite l'analizzatore di spettro.

L'ASA restituirà le ampiezze delle componenti alle varie frequenze ( in un array ) che verranno gestite dal programma e inserite in una matrice, nella quale ogni nuovo array acquisito sarà una nuova riga della matrice. Poichè l'acquisizione dallo strumento viene eseguita solo dopo la rotazione del motore, ogni riga della matrice rappresenta le emissioni irradiate in funzione dell'angolo di rotazione del motore. In particolare lo spostamento del motore è stato opportunamente tarato affinché la rotazione globale fosse di 360 gradi e le emissioni con il relativo spostamento potessero essere pensate come coordinate polari di uno grafico.

I dati acquisiti e il grafico polare, che costituiscono i dati di interesse per la misura, possono essere salvati in un report. Il salvataggio del report di misura avviene in formato html. Questo formato presenta il vantaggio di essere portabile in tutti i sistemi operativi in quanto è sufficiente un browser per visualizzarlo. Potrebbe essere inoltre già adatto ad un'eventuale pubblicazione sul web.



## Capitolo 2

# Analizzatore di spettro



Figura 2.1: Analizzatore di spettro utilizzato in laboratorio ( Agilent E4407B ).

## 2.1 Introduzione teorica sui segnali

### 2.1.1 Trasformata di Fourier

Un generico segnale  $s(t)$  può essere descritto analizzandone l'andamento in funzione del tempo. Tale segnale può essere anche espresso, idealmente senza alcuna perdita d'informazione, in funzione di un'altra grandezza: la frequenza.

La teoria di Fourier dimostra come un segnale, *periodico* nel tempo, possa essere espresso come *sommatoria* di opportune componenti sinusoidali. Se chiamiamo il segnale  $s(t)$  si avrà:

$$s(t) = \sum_{n=-\infty}^{+\infty} S_n e^{j2\pi f_n t}, \quad f_n = nF = n\frac{1}{T}, \quad n \in \mathbb{Z} \quad (2.1)$$

dove gli  $S_n$  sono chiamati *coefficienti di Fourier* e sono definiti come:

$$S_n = \frac{1}{T} \int_{-T/2}^{T/2} s(t) e^{-j2\pi f_n t} dt \quad (2.2)$$

Si nota come in questo caso particolare di periodicità nel tempo, il segnale possa essere espresso attraverso una sommatoria di componenti *discrete* in frequenza; in particolare il quanto in frequenza (  $F$  ) risulta essere pari al reciproco del periodo.

Nel caso di segnali non periodici si dovrà passare ad una trattazione continua in frequenza, utilizzando pertanto l'integrale.

In analogia con  $S_n$  ci sarà una funzione  $S(f)$ , chiamata **trasformata di Fourier**, definita in questo modo:

$$S(f) = \int_{-\infty}^{+\infty} s(t)e^{-j2\pi ft} dt, \quad f \in \mathbb{R} \quad (2.3)$$

mentre il segnale  $s(t)$  sarà espresso attraverso la cosiddetta **antitrasformata di Fourier**:

$$s(t) = \int_{-\infty}^{+\infty} S(f)e^{j2\pi ft} df, \quad t \in \mathbb{R} \quad (2.4)$$

### 2.1.2 Basi teoriche sulle onde trasmesse

Una generica informazione è trasmessa nello spazio facendo variare una qualche caratteristica fisica del mezzo di trasmissione. Tale variazione viene propagata all'interno del mezzo con una certa velocità fino a raggiungere il ricevente, il quale si occupa di interpretare tale variazione correttamente al fine di ottenere l'informazione.

Il fenomeno fisico che viene utilizzato è l'**onda elettromagnetica**, ossia una combinazione di campo magnetico e campo elettrico variabili nel tempo; tale onda si propaga dalla *sorgente* alla *destinazione* in modi diversi a seconda del mezzo utilizzato.

Se rappresentiamo il valore del fenomeno fisico caratterizzante l'informazione, attraverso una generica funzione  $s(t)$ <sup>1</sup> variabile nel tempo, si ha che esso potrà essere studiato matematicamente. Supponendo ad esempio che  $s(t)$  sia un segnale periodico, esso sarà esprimibile attraverso una sommatoria di infinite componenti sinusoidali:

$$s(t) = a_0 + \sum_{n=1}^{\infty} a_n \sin(2\pi nFt) + \sum_{n=1}^{\infty} b_n \cos(2\pi nFt) \quad (2.5)$$

dove  $a_n$  e  $b_n$  sono le ampiezze delle armoniche di ordine  $n$ , ovvero delle onde con frequenza pari a  $n$  volte la frequenza fondamentale ( che si ottiene per  $n = 1$  );  $a_0$  è il valore medio o la componente continua.

## 2.2 Introduzione

### 2.2.1 Confronto tra dominio del tempo e della frequenza

I fenomeni fisici sono normalmente osservabili nel dominio del tempo. Per visualizzare l'andamento di un'onda in funzione del tempo si potrà utilizzare un oscilloscopio; questo tipo di approccio "temporale" risulta adatto ad esempio al calcolo di tempi di salita e/o di discesa, individuazione di picchi, ecc. . .

Ricavare la rappresentazione equivalente di un segnale nel suo dominio della frequenza è altrettanto importante e risulta utile in diversi ambiti:

<sup>1</sup> $s(t)$  può rappresentare ad esempio il valore di un campo elettrico, di una corrente o di tensione.

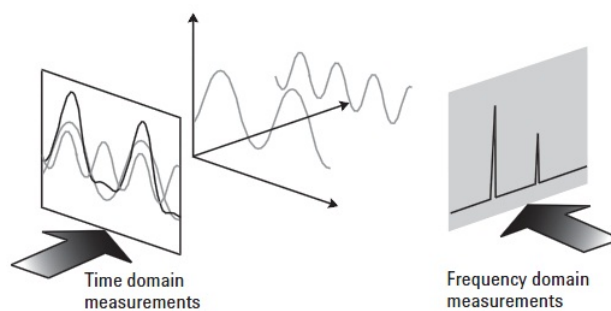


Figura 2.2: Relazione tra il dominio del tempo e della frequenza

- Nel campo della compatibilità elettromagnetica <sup>2</sup> vengono effettuate accurate misure delle emissioni di un dispositivo. Queste onde non volute, siano esse irradiate o trasmesse su cavo, possono comportare malfunzionamenti dovuti ad interferenze con dispositivi adiacenti. Praticamente ogni nuovo dispositivo prima di essere introdotto nel mercato deve effettuare un test che misuri l'emissione in funzione della frequenza e deve soddisfare le normative imposte dallo Stato per tale tipo di applicazione.
- L'occupazione in banda di un dispositivo è inoltre di fondamentale importanza. Lo Stato si occupa di assegnare determinate frequenze a ciascun tipo di applicazione, anche in funzione delle necessità legate al proprio funzionamento. È importante che le emissioni rimangano all'interno del livello stabilito: emissioni provenienti da canali adiacenti potrebbero causare infatti nel complesso un segnale disturbato da rumore.
- Individuare la presenza di armoniche all'interno dello spettro risulta importantissimo soprattutto per chi opera con sistemi di trasmissione wireless. Ad esempio, per i dispositivi cellulari, dev' essere controllata la presenza di armoniche che potrebbero interferire con altri sistemi operanti alla stessa frequenza delle armoniche.

### 2.2.2 Tipi di misurazioni

Nei normali analizzatori di spettro vi è la possibilità di misurare frequenze di emissione, potenza, distorsione e rumore.

La quantità di potenza trasmessa è sicuramente un importante parametro da valutare: poca potenza potrebbe non essere sufficiente a far arrivare il segnale a destinazione; troppa potenza invece potrebbe consumare eventuali batterie rapidamente e causare temperature elevate.

### 2.2.3 Schermo

Il display è suddiviso in regioni delimitate da una griglia: vi sono 10 divisioni sull'asse orizzontale e generalmente 10 anche sull'asse verticale. L'asse orizzontale è linearmente calibrato in modo tale che la frequenza aumenti da sinistra verso destra. Settare la frequenza è un processo a 2 passi: prima si imposta la frequenza centrale, poi l'intervallo

<sup>2</sup>EMI: ElectroMagnetic Interference.

( *span* ) visualizzato. Ovviamente questi controlli sono indipendenti uno dall'altro e in alternativa è normalmente possibile settare frequenza iniziale e finale.

L'asse verticale viene calibrato per l'ampiezza. Possiamo scegliere di norma tra scala lineare e logaritmica [dB]; quest'ultima è normalmente la più utilizzata in quanto permette di rappresentare contemporaneamente un più ampio intervallo di frequenze. Si imposta un valore detto *di riferimento* all'ultima linea in alto sull'asse verticale e in funzione di tale valore si ricavano tutti i valori intermedi<sup>3</sup>.

## 2.3 Architettura e funzionamento base

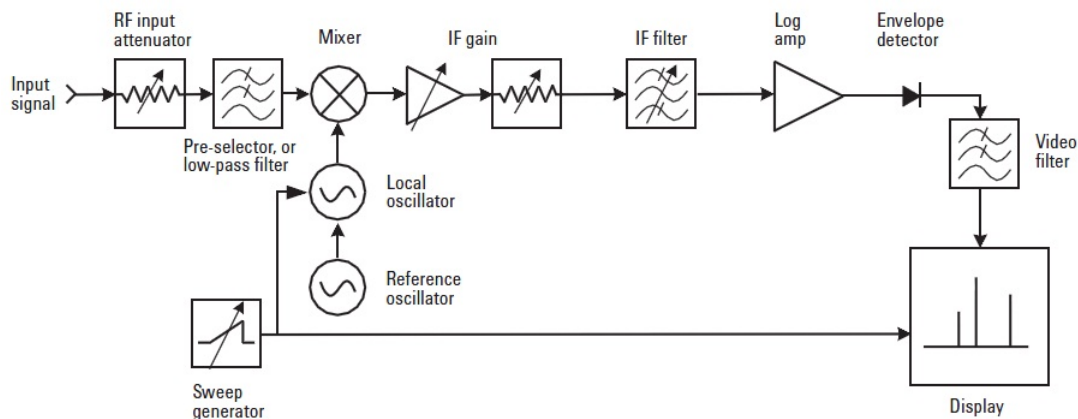


Figura 2.3: Architettura di un analizzatore di spettro analogico.

In fig. 2.3 è riportata l'architettura di un analizzatore di spettro.

Il segnale di cui si vuole misurare lo spettro, viene fornito in ingresso allo strumento. I primi due stadi dell'architettura, composti da un attenuatore a radio frequenza e da un eventuale filtro preselettore passa basso, servono per adattare il segnale acquisito alle caratteristiche del successivo mixer. Questo blocco opera una traslazione in frequenza del segnale ricevuto, in funzione della frequenza del segnale fornito dall'oscillatore locale e a sua volta gestito dal generatore di rampa (sweep generator). Dopo la traslazione in frequenza, la componente del segnale che viene a trovarsi alla stessa frequenza  $f_{IF}$  del filtro selettivo a frequenza intermedia (intermediate frequency filter IF) produrrà, in uscita da tale filtro, un segnale sinusoidale avente frequenza  $f_{IF}$  (supponendo di poter considerare infinitesima la banda passante del filtro) ed ampiezza funzione dell'ampiezza della componente a frequenza  $f_{LO} - f_{IF}$  del segnale di ingresso. Tale ampiezza viene misurata per mezzo di un rivelatore di inviluppo posto in uscita dal filtro selettivo e viene successivamente visualizzata sul display dello strumento alla corrispondente frequenza.

Un generatore di rampe genera le variazioni sulla scala orizzontale del display, da sinistra verso destra.

In funzione della rampa viene regolato anche il LO in modo tale che la sua frequenza sia proporzionale al corrispondente valore sulla scala orizzontale.

La frequenza dell'IF viene scelta in modo tale che essa risulti maggiore della massima frequenza misurabile del segnale in ingresso.

<sup>3</sup>Valore di riferimento/10=Incremento per ogni riga.



Se un analizzatore di spettro deve poter sintonizzarsi su frequenze fino a  $x$  Ghz si dovrà rendere disponibile la sintonizzazione fino a  $x$  Ghz attraverso l'utilizzo della seguente relazione, che verrà approfondita nella sezione *sintonizzare l'analizzatore*:

$$f_{sig} = f_{LO} - f_{IF} \quad (2.6)$$

dove  $f_{sig}$  è la frequenza del segnale,  $f_{LO}$  la frequenza dell'oscillatore locale e  $f_{IF}$  la frequenza intermedia.

### 2.3.1 RF attenuator

Lo scopo di questo dispositivo è di assicurare che il segnale entri nel mixer al livello ottimale: prevenendo ad esempio sovraccarico, distorsione. . .

Normalmente il valore di attenuazione è regolata in base al *livello di riferimento* scelto. Nell'architettura dell'RF vi è normalmente anche una capacità, utilizzata per prevenire eventuali danni all'analizzatore, che verrebbero causati dall'utilizzo di un segnale DC in ingresso. Sfortunatamente, tale capacità alza il minimo valore della frequenza iniziale analizzabile dallo strumento.

### 2.3.2 Low-pass filter or preselector

Il filtro passa-basso blocca segnali ad alta frequenza prima di raggiungere il mixer. In questo modo si evita che segnali fuori dalla banda di interesse vengano mixati e eventualmente accoppiati con la banda passante dell'IF.

### 2.3.3 Sintonizzare l'analizzatore

La sintonizzazione dello strumento dipende principalmente da 3 parametri: la frequenza centrale dell'IF filter, l'intervallo di frequenze assumibili dal LO e le frequenze che possono raggiungere il mixer dall'esterno ( attraverso il low-pass filter ).

Fra tutti i prodotti uscenti dal mixer, i due contributi di interesse sono dati dalla somma e dalle differenza del segnale d'ingresso con il LO. A seconda della frequenza dell'oscillatore locale, le componenti che passano attraverso l'IF filter sono date da:

$$f_{IF} = f_{LO} + f_{sig} \quad (2.7)$$

$$f_{IF} = f_{LO} - f_{sig} \quad (2.8)$$

Quindi, a seconda della scelta tra somma e differenze delle frequenze, basterà fare in modo che la frequenza del segnale sia uguale a quella del LO più o meno IF.

Se supponiamo di volere  $f_{LO} - f_{sig}$  sulla banda passante dell'IF, come in fig. 2.4, allora di dovrà avere, dalla eq. 2.8, che  $f_{LO} = f_{IF} + f_{sig}$ .

### 2.3.4 IF gain

L'amplificatore a guadagno variabile viene utilizzato per regolare la posizione verticale dei segnali sul display; il segnale all'ingresso del mixer resta inalterato. In conseguenza alla variazione del guadagno, si modifica anche il livello di riferimento sulla scala verticale dello schermo, in modo tale che se il segnale viene attenuato, il livello di riferimento lo indichi e il segnale risulti scalato automaticamente.

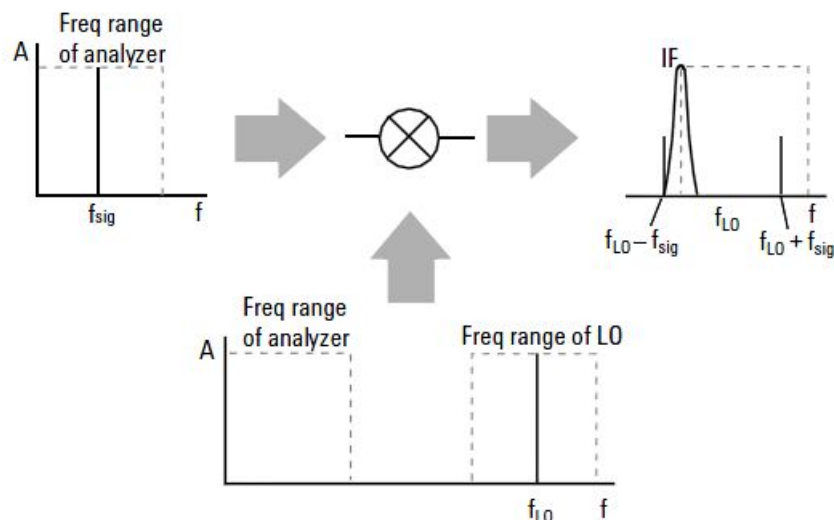


Figura 2.4: Il LO deve essere sintonizzato su  $f_{IF} + f_{sig}$ .

## 2.4 Filtri analogici

### 2.4.1 Risoluzione

La risoluzione in frequenza è l'abilità dell'analizzatore spettrale di separare due ingressi sinusoidali in due differenti componenti in uscita.

Principalmente questa dipende dalla banda passante dell'IF filter: se essa è abbastanza stretta due segnali ingresso, anche vicini, saranno catturati dalla banda del filtro in diversi momenti, ciò è sufficiente a garantire la distinzione delle componenti.

### 2.4.2 Sweep time

Se la risoluzione fosse l'unico criterio in base al quale stabilire la bontà di un analizzatore di spettro, basterebbe fare in modo che la banda dell'IF fosse la più stretta possibile.

La risoluzione però influenza un altro importante parametro, il *sweep time*, dal quale dipende direttamente il tempo necessario per effettuare la misura.

La dipendenza deriva dal fatto che, senza entrare nei dettagli, i circuiti costituenti l'IF hanno ovviamente una banda limitata, la quale è legata inversamente al tempo di scarica/carica dei componenti utilizzati. In pratica se il segnale che passa attraverso il filtro varia troppo velocemente, la risposta non riuscirà a seguire perfettamente il segnale d'ingresso. Sarà dunque necessario che il segnale resti all'interno della banda passante dell'IF per un tempo sufficiente ( *Time in passband* ) da considerare esauriti i transistori. Risulta valida la seguente relazione:

$$\frac{\text{Time in passband}}{\text{Sweep time}} = \frac{\text{RBW}}{\text{Span}} \quad (2.9)$$

dove Sweep time ( ST ) è il tempo necessario per effettuare la misura completa, RBW = Resolution Bandwidth è il minimo intervallo di frequenze analizzato contemporaneamente, Span è già stato definito ed è l'intervallo di frequenze che l'utente ha scelto di analizzare.

Poichè i tempi di variazione sono legati alla banda passante, si può affermare che:

$$\text{Rise time} = \frac{k}{RBW} \quad (2.10)$$

dove  $k$  è una costante di proporzionalità. *Time in passband* dovrà pertanto essere almeno pari al *rise time*; dalla prop. 2.9:

$$\frac{k}{RBW} = \frac{RBW \cdot ST}{Span} \quad (2.11)$$

che risolta rispetto al Sweep time dà:

$$ST = \frac{k \cdot Span}{RBW^2} \quad (2.12)$$

È evidente dall'eq. 2.12 come un cambiamento in risoluzione abbia una drammatica ripercussione sul tempo impiegato per compiere la misura. La fig. 2.5 riporta il grafico di una generica funzione  $y = \frac{1}{x^2}$ : si nota come il tempo di misura aumenti in maniera estremamente rapida con il diminuire della risoluzione.

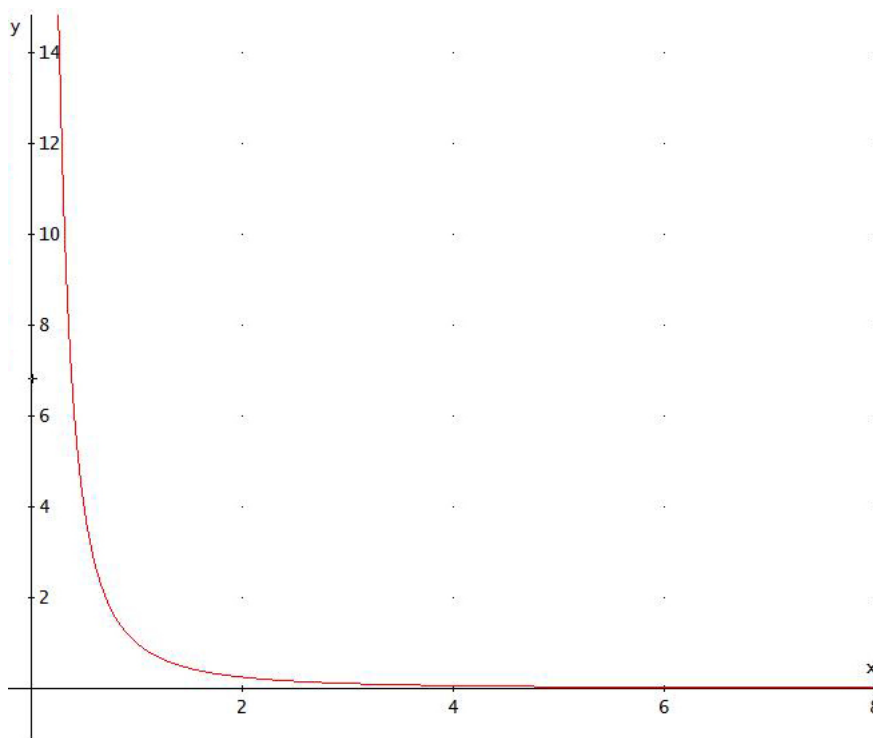


Figura 2.5: Andamento funzione  $y = \frac{1}{x^2}$ .



## Capitolo 3

# Programma LabVIEW

### 3.1 Introduzione a LabVIEW

LabVIEW ( acronimo di *Laboratory Virtual Instrument Engineering Workbench* ) è un ambiente per lo sviluppo di programmi che, a differenza degli altri sistemi testuali, è basata su un linguaggio di programmazione grafico chiamato *G language*. Il programma vero e proprio ha infatti la forma di un diagramma a blocchi.

Ogni operazione, descritta nei classici linguaggi di programmazione attraverso puro testo, viene eseguita in LabVIEW selezionando un opportuno “blocchetto” che raffigura un’immagine descrittiva dell’operazione compiuta. Dati, costanti, variabili e così via sono collegati al blocchetto ( e in generale all’interno del programma ) attraverso dei fili che legano dati a operazioni e determinano l’ordine di esecuzione del programma.

LabVIEW è inoltre un linguaggio di programmazione intrinsecamente *multi-tasking*, pertanto se si vuole che alcune operazioni siano tassativamente eseguite prima o dopo di altre si dovrà utilizzare un opportuno costrutto temporale.

Nella libreria del programma sono presenti un gran numero di funzioni per la maggior parte delle operazioni effettuate durante la programmazione. Le librerie inoltre contengono driver di dispositivi e permettono l’integrazione con il sistema operativo e con diverse interfacce di comunicazione.

Vi sono inoltre strumenti per salvare e visualizzare dati oltre che le normali funzioni degli ambienti di programmazione: breakpoints ed esecuzione passo-passo per il debugging. È anche possibile eseguire il programma in modo animato per vedere come i dati passano attraverso il flusso di esecuzione.

I programmi in LabVIEW sono chiamati *VI* ( *Virtual Instruments* ) perché il loro aspetto e il modo in cui funzionano imita quello di uno strumento reale. Di seguito si riportano le principali caratteristiche di un VI:

- I VI contengono un’interfaccia utente interattiva, chiamata *front panel* perché simula il pannello frontale di uno strumento reale. Questa parte contiene bottoni, interruttori, grafici, e altri controlli e indicatori.
- Le istruzioni che comandano il VI vengono dal *block diagram* che viene costruito in G. Il diagramma a blocchi è in pratica il codice del programma.
- Un VI può essere utilizzato a diversi livelli: se un VI è inserito all’interno di un altro VI, esso viene chiamato *subVI* e per esso dovranno essere definiti connettori di input/output, necessari per interfacciarlo col programma a più alto livello.

## 3.2 Il programma di controllo del sistema

### 3.2.1 Il front panel

Per descrivere il funzionamento del pannello frontale viene fatto riferimento alla fig. 3.1. Ogni sezione, indicata con un numero in figura sarà spiegata in dettaglio.

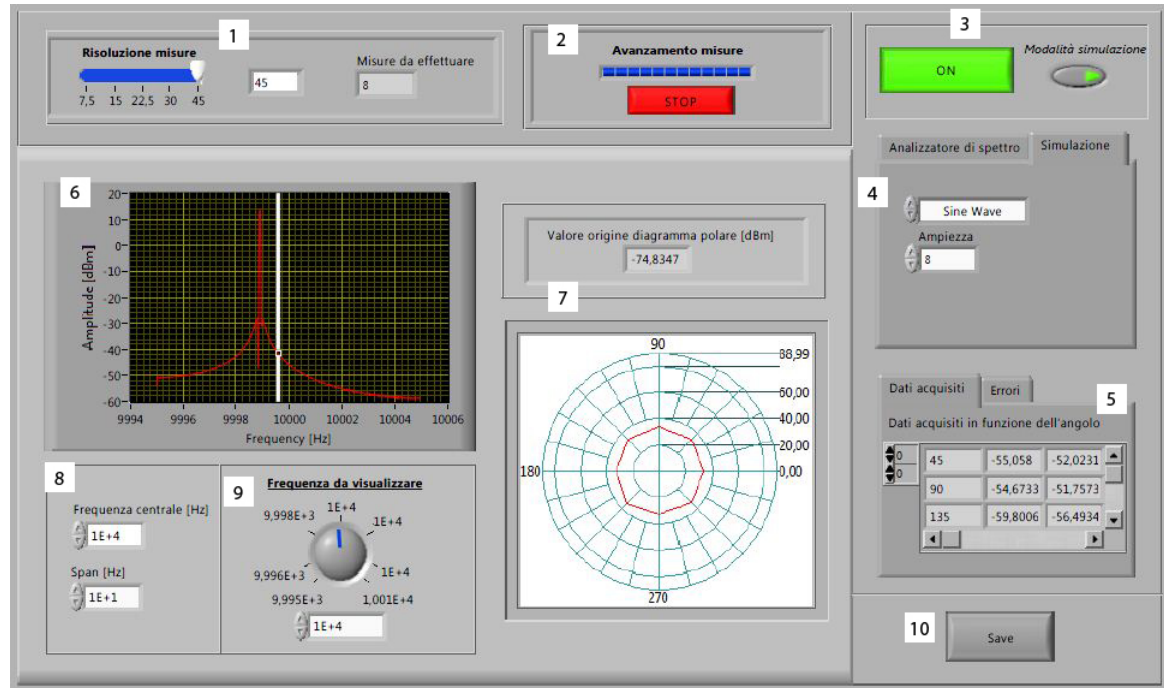


Figura 3.1: I numeri indicati nella varie parti del front panel sono i riferimenti utilizzati per descriverne il funzionamento.

1. La barra orizzontale con l'etichetta *risoluzione misure* ha l'obiettivo di permettere all'utente di scegliere ogni quanti gradi andare ad effettuare l'acquisizione spettrale. Questo parametro influenza direttamente lo spostamento dei motori: una minore risoluzione comporta meno misure e di conseguenza più *step* effettuati dal motore in modo consecutivo, senza alcuna misurazione da parte dell'analizzatore spettrale. Come si mostrerà in seguito, una minore risoluzione comporta un *diagramma polare* meno dettagliato. In particolare, i valori numerici riportati sotto la barra indicano l'angolo di rotazione effettuato dal motore. L'indicatore numerico avente etichetta *misure da effettuare* indica il numero di misure che si devono effettuare al fine di completare un'intera rotazione (  $360^\circ$  ) della piattaforma. Ad esempio scegliendo di effettuare le misure con la massima risoluzione possibile, si sceglie il primo valore sulla barra orizzontale, a cui corrisponde una misura ogni  $7,5^\circ$  e si ottiene:

$$\frac{360}{7,5} = 48 \quad (3.1)$$

ciò significa che verranno effettuate 48 misure per completare la rotazione. Nel diagramma polare si avranno dunque 48 punti, distanziati rispettivamente di  $7,5$  gradi.

2. La barra orizzontale con l'etichetta *Avanzamento misure*, indica appunto l'avanzamento temporale. Questo indicatore è utile per dare all'operatore la percezione di quante misure sono state effettuate e di quante ne mancano.

Il tempo  $T_{misura}$  impiegato dal VI per effettuare le misura dipende principalmente dal tempo di rotazione del motore e dal tempo  $T_{sweep}$  che l'analizzatore di spettro impiega per acquisire lo spettro. Possiamo pertanto scrivere:

$$T_{misura} = T_{elaborazione} + T_{rotazione} + T_{sweep} + T_{elaborazione} \quad (3.2)$$

dove  $T_{elaborazione}$  è il tempo legato all'inizializzazione del programma e alla gestione dei dati acquisiti. Considerando che la misura sarà ripetuta finché non viene completata un'intera rotazione della piattaforma, si avrà un tempo totale  $T_{tot}$ :

$$T_{tot} = T_{elaborazione} \cdot (T_{rotazione} + T_{sweep}) \cdot risoluzione\ misure \quad (3.3)$$

In caso di qualsiasi tipo di problema è stata implementata la possibilità di fermare l'acquisizione attraverso il pulsante *stop*.

La pressione del pulsante stop ferma il programma, ma non esce da esso. Verranno pertanto visualizzati nel diagramma polare, i punti che sono stati acquisiti, che possono inoltre essere salvati attraverso il pulsante *Save*. Un esempio è presentato

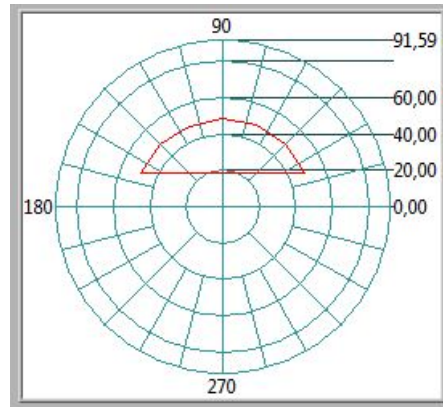


Figura 3.2: Diagramma polare incompleto: visualizzato premendo il pulsante stop prima del termine delle misure.

in fig. 3.2, nella quale la risoluzione scelta è di 22,5 gradi ed è stato premuto stop dopo la settima misura.

3. Il pulsante *ON* accende e spegne il VI. Il pulsante alla destra di *ON*, con l'etichetta *Modalità simulazione*, comanda appunto l'attivazione di tale modalità. Quando questa è attivata nessuno strumento reale è necessario per eseguire il programma; tale funzione è risultata molto utile soprattutto in fase di sviluppo del programma. Alcuni parametri che controllano l'onda simulata sono descritti nella parte relativa alla *sezione 4*, contenuti nel tab denominato appunto *simulazione*.
4. Il *tab control* utilizzato è stato diviso nelle sezioni "Analizzatore di spettro" e "Simulazione" perché queste modalità sono mutualmente esclusive: se siamo in modalità simulazione utilizzeremo il tab corrispondente senza preoccuparci delle

opzioni selezione nel tab *Analizzatore di spettro*, in quanto queste non influiranno in alcun modo il funzionamento del programma; vale lo stesso per il viceversa<sup>1</sup>.



Figura 3.3: Parametri e opzioni contenute nei due tab: a sinistra per il funzionamento reale a destra per quello simulato.

- **Analizzatore di spettro:** contiene i parametri utili per il funzionamento con gli strumenti reali. Il controllo sotto la voce *VISA resource name* permette di sezionare l'analizzatore di spettro collegato, tramite appunto l'interfaccia VISA<sup>2</sup>. La voce *Numero campioni* riporta il valore di punti acquisiti dall'analizzatore di spettro. Benchè questo sia un parametro non modificabile nello strumento ( pari a 401 *points* nel caso considerato, vedi dettaglio fig. 3.4 ) è stata implementata la possibilità di modificarlo per aumentare la versatilità del programma, anche ad alto livello.

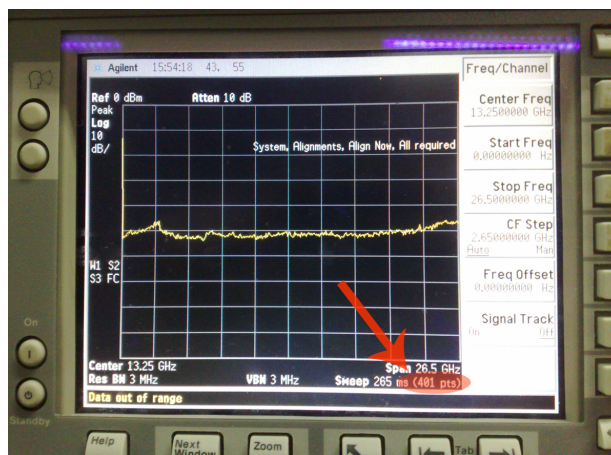


Figura 3.4: Schermo dell'analizzatore di spettro; dettaglio indicate il numero di punti acquisiti.

- **Simulazione:** in questa sezione è possibile scegliere il segnale per il quale verrà simulato lo spettro. È da notare che lo spettro visualizzato non sarà quello

<sup>1</sup>In realtà la modifica di *Numero campioni* modifica il numero di punti acquisiti in modalità simulazione. Questo apparente errore è in realtà pensato per simulare lo stesso numero di punti che normalmente utilizziamo con lo strumento. L'alternativa in fase di programmazione era inserire un valore costante ma la prima scelta è sembrata essere migliore.

<sup>2</sup>Si veda il lavoro di tesi [12].



di un'onda ideale in quanto, come si vedrà poi nel codice, si è utilizzato uno strumento che simula il segnale nel mondo reale ( con rumore quindi ). Il segnale selezionabile può essere: sinusoidale, a dente di sega, quadrata e triangolare. È possibile inoltre indicarne l'ampiezza nell'apposito controllo. La modalità simulazione è stata utilizzata solo per testare il programma, pertanto cambiando i valori dei segnali in modo casuale è possibile incorrere in errori, in particolare si deve sempre verificare che i parametri scelti rispettino i limiti imposti dal teorema del campionamento, ovvero:

$$F_{centrale} \leq \frac{F_s}{2} \quad (3.4)$$

Non è stata prevista la possibilità di modificare la frequenza di campionamento  $F_s$  o di altri parametri ad un più basso livello della simulazione; ciò avrebbe infatti complicato non poco il pannello frontale senza al contempo rappresentare una funzionalità utile per il progetto. Valori di default sono:

$$\begin{aligned} \text{Frequenza centrale} &= 10 \text{ kHz} \\ \text{Span} &= 100 \text{ Hz} \end{aligned}$$

con questi valori è possibile testare il programma senza problemi, essendo rispettato quanto previsto dal teorema del campionamento.

Per completezza si fa notare che il valore di default di  $F_s$  è 51,2 kHz, richiamando la condizione 3.4 si ha:

$$F_{centrale} \leq 25,6 \text{ [kHz]}$$

Pertanto la massima frequenza simulabile è 25,6 kHz.

Per quanto riguarda lo span non vi sono particolari vincoli: esso deve essere impostato in modo tale da garantire che la frequenza massima sia inferiore alla frequenza limite di 25,6 kHz. Si ricorda inoltre che un elevato span comporta una minore risoluzione in frequenza.

5. In questa sezione vengono riportati dati ed eventuali errori. La matrice conte-

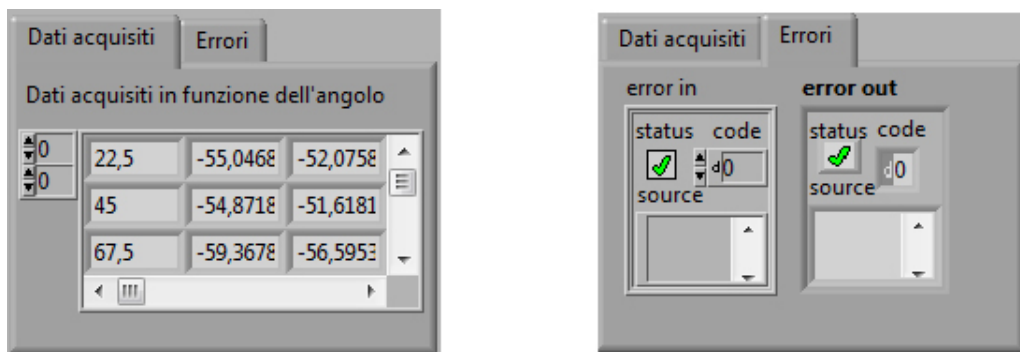


Figura 3.5: Dati acquisiti ed eventuali errori.

nente i dati è composta da una prima colonna che indica gli angoli raggiunti dal motore ( o simulati ) per i quali i dati sono stati acquisiti, in dBm. La riga a fianco all'angolo contiene un numero di ampiezze pari al numero di campioni acquisiti dallo strumento. Non è stata implementata la possibilità di scegliere il numero di dati acquisiti per la modalità in simulazione.

Le misure rappresentano le ampiezze delle componenti spettrali all'interno dello span analizzato. I valori delle relative frequenze sono distribuiti ad intervalli regolari all'interno dello span. Se ad esempio si hanno 401 punti acquisiti, allora la distanza in frequenza tra i campioni acquisiti  $df$  sarà:

$$df = \frac{span}{401} \quad (3.5)$$

6. Il grafico visualizza lo spettro del segnale ( in dBm ), in funzione della frequenza ( indicata in Hz ).

Lo scopo del virtual instrument realizzato tuttavia non è visualizzare l'intero spettro per un preciso angolo in cui la piattaforma si trova, pertanto quando la piattaforma si sposta e l'analizzatore di spettro effettua una nuova acquisizione, il grafico precedentemente visualizzato viene sovrascritto con le nuove misure effettuate e non è più possibile visualizzarlo. I dati della precedente acquisizione sono comunque ancora consultabili dalla tabella dati. In particolare il grafico in questa sezione visualizza una riga della matrice contenente i dati acquisiti ( si esclude la prima cella della riga, in quanto questa include l'angolo che è inutilizzato ).

La barra verticale presente nel grafico indica la frequenza per la quale è stato calcolato il diagramma polare. Spostando il valore di frequenza selezionato mediante tale barra, il diagramma polare verrà ricalcolato considerando i dati relativi alla frequenza scelta.

7. Il diagramma polare fornisce l'emissione in funzione dell'angolo di rotazione della piattaforma ( e del motore che la muove ). Nella fig. 3.6(a) è stata proposta

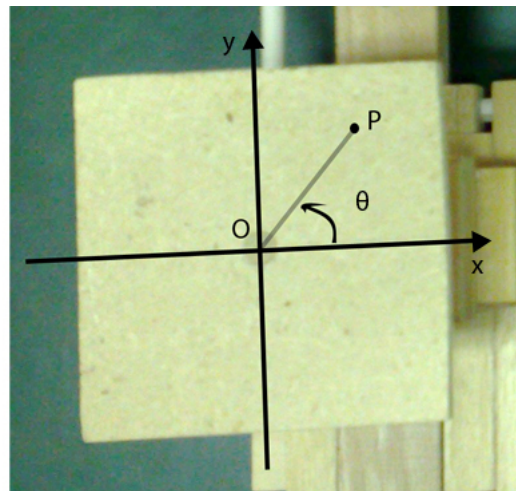
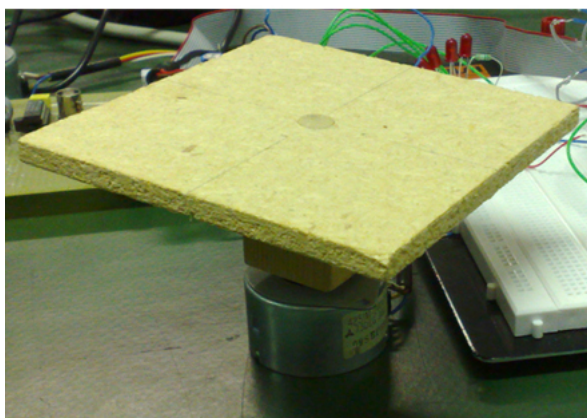


Figura 3.6: A sinistra (a) la piattaforma sopra al motore. A destra (b) la piattaforma sull'apparecchio vista dall'alto.

un'implementazione puramente dimostrativa della rotazione motore-piattaforma. Il motore come si vede in figura, si trova vicino all'eventuale sorgente, che dovrebbe essere posta idealmente sopra alla piattaforma, introducendo così disturbi e riflessioni che finirebbero per falsare la misura.

Per ovviare a questo problema si rende necessario l'utilizzo di un meccanismo più sofisticato, come quello riportato a titolo di esempio in fig. 3.7. In questo caso il dispositivo da testare è posizionato sopra la piattaforma che viene azionata a

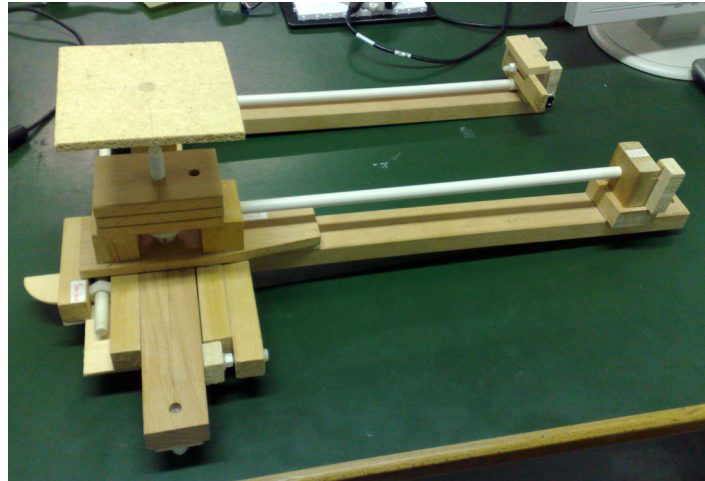


Figura 3.7: L'apparecchio per effettuare le misure di radiazione.

distanza attraverso l'ausilio di due motori passo-passo. Il moto viene trasferito dall'albero dei motori alla piattaforma attraverso l'ausilio di viti senza fine ed altri meccanismi per la trasmissione del moto in plastica. Come si nota, tutta la struttura d'appoggio è stata realizzata in legno e plastica per evitare l'insorgenza di onde riflesse.

In fig. 3.6 (b) sono indicati i riferimenti per la posizione angolare. Sul piano della piattaforma sono state tracciate, per maggiore chiarezza, due rette passanti per il centro dell'asse di rotazione. Scegliendo queste rette ortogonali tra loro, una orizzontale e l'altra verticale, si ottengono dei riferimenti analoghi a quelli in figura. L'angolo  $\theta$  in particolare sarà quello che viene a formarsi tra il semiasse positivo delle  $x$  e il segmento  $\overline{OP}$  indicante la rotazione del rotore.

In particolare si considera che il punto P, prima di iniziare la rotazione del motore, si trovi sull'asse positivo delle  $x$ . In questo modo l'angolo  $\theta$  vale inizialmente zero. I riferimenti  $x$ ,  $y$  sono fissati e non ruotano, mentre il punto P si sposta con la piattaforma: l'angolo  $\theta$  indica pertanto la rotazione della piattaforma ed è questo il riferimento preso per gli angoli indicati nella tabella dati.

Il diagramma polare è un grafico dove le coordinate corrispondono a *modulo* e *fase*. In particolare esso visualizza la funzione:

$$f(r, \theta) = r \cdot e^{j\theta}$$

dove  $r$  in questo diagramma rappresenta l'ampiezza della componente spettrale acquisita e  $\theta$  è l'angolo indicante la rotazione.

L'indicatore grafico per la costruzione del diagramma polare disponibile in LabVIEW accetta come valori del modulo  $r$  solo valori positivi, non contemplando la possibilità di fornire valori negativi espressi con scala logaritmica. Tuttavia, i valori acquisiti dall'analizzatore di spettro sono espressi in dBm, quindi utilizzando una scala logaritmica che permette di avere valori di modulo anche negativi. Per poter utilizzare il diagramma polare disponibile in LabVIEW sarà necessario un'opportuna correzione.

Viene effettuato il controllo di tutti i dati acquisiti e, in particolare, viene calcolata

la minima ampiezza acquisita. Se tale valore risulta essere negativo ( in dBm ), vengono traslati tutte le componenti spettrali acquisite di tale valore, sommando il modulo del minimo calcolato. In questo modo l'origine corrisponde al minimo, infatti:

$$x + |x| = 0, \quad \text{se } x < 0 \quad (3.6)$$

Il valore dell'origine *reale* ( non traslato ) è indicato nella sezione appena sopra al diagramma polare.

Il grafico viene creato utilizzando i campioni nella *colonna* dei dati acquisiti, corrispondenti alla frequenza selezionata dall'utente: la colonna infatti contiene le ampiezze della emissione in funzione di tutti gli angoli raggiunti dal motore. Variando il valore della "frequenza da visualizzare" il grafico polare cambia, essendo in tal caso selezionata una diversa colonna nella matrice dei dati acquisiti.

8. La *frequenza centrale* rappresenta la frequenza alla quale è centrato lo *span*. In pratica, se introduciamo il concetto di  $f_{iniziale}$  e  $f_{finale}$  di acquisizione si può scrivere:

$$f_{iniziale} = f_{centrale} - \frac{span}{2} \quad (3.7)$$

$$f_{finale} = f_{centrale} + \frac{span}{2} \quad (3.8)$$

sottraendo l'eq. 3.7 all'eq. 3.8 ottengo:

$$f_{finale} - f_{iniziale} = span \quad (3.9)$$

I dati acquisiti vengono acquisiti partendo da  $f_{iniziale}$  fino a  $f_{finale}$  con un intervallo dato da:

$$df = \frac{span}{points} \quad (3.10)$$

dunque:

$$f_{acquisizione} = f_{iniziale} + k \cdot df, \quad k = 0, 1, \dots, points \quad (3.11)$$

quando  $k=points$ , dall'eq. 3.10, ho  $df \cdot points = Span$ . Dall'eq. 3.9 si ha dunque che  $f_{iniziale} + span = f_{finale}$ .

9. La *frequenza da visualizzare* influenza direttamente il diagramma polare. Agendo sulla manopola o inserendo un valore nel controllo sottostante, si va a spostare la barra verticale presente sul grafico rappresentante lo spettro del segnale acquisito in riferimento ad una determinata posizione angolare e, di conseguenza, a selezionare la colonna corrispondente della matrice contenente i dati acquisiti. La selezione di nuovi dati, in particolare di una nuova colonna, comporta automaticamente l'aggiornamento del diagramma polare.
10. Il pulsante *Save* permette il salvataggio del diagramma polare e dei dati visualizzati. Tale funzione genera un file con estensione html contenente:
- il titolo, che indica la frequenza per la quale i dati sono stati estratti;
  - il diagramma polare;
  - una tabella contenente gli stessi valori presentati graficamente nel diagramma.

Dopo la pressione del tasto viene chiesto inizialmente dove salvare l'immagine del grafico e successivamente dove salvare il file.html.

Un esempio di report verrà mostrato nel capitolo 4, dove verranno descritte con maggior dettaglio tutte le fasi di esecuzione del programma.

### 3.2.2 Schema a blocchi

In questa sezione viene illustrato il codice sorgente del virtual instrument creato. Il programma principale può essere diviso concettualmente in 3 macro-sezioni ( si veda fig. 3.8 ):

1.   • Simulazione
  - Analizzatore di spettro e motori
2. Elaborazione dati e diagramma polare
3. Salvataggio dati

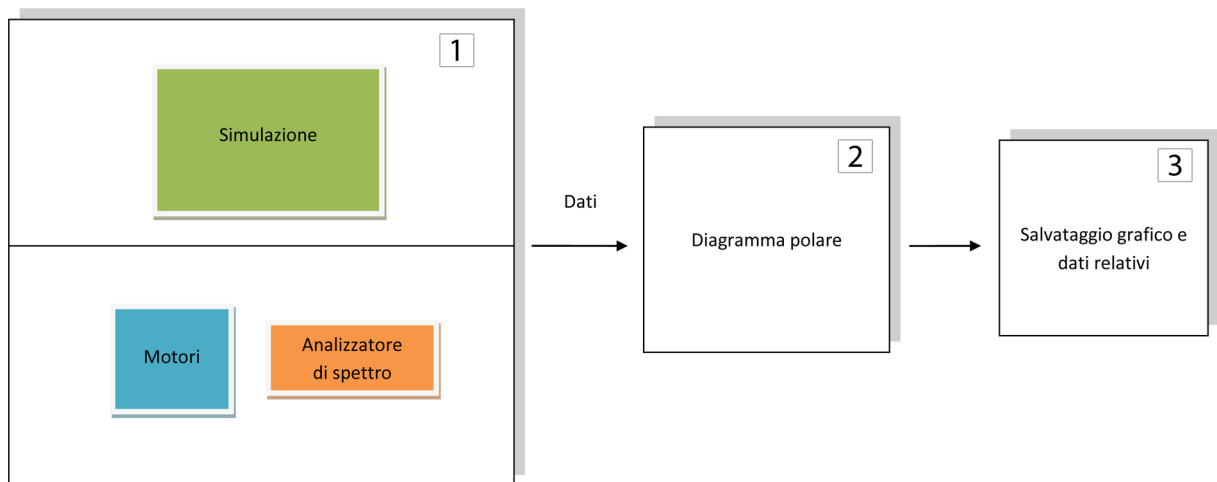


Figura 3.8: Macro-aree di funzionamento del programma.

Per descrivere il programma verranno utilizzate le aree indicate in fig. 3.8, per ciascuna delle quali verrà mostrato il codice descrivendone il funzionamento.

Prima dell'analisi delle sottosezioni, viene di seguito descritto il programma utilizzando un pseudo-codice, in modo da presentarne sia un riassunto sia un'introduzione alle strutture usate.

```

IF il programma è acceso THEN
  IF non siamo in modalità simulazione THEN
    inizializza l'analizzatore di spettro;
    FOR i FROM 0 TO misure da effettuare
      IF viene premuto stop OR si spegne il VI THEN
        interrompi rotazione;
        chiudi collegamento con lo strumento;
        BREAK;
      ELSE
        IF rotazioni motori è abilitata THEN
          ruota il motore;
          acquisisci spettro;
          visualizza grafico acquisito;
          salva dati e angolo di rotazione corrispondente;
        END IF;
      END IF;
    END FOR;
  
```

```

ELSE          ( modalità simulazione )
  FOR i FROM 0 TO misure da effettuare
    IF viene premuto stop OR si spegne il VI THEN
      BREAK;
    ELSE
      simula spettro e rotazione;
      salva dati e angolo di rotazione corrispondente;
    END IF;
  END FOR;
END IF;
WHILE il programma è acceso
  per la frequenza selezionata mostra diagramma polare;
  IF viene premuto save THEN
    genera report;
  END IF;
END WHILE;
END IF;

```

### 3.2.2.1 Controlli e inizializzazioni

Il programma inizialmente considera i valori inseriti di frequenza centrale, span e numero campioni e con essi calcola rispettivamente: frequenza iniziale, frequenza finale e df ( sono valide le eq. 3.7, 3.8 e quanto in frequenza 3.10, vedi fig. 3.9 ). Vengono inoltre calcolati,

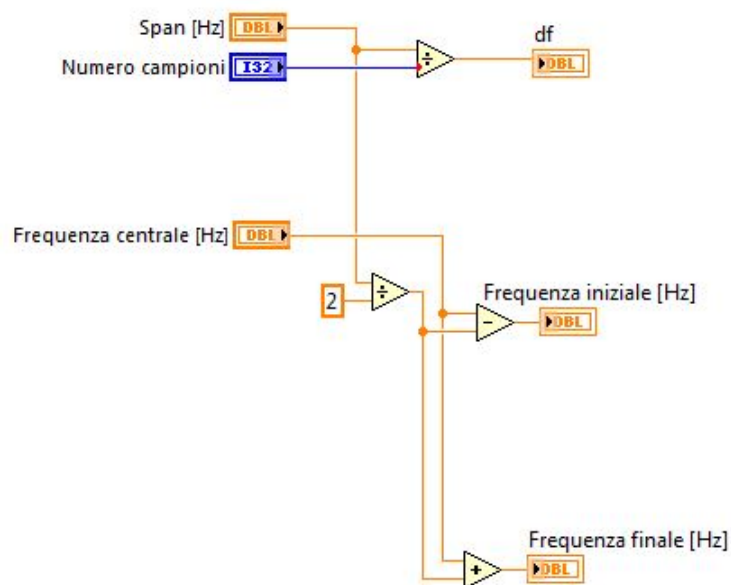


Figura 3.9: Calcoli iniziali per ricavare parametri e valori utili durante l'esecuzione.

in funzione della risoluzione scelta per le misure, il numero di misure da effettuare e il numero di passi che il motore deve eseguire prima di fare la misura.

Il motore utilizzato ha una risoluzione minima di 7,5 gradi, come si vede dal dettaglio in fig. 1.4. Questo significa che ogni impulso fornito fa fare un passo al motore, tale passo comporta una rotazione di 7,5 gradi.

Se ad esempio è stata scelta una risoluzione di 45 gradi il motore dovrà fare:  $45/7,5 = 6$  passi, prima di effettuare una misura. Inoltre, essendo  $360/45 = 8$ , il numero di misure

che verranno eseguite con l'analizzatore di spettro saranno complessivamente pari a 8 misurazioni, ogni misura sarà distanziata da quella adiacente di 45 gradi; infine mentre il motore ruota di un angolo pari alla risoluzione non verranno effettuate misure intermedie. Riassumendo:

1. Scelgo la *risoluzione delle misure*. Supponiamo che questa sia di 45 gradi.
2. Calcolo il numero di passi che il motore dovrà compiere prima di portarsi alla posizione intermedia. Nel caso preso ad esempio sarà:

$$\frac{45}{7,5} = 6 \quad (3.12)$$

3. Calcolo il numero di misurazioni da effettuare, che corrisponde al rapporto tra angolo giro (corrispondente ad una rotazione intera) e risoluzione angolare. Nel caso considerato si ottiene:

$$\frac{360}{45} = 8 \quad (3.13)$$

4. Muovo il motore del numero di passi necessario per raggiungere la posizione angolare di misura (nell'esempio il numero di passi è 6).
5. Viene misurato lo spettro nell'intervallo di frequenze considerato tramite l'analizzatore di spettro.
6. Si ripetono in successione i punti 4) e 5) per un numero di volte pari al numero di misurazioni da effettuare (ovvero 8).

Il calcolo dei parametri relativi a numero di passi del motore per posizione angolare e numero totale di misurazioni, è effettuato nella parte di programma mostrata in fig. 3.10.

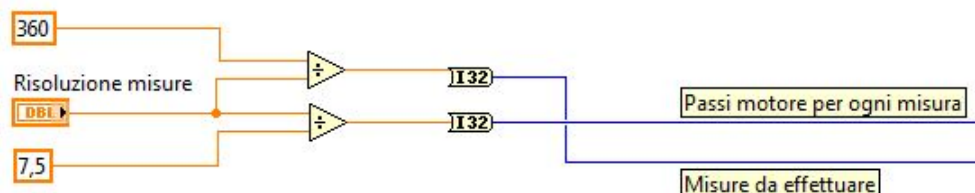


Figura 3.10: Calcolo dei valori che caratterizzano la risoluzione delle misure.

### 3.2.2.2 Il ciclo while principale

Il ciclo while che gestisce la rotazione e l'acquisizione dei dati è una delle strutture principali del programma e viene utilizzato per ripetere l'acquisizione dello spettro del segnale finché l'angolo giro non è stato completato.

Il ciclo, si veda fig. 3.11, contiene nidificata una *stacked sequence structure*, ovvero una struttura che stabilisce un ordine temporale di esecuzione per i frame che la costituiscono. La struttura utilizzata contiene **due frame**:

1. Nel primo viene controllato se l'esecuzione del ciclo deve essere terminata. Il ciclo termina se: viene premuto il tasto stop *oppure* viene raggiunto il termine del ciclo *oppure* se viene spento lo strumento. Quando una di queste condizioni si verifica è necessario terminare la comunicazione con lo strumento e fermare la rotazione dei motori.
2. Nel secondo frame viene eseguito il programma vero e proprio.  
Questa struttura contiene un costrutto di tipo *case* che esegue la simulazione o la modalità normale con motori e analizzatore di spettro.



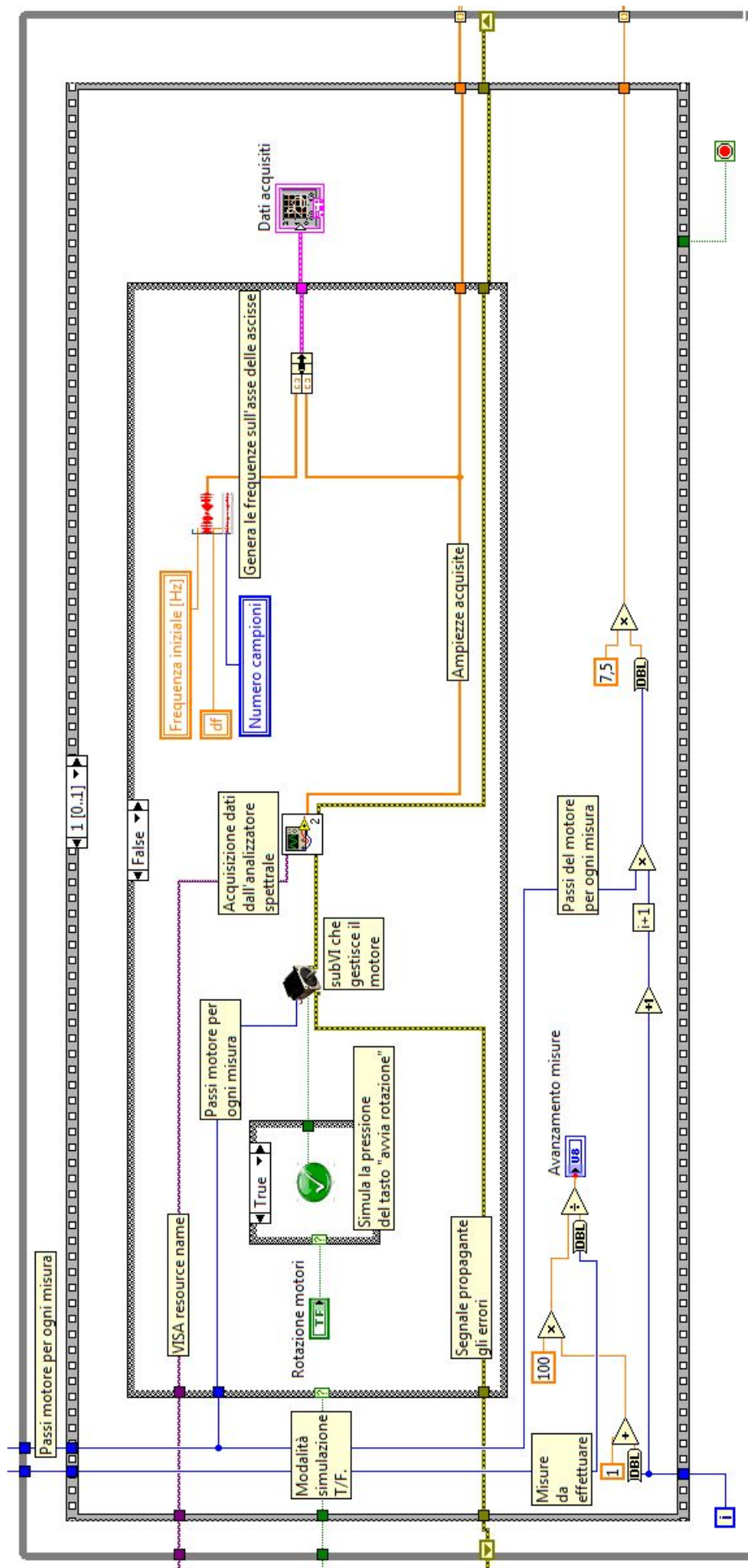


Figura 3.11: Ciclo while principale.

Il ciclo, se non è interrotto, viene eseguito un numero di volte pari a:

$$\frac{360}{\text{Risoluzione misure}} \quad (3.14)$$

L'avanzamento della misura è monitorato attraverso la barra orizzontale nel front panel, già analizzata nella sezione corrispondente. Tale barra viene riempita in base ad un valore numerico che le viene dato in ingresso: 0=la barra è vuota, 100=barra completamente piena. Pertanto per fare in modo che la barra tenga conto dell'avanzamento delle misure basterà dare in ingresso un valore  $x$  tale che:

$$(i + 1) : \text{Misure da effettuare} = x : 100 \quad (3.15)$$

dove  $i$  è l'indice corrispondente al numero di iterazioni compiute dal ciclo while principale. In particolare si è aggiunto l'1 perché l'indice  $i$  va da 0 al *numero di iterazioni totali* - 1, pertanto aggiungendo 1 si è fatto in modo che durante l'ultima iterazione l'indice  $i$  valga "misure da effettuare" e dunque l'avanzamento temporale indicato dalla barra risulti completo, si veda fig. 3.12.

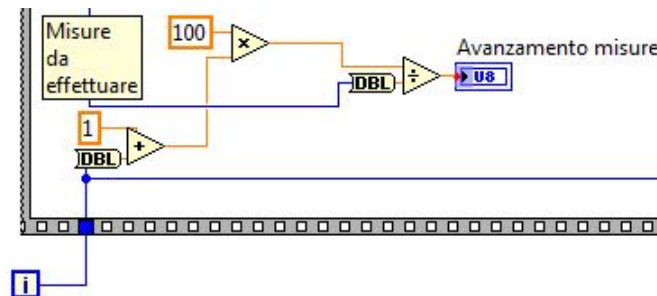


Figura 3.12: Controllo della barra indicante l'avanzamento delle misure.

### 3.2.2.2.1 Primo frame

Nel primo frame vengono eseguiti i controlli di terminazione del ciclo. Per la descrizione del frame viene fatto riferimento al codice a blocchi riportato in fig. 3.13.

La struttura più esterna in fig. 3.13 è il *ciclo while*.

L'indice  $i$ , ovvero il numero di iterazioni compiute (nell'angolo in basso a sinistra), viene confrontato col numero di misure da effettuare (meno 1). Tale confronto restituisce *true* quando  $i = \text{misure da effettuare} - 1$ ; questa verifica condiziona il limite massimo di volte che il ciclo può essere ripetuto.

Altre condizioni di terminazione si verificano quando viene premuto il pulsante stop o è stato spento lo strumento con l'interruttore *ON-OFF*.

Se una qualsiasi di queste condizioni si verifica, il ciclo termina e viene attivata la struttura case interna al frame. In questa struttura, se siamo in modalità normale (in caso di modalità di simulazione non verrà eseguita alcuna operazione), verrà chiuso il collegamento con l'ASA e fermata la possibilità di far ruotare i motori. In questo modo quando l'esecuzione passerà al secondo frame il programma non potrà né far ruotare i motori né acquisire i dati dall'analizzatore, e il ciclo terminerà.

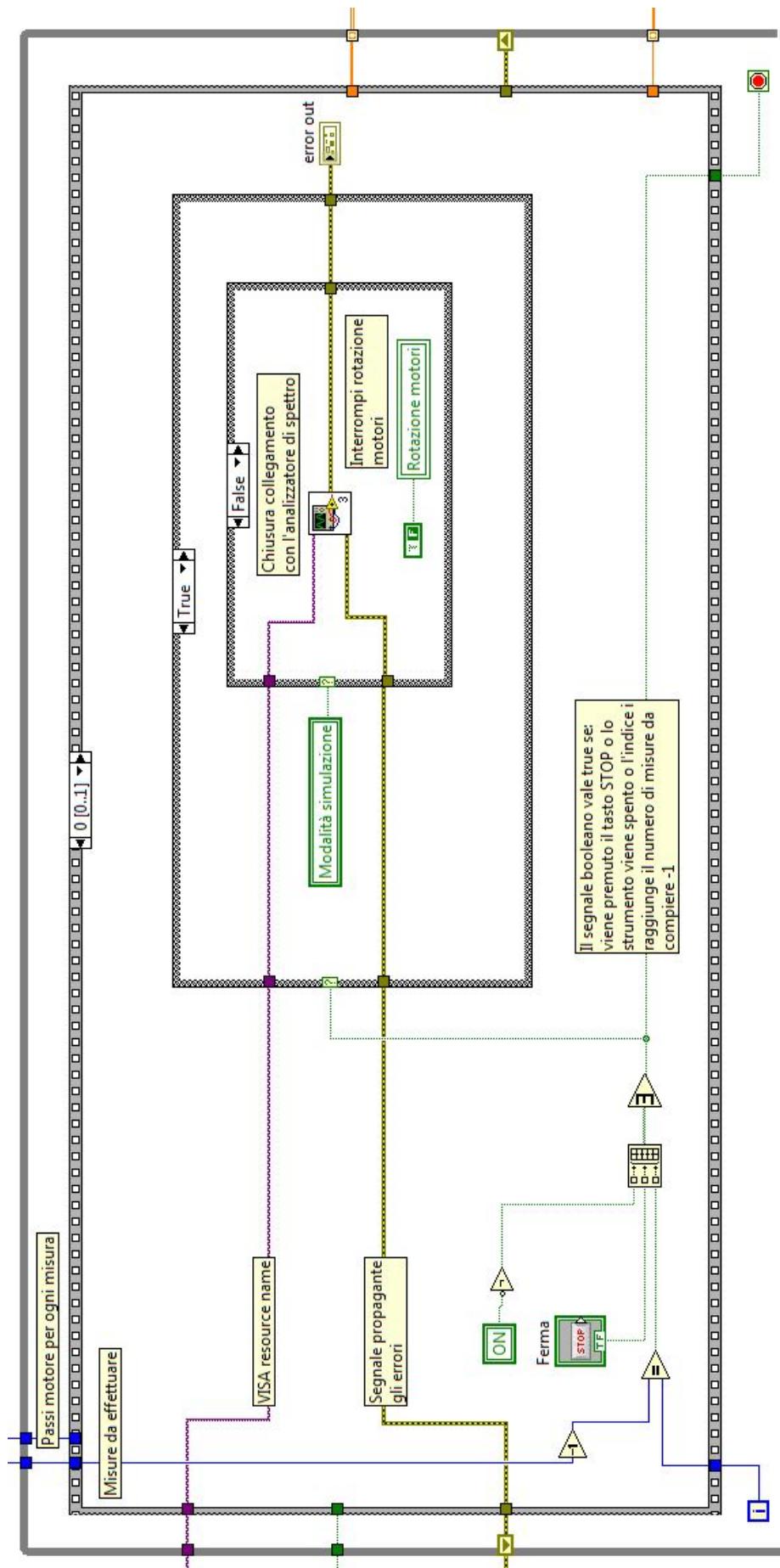


Figura 3.13: Primo frame interno al ciclo while: verifica se il ciclo deve essere terminato.

### 3.2.2.2.2 Secondo frame

Nella sezione riguardante il secondo frame ( interno al ciclo while principale ) verrà mostrata:

1. la modalità simulazione;
2. la modalità con motore e analizzatore di spettro.

#### 1) Modalità simulazione

La modalità simulazione si occupa di simulare l'acquisizione delle radiazioni emesse da una sorgente.

I dati e lo spettro visualizzato non hanno alcuna rilevanza: il loro unico scopo è quello di testare le funzionalità del VI senza avere gli strumenti, necessari per il normale funzionamento, a disposizione.

Per completezza viene implementata nel Front Panel la possibilità di modificare ampiezza e tipo di segnale, come già mostrato.

Il subVI<sup>3</sup> più importante in questa sezione è chiamato *SpectrumSim* ed è utilizzato per generare lo spettro, come si legge dalla descrizione in fig. 3.14. Gli ingressi sono 2 cluster che hanno rispettivamente lo scopo di specificare il segnale da generare e il numero di campioni complessivo. Queste applicazioni secondarie sono utilizzate nel programma

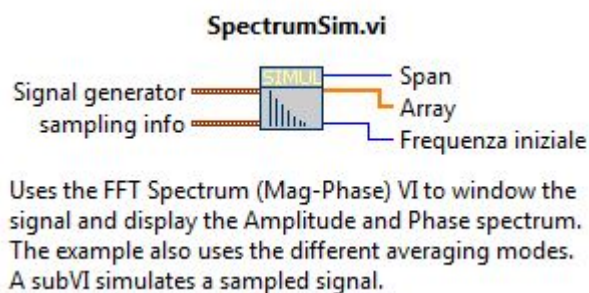


Figura 3.14: La descrizione del subVI *SpectrumSim* data dal *Context Help* di LabVIEW.

per effettuare la simulazione.

<sup>3</sup>Si veda la sezione: Introduzione a LabVIEW

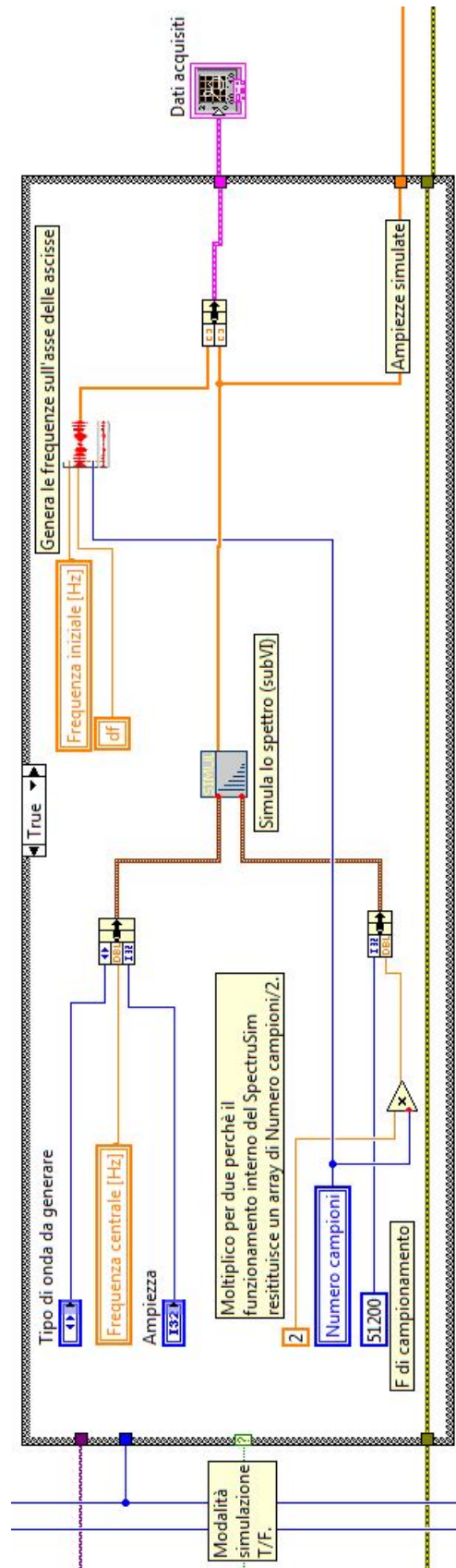


Figura 3.15: Sezione del programma che gestisce la simulazione.

In fig. 3.15 si nota la presenza di un altro subVI: questo genera le frequenze corrispondenti alle ampiezze acquisite; tali frequenze costituiranno l'asse delle ascisse nel grafico "Dati acquisiti". In particolare dati gli ingressi frequenza iniziale, numero campioni e  $df$ , esso genera un array nel quale ogni cella avrà la frequenza:

$$f_k = f_{iniziale} + k \cdot df, \quad k = 0, 1, \dots, \text{numero campioni}$$

Essere in modalità simulazione comporta ovviamente il non dover controllare il collegamento degli strumenti; pertanto, come si vedrà in seguito, vi saranno opportuni costrutti di tipo *case* adatti a verificare la modalità di funzionamento e, in base ad essa, interfacciare e inizializzare gli strumenti o meno.

## 2) Modalità con motore e analizzatore di spettro

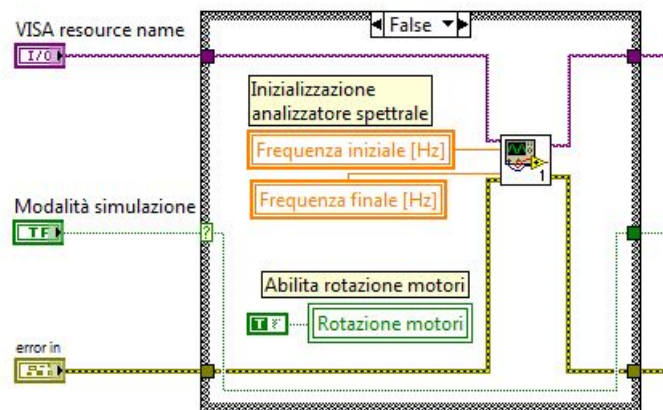


Figura 3.16: Inizializzazione dell'ASA e del motore.

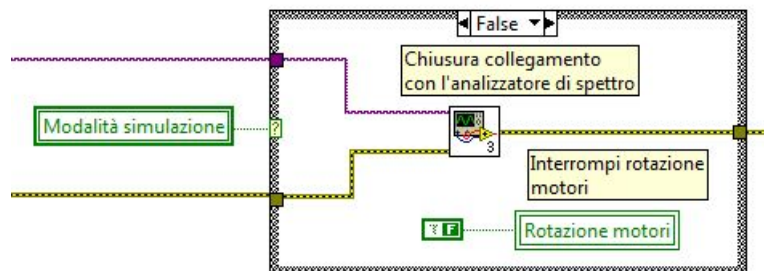


Figura 3.17: Chiusura collegamento con l'ASA e disabilitazione della rotazione del motore.

Questa sezione ha lo scopo di gestire e sincronizzare la rotazione dei motori e l'acquisizione dello spettro da parte dell'analizzatore di spettro.

Per la descrizione del programma che muove i motori si rimanda al lavoro di tesi [6]. In questa sede il subVI del motore passo-passo sarà trattato come una *black box*.



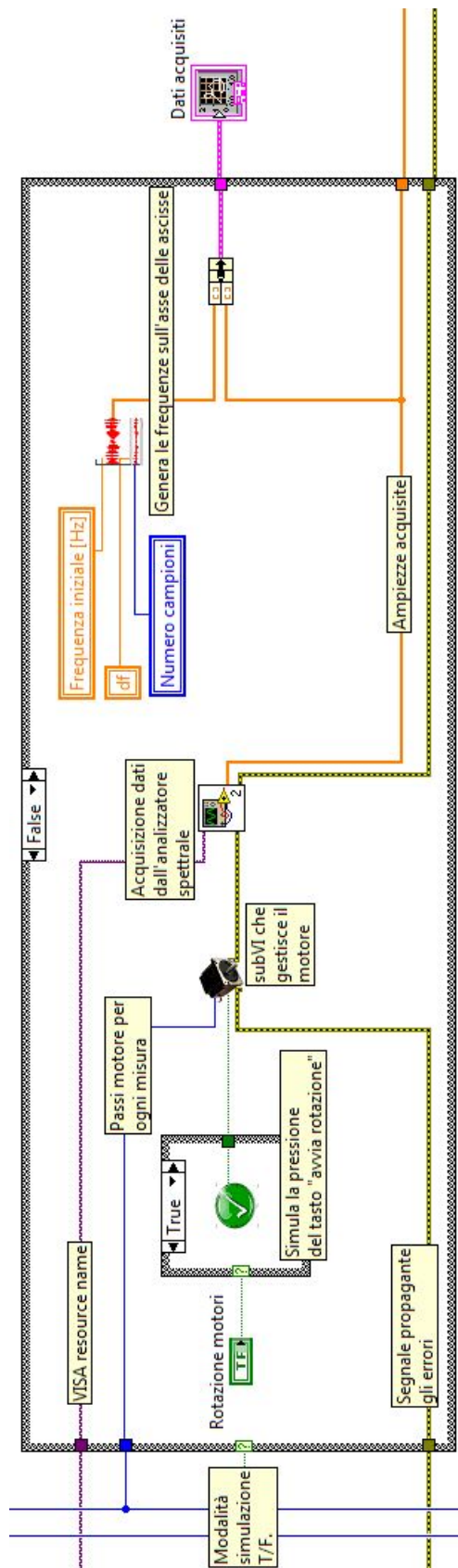


Figura 3.18: Sezione del programma che gestisce e sincronizza rotazione motori e acquisizione dati.

Il subVI per la gestione del **motore** riceve in ingresso un segnale booleano che avvia la rotazione (simula la pressione di un tasto, si veda il funzionamento interno del programma che gestisce il motore), ed un segnale numerico indicante il numero di passi da effettuare. Eventuali errori verificatisi prima del subVI o durante la rotazione stessa vengono propagati all'interno dal blocco.

La rotazione del motore ha inizio quando il segnale booleano in ingresso esegue una transizione del tipo  $false \rightarrow true \rightarrow false$ , simulando in questo modo la pressione di un tasto di avvio.

Il controllo *rotazione motori* è attivo durante l'esecuzione, in quanto questo viene sempre portato a *true* durante l'inizializzazione del programma<sup>4</sup>. Pertanto essendo il segnale *rotazione motori* true, si entrerà nella struttura che simula la pressione di un tasto (si veda fig. 3.18); tale simulazione sarà gestita dal subVI indicato in fig. 3.19. Questo comporterà l'avvio della rotazione del motore.

Si è scelto di implementare il controllo della rotazione dei motori (pur non rendendo

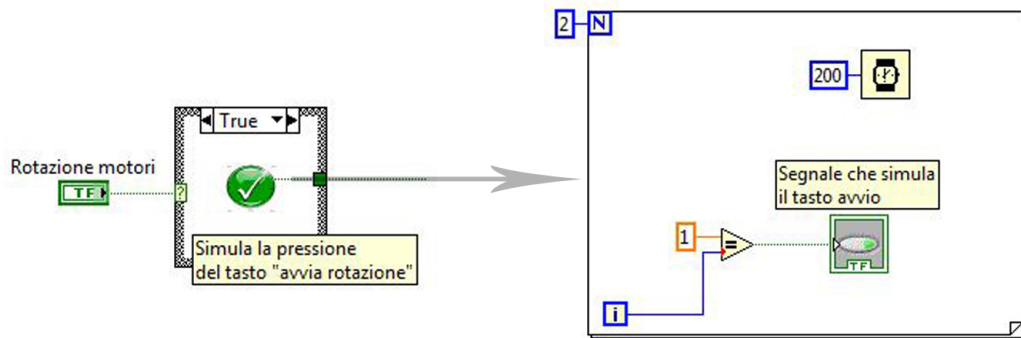


Figura 3.19: Simulazione di un un tasto:  $FALSE \rightarrow TRUE \rightarrow FALSE$ .

disponibile la modifica manuale di tale opzione all'utente) in quanto, quando il ciclo while principale viene interrotto, oltre alla chiusura del collegamento con l'ASA, viene portato a false anche il controllo *rotazione motori*. Quando viene eseguito il programma, se siamo in modalità normale, la rotazione verrà nuovamente abilitata (si veda fig. 3.16). Tale scelta, benchè non obbligata, è sembrata essere la migliore, infatti predispone il programma ad eventuali integrazioni future, come ad esempio la scelta di acquisire lo spettro in funzione di altri parametri, indipendenti dalla rotazione del motore.

Appena il motore ha effettuato il numero di *step* richiesti, il programma prosegue verso il subVI che gestisce l'acquisizione dall'analizzatore di spettro.

Il programma che gestisce la lettura dei dati dall'**analizzatore di spettro** è a sua volta diviso in 3 parti:

1. inizializzazione (si veda fig. 3.16);
2. acquisizione dati;
3. chiusura della comunicazione (si veda fig. 3.17).

<sup>4</sup>Per il corretto funzionamento del programma è necessario che i motori ruotino, viene quindi esclusa la possibilità di disabilitare la rotazione.



1) L'inizializzazione viene eseguita da un semplice programma che apre la connessione con lo strumento ( collegato al computer tramite un'interfaccia GPIB ). Inoltre, durante questa fase, vengono settati frequenza centrale e span. Il codice del subVI che si occupa di inizializzare l'analizzatore di spettro viene mostrato in fig. 3.20.

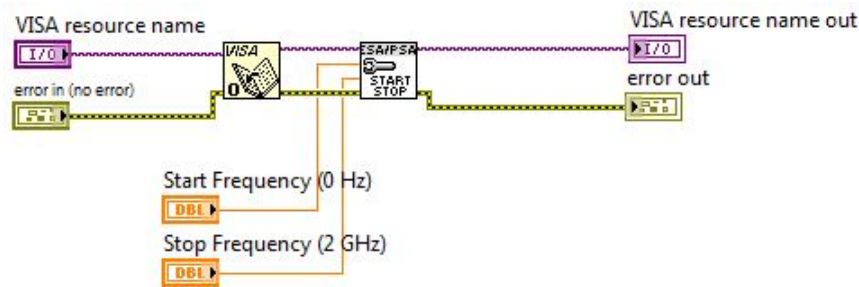


Figura 3.20: Programma che inizializza l'analizzatore di spettro.

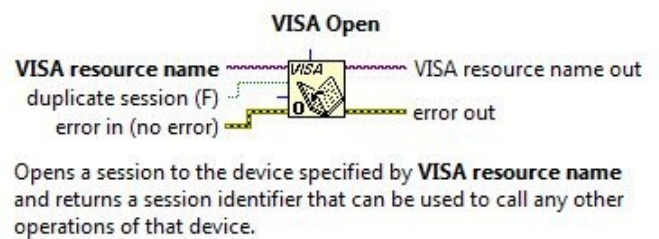


Figura 3.21: SubVI che apre il collegamento con l'ASA.

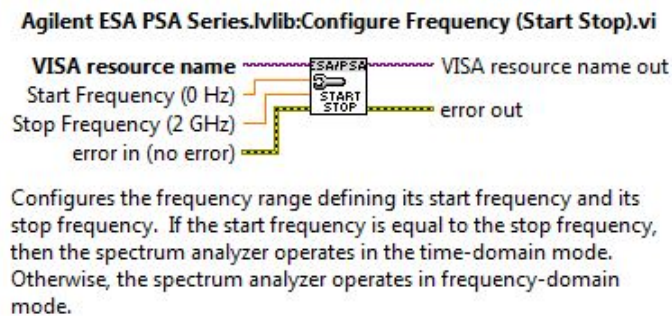


Figura 3.22: SubVI che setta frequenza centrale e span dell'ASA.

2) La lettura dei dati avviene tramite un semplice programma che si occupa di acquisire la traccia visualizzata dallo strumento. In particolare, l'array fornito in uscita dal subVI che si occupa della lettura dei dati contenuti nella memoria dell'analizzatore di spettro, contiene un numero di ampiezze pari ai *punti* indicati sullo schermo dello strumento stesso. I punti *devono* essere gli stessi impostati nell'apposita sezione del Front Panel; altrimenti, per come il programma è strutturato, la frequenza associata all'ampiezza non sarebbe corretta.

Il codice del subVI che gestisce l'acquisizione della traccia viene presentato in fig. 3.23.

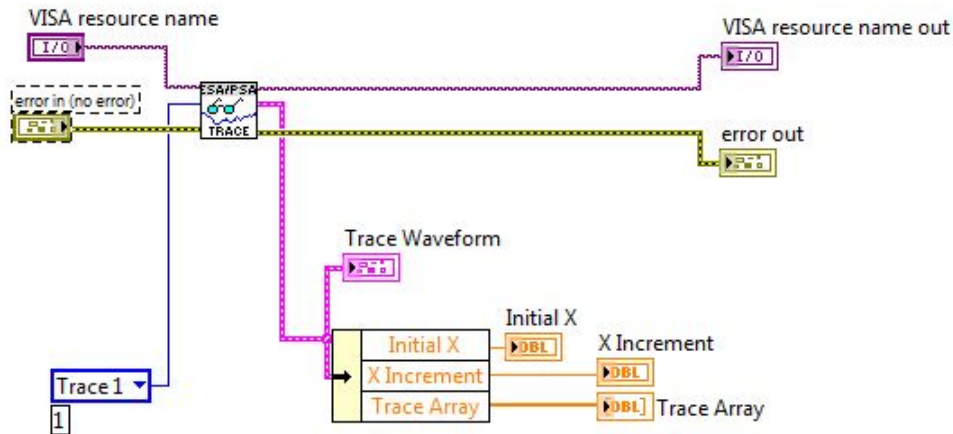


Figura 3.23: Programma che acquisisce lo spettro visualizzato dall'ASA.

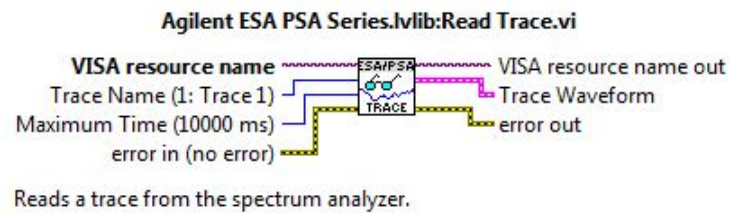


Figura 3.24: SubVI che legge la traccia acquisita dall'ASA.

3) La chiusura dello strumento avviene quando il ciclo *while* principale termina. In questo sotto-programma l'unica operazione eseguita è quella di chiudere la comunicazione con l'analizzatore. Il codice del subVI che chiude il collegamento con l'ASA viene presentato in fig. 3.25.

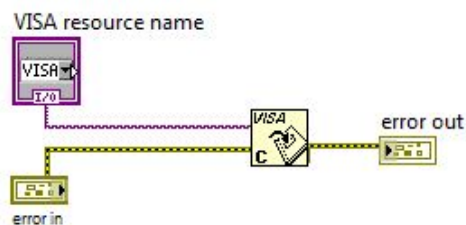


Figura 3.25: Programma che chiude il collegamento con lo strumento.

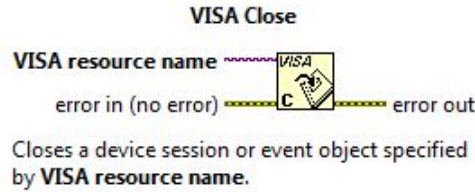


Figura 3.26: SubVI che chiude il collegamento con l'analizzatore di spettro.

### 3.2.2.3 Diagramma polare

La parte del programma principale riguardante la gestione del grafico polare è indicata nel funzionamento logico in fig. 3.8 e corrisponde alla seconda macro-area del programma. Essa, come si nota, riceve in ingresso i dati in uscita dalla parte iniziale e si occupa di processarli in modo tale da generare il diagramma.

Questa seconda macro-area riceve in ingresso i dati relativi alle radiazioni con i rispettivi angoli di acquisizione. In particolare i segnali che entrano in questa sezione sono due: uno contiene la matrice delle ampiezze acquisite, l'altro invece è composto da un array con gli angoli corrispondenti: questi si fondono in modo tale che la prima colonna corrisponda agli angoli mentre il resto della riga contenga le ampiezze rispettive ( si veda fig. 3.27 ). Il codice che si occupa di effettuare l'unione è mostrato in fig. 3.28, l'area evidenziata è il blocco di LabVIEW che unisce i due ingressi.

L'array avrà dunque un numero di colonne pari a  $points + 1$  ( ampiezze acquisite +

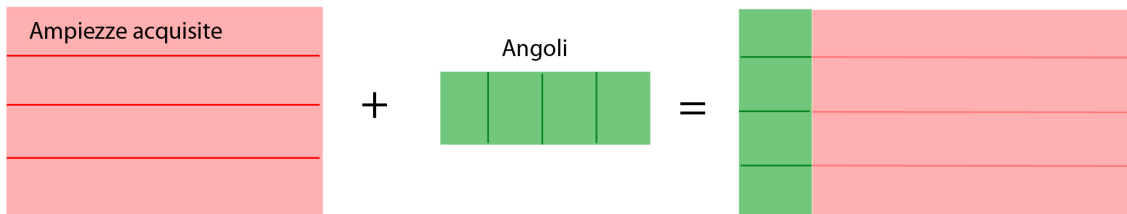


Figura 3.27: Visualizzazione grafica dell'unione dei segnali contenenti angoli e ampiezze acquisite.

angolo) e un numero di righe determinate dalla risoluzione scelta per le misure<sup>5</sup>. In particolare gli angoli contenuti nell'array e successivamente nella tabella dati vengono calcolati utilizzando due informazioni: l'indice  $i$  del ciclo while e il numero di passi intermedi tra le acquisizioni. L'angolo dell'acquisizione  $i$ -esima sarà dato dal prodotto tra  $i$  ( indice che tiene conto delle iterazioni effettuate nel ciclo while ) e il numero di passi intermedi.

$$\alpha_i = i \cdot \text{passi intermedi tra le acquisizioni} \quad (3.16)$$

dove  $\alpha_i$  è l'angolo corrispondente all' $i$ -esima iterazione. La frequenza da visualizzare viene modificata nel pannello frontale attraverso lo spostamento della barra verticale, indicante la frequenza selezionata sul grafico *dati acquisiti*, oppure tramite il controllo diretto, inserendo la frequenza voluta.

In fig. 3.29 viene presentato il blocco del subVI che genera il diagramma polare. Tale blocco è presente nel programma ad alto livello, avente:

<sup>5</sup>Si vedano le sezioni 5 e 8 del pannello frontale.

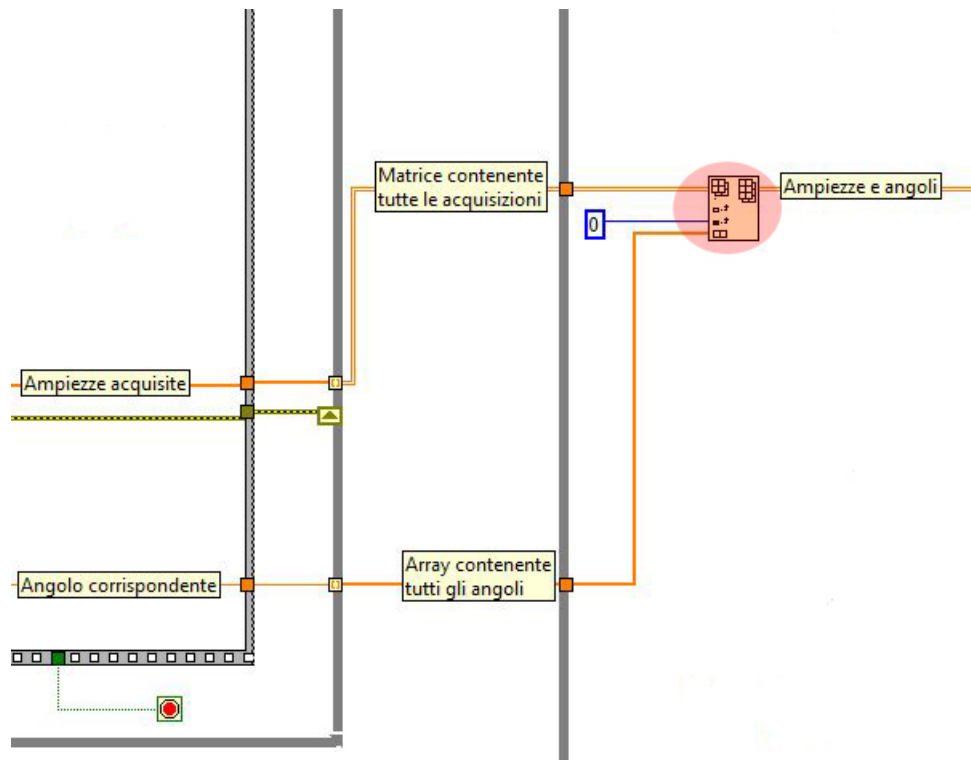


Figura 3.28: Diagramma a blocchi contenente il codice per unire i dati acquisiti con i rispettivi angoli.

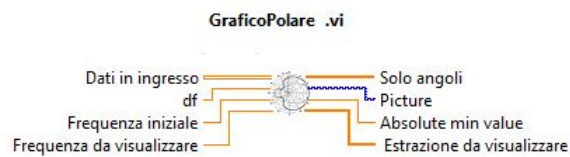


Figura 3.29: Icona del subVI che genera il diagramma polare.

- Ingressi:
  1. dati in ingresso ( ampiezze acquisite e rispettivi angoli );
  2. quanto in frequenza  $df$ ;
  3. frequenza iniziale;
  4. frequenza da visualizzare;
- Uscite:
  1. solo angoli;
  2. picture;
  3. absolute min value;
  4. estrazione da visualizzare.

L'uscita del subVI che viene visualizzata sul pannello frontale è *Picture*. *Absolute min value* invece, rappresenta il più basso valore presente nei dati acquisiti. Nel diagramma

polare, come abbiamo già mostrato, il minimo assoluto corrisponde al valore dell'origine del diagramma.

Le altre uscite: ( *solo angoli* ed *estrazione da visualizzare* ) vengono utilizzate per generare il *report* del programma. In particolare l'uscita *solo angoli* sarà la prima colonna della tabella generata dal report, mentre i *dati da visualizzare* saranno la seconda.

Il codice che si occupa di modificare i dati affinché diventino compatibili con il generatore di report è mostrato in fig. 3.30. Il ciclo for in particolare traduce i valori numerici in stringhe, raggruppandole in uscita in un array-bidimensionale: angoli e ampiezze.

L'utilizzo degli ingressi del blocco sarà visto nella sezione "il subVI del diagramma po-

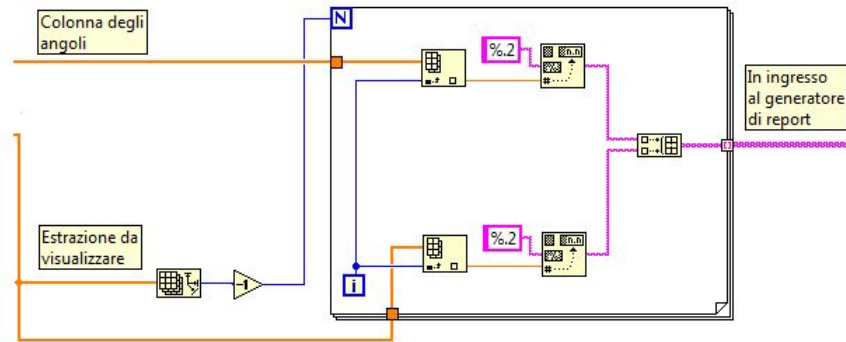


Figura 3.30: Il codice che converte i valori numerici visualizzati in un formato opportuno per il generatore di report.

lare". Nella quale viene mostrato il funzionamento interno del subVI che si occupa di convertire i dati in ingresso in modo da generare il diagramma polare.

### 3.2.3 Il subVI del diagramma polare

Il subVI del diagramma polare comprende principalmente una serie di blocchi per la manipolazione di strutture dati, in particolare degli array.

I segnali utilizzati in questo sotto programma sono stati riportati in fig. 3.31. Si descriverà qui di seguito come questi vengono utilizzati.

Il codice della **parte iniziale** del programma è mostrato in fig. 3.31. In questa sezione vengono effettuate principalmente due operazioni:

- viene calcolata la posizione dell'array che deve essere visualizzata nel diagramma;
- i dati in ingresso vengono separati in ampiezze acquisite e rispettivi angoli.

Per quanto riguarda la determinazione della posizione dell'array viene utilizzata l'eq. 3.11. Lo strumento acquisisce le ampiezze delle radiazioni per frequenze che stanno nel range specificato dallo span. Pertanto se si vorrà visualizzare il diagramma polare per una determinata frequenza si sfrutterà il fatto che le frequenze acquisite saranno tutte in tale intervallo e in particolare vi saranno un numero di ampiezze pari a *points*. Per comodità si riporta qui sotto l'eq. utilizzata.

$$f_{visualizzata} = f_{iniziale} + k \cdot df \quad (3.17)$$

$$k = \frac{f_{visualizzata} - f_{iniziale}}{df} \quad (3.18)$$

dove  $k$  è la posizione dell'array contenente la frequenza più vicina a quella chiesta. È stato utilizzato il concetto di “frequenza più vicina” perchè vi è un intervallo discreto  $df$  tra le varie acquisizioni, pertanto se si chiede di visualizzare una frequenza ad esempio tra  $f_{iniziale} + k \cdot df$  e  $f_{iniziale} + (k + 1) \cdot df$  ( $k < points$ ):

verrà selezionato l'indice  $k$ , se la frequenza è più vicina a  $f_{iniziale} + k \cdot df$ ; altrimenti verrà selezionato  $(k + 1)$ , se la frequenza è più vicina a  $f_{iniziale} + (k + 1) \cdot df$ .

La **seconda parte**, riportata nella parte inferiore della fig. 3.31, si occupa invece di

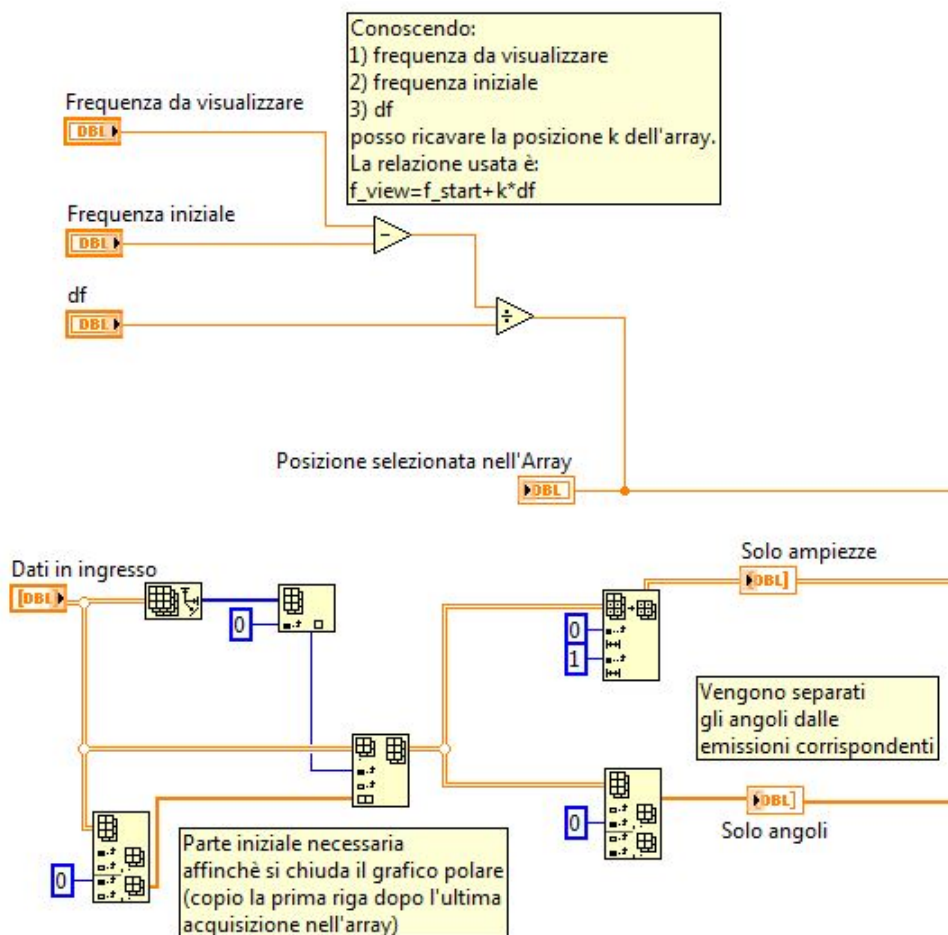


Figura 3.31: La prima parte del subVI che genera il diagramma polare.

separare i dati entranti in ampiezze e angoli corrispondenti.

Prima della divisione il programma copia l'intera prima riga dopo l'ultima acquisita. Questa operazione è necessaria affinché il diagramma polare si chiuda. Il subVI infatti unisce tutti i punti presenti nei dati acquisiti e, poichè il motore non effettua due acquisizioni per uno stesso angolo, alla fine il diagramma risulta aperto per l'ultimo tratto. Per evitare che questo accada viene effettuata questa operazione di copiatura. Successivamente vengono separati gli angoli ( prima colonna ) dal resto della matrice ( tutte le altre colonne ).

In fig. 3.32 viene mostrato il codice che si occupa di calcolare il minimo e il massimo assoluto acquisito dallo strumento. Utilizzando la matrice contenente le ampiezze acqui-

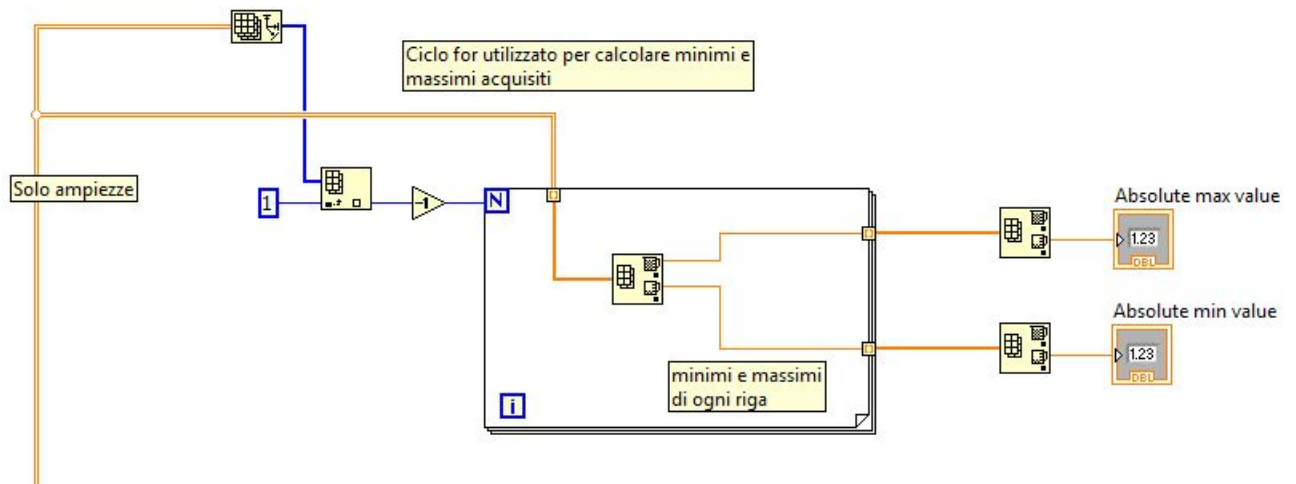


Figura 3.32: La seconda parte del subVI che genera il diagramma polare.

site, viene prima calcolato il minimo e il massimo relativo per ogni riga, attraverso l'uso di un apposito blocco di LabVIEW, poi viene calcolato il minimo e il massimo assoluto tra tutti i minimi e i massimi calcolati precedentemente. Per fare questo è stato utilizzato un ciclo for che viene ripetuto per ogni riga dell'array bi-dimensionale.

Nella **terza parte** del diagramma a blocchi, si veda la fig. 3.33, viene selezionata la colonna da visualizzare nel diagramma. Se il minimo assoluto è minore di zero, allora viene sommato ad ogni valore della colonna il modulo di tale valore. In questo modo, poichè il diagramma polare vuole come coordinate modulo e fase, si avrà la certezza di avere tutti i valori del modulo maggiori o uguali a zero e, scegliendo di sommare a tutte le eventuali colonne selezionate lo stesso minimo, avrò la stessa traslazione per tutte le misure e pertanto verranno mantenute le differenze tra i valori riportati. In particolare basterà sottrarre ai valori riportati nel diagramma il modulo del valore assoluto ( mostrato nel pannello frontale esattamente sopra al diagramma ) per avere il valore reale acquisito.



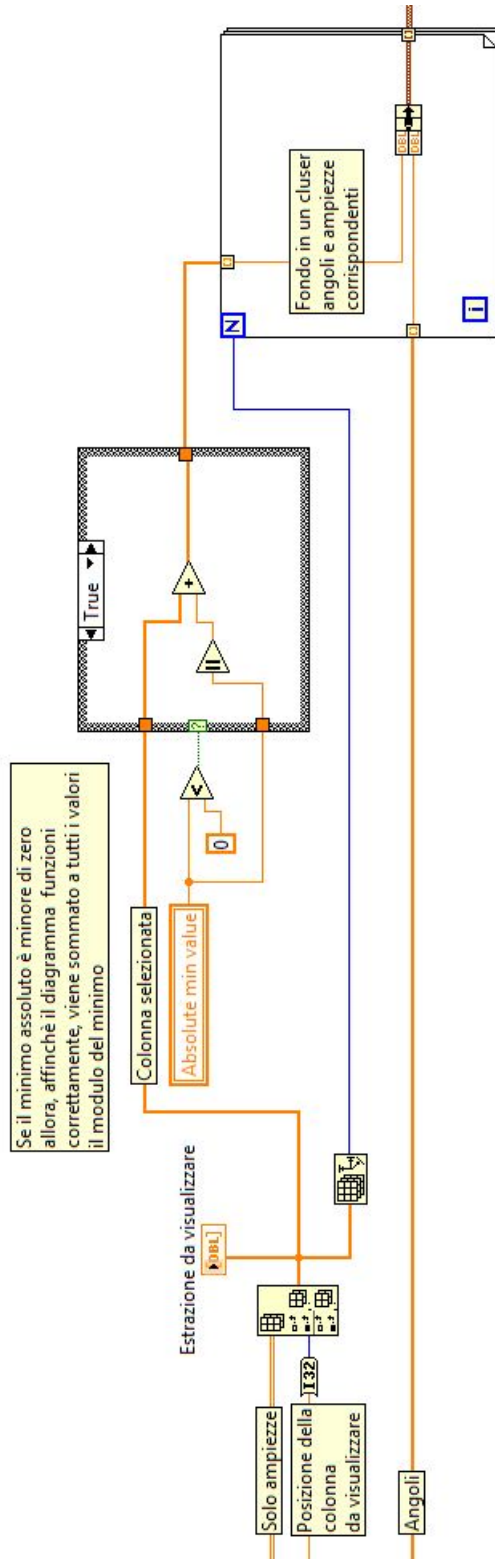


Figura 3.33: La terza parte del subVI che genera il diagramma polare.





Figura 3.34: La quarta parte del subVI che genera il diagramma polare.

Nella **quarta parte** viene utilizzato il subVI di LabVIEW che genera, dato un cluster contenente modulo e fase, il diagramma polare, si veda fig. 3.34.

### 3.2.4 Generatore di report

L'ultima parte del programma si occupa di generare un report, in automatico, delle misure acquisite.

Per una determinata frequenza, infatti, il diagramma polare visualizza le radiazioni ottenute in funzione dell'angolo. La descrizione del report è già stata fatta nella sezione corrispondente del pannello frontale, pertanto in questa parte verrà illustrato esclusivamente il codice del diagramma a blocchi.

Il codice che genera il diagramma è principalmente diviso in due parti: la prima si occupa della gestione dell'immagine rappresentante il diagramma polare, mentre la seconda crea il documento in html che include l'immagine precedentemente salvata.

Nella fig. 3.35 il programma visualizza una finestra di dialogo. In questa viene chiesto

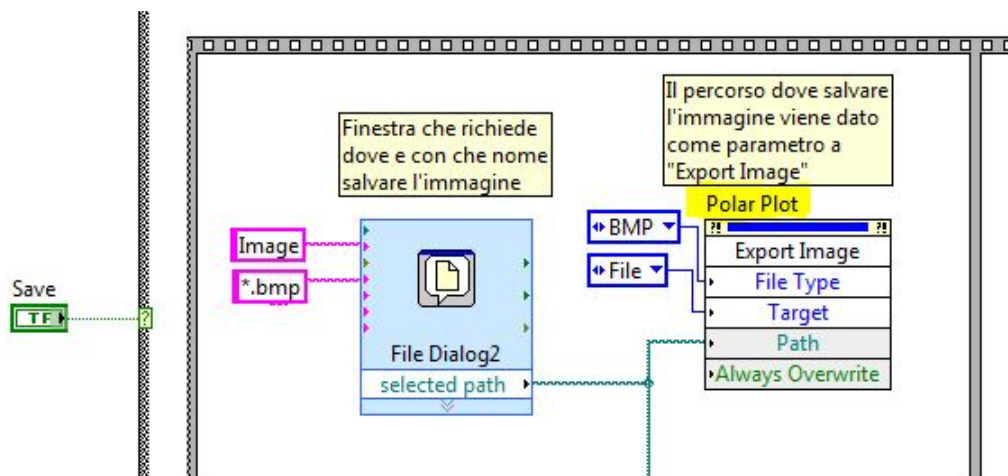


Figura 3.35: La prima parte del programma che genera il report.

all'utente il nome dell'immagine, che di default viene impostato a *image*, e la posizione nella quale questa deve essere salvata.

- In particolare osservando il blocco in fig. 3.35 *File Dialog2*, si nota come venga specificato, oltre al nome di default, il formato nel quale l'immagine viene salvata (bmp). Nella finestra di dialogo che appare per il salvataggio è possibile cambiare

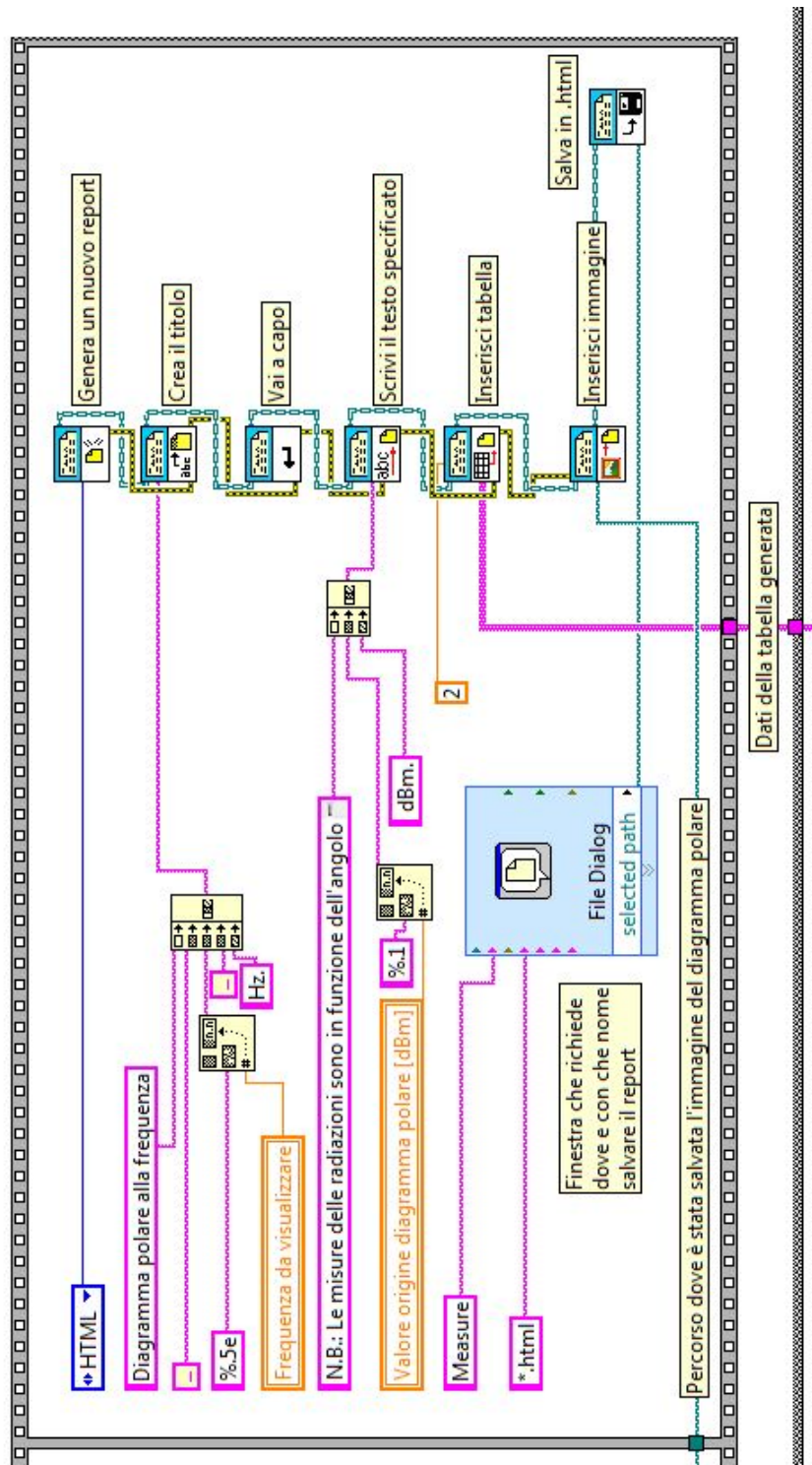


Figura 3.36: La seconda parte del programma che genera il report.

tale formato.

- Successivamente si ha, nella fig. 3.36, la parte del programma che si occupa di

generare il documento html, contenente dati, testo e l'immagine del diagramma polare.

Nella fig. 3.36 possiamo considerare il programma come se fosse diviso in due parti: una parte, corrispondente all'area situata sulla sinistra dell'immagine, contiene le opzioni e le informazioni necessarie per il funzionamento della seconda parte, ovvero i blocchi contenuti sull'area destra della fig. 3.36.

I blocchi dell'area appartenente alla parte destra dell'immagine sono collegati in modo seriale: un blocco passa al successivo il report che fino a quel momento è stato generato.

Ad esempio se il primo blocco si occupa di generare il titolo, esso passerà il titolo al secondo blocco. Se il secondo blocco genera del testo, esso passerà al blocco successivo il titolo e anche il testo.

In questa seconda parte si vedono entrare due *wire* provenienti da altre aree del programma. Il filo verde-acqua porta l'informazione relativa all'immagine che deve essere inserita, mentre il filo fucsia contiene gli stessi dati visualizzati nel diagramma polare che vengono, nel report, inseriti in una tabella. Questa è composta da due colonne: nella prima ci sono gli angoli per i quali sono state effettuate le acquisizioni, sulla seconda ci sono le relative acquisizioni.

Le funzioni dei rispettivi blocchi sono indicate in fig. 3.36.

Il generatore di report viene eseguito solo quando viene premuto il tasto *save*. Se si analizza il diagramma a blocchi infatti, si vede come il generatore di report sia contenuto in un costrutto case, il quale risulta essere contenuto in un costrutto while, condizionato dallo stato di *on* dello strumento. La struttura è schematicamente riportata in fig. 3.37.

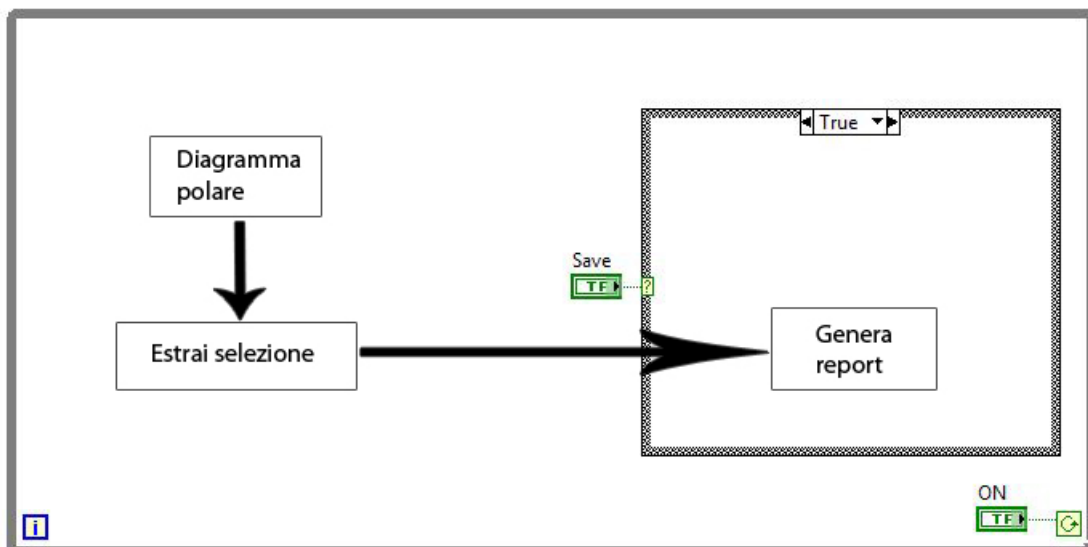


Figura 3.37: Schema dei costrutti utilizzati per visualizzare il diagramma polare e generare il report.



## Capitolo 4

# Utilizzo dello strumento virtuale

In questo capitolo verranno mostrati dettagliatamente i passi e i parametri da settare per effettuare misure con lo strumento virtuale.

L'esecuzione dello strumento segue il flusso riportato in fig. 4.3.

1. Inizialmente si chiede che vengano scelte la frequenza centrale e lo span dello strumento. A tal proposito si ricorda che, se si volesse utilizzare lo strumento in modalità simulazione, non tutte le frequenze possono essere utilizzate. La scelta delle frequenze dovrà essere ponderata considerando sia la sorgente in esame che l'antenna utilizzata. Possibilmente il range di frequenze analizzate dovrà trovarsi all'interno della banda passante dell'antenna collegata all'ASA. Si chiede inoltre di scegliere la risoluzione delle misure. Per questa scelta si dovrà tenere conto che un numero maggiore di misure influenza il tempo utilizzato per effettuare la misura, migliorando però al contempo la risoluzione del diagramma polare. La scelta dipenderà dallo scopo della misura e in ogni caso l'utente potrà interrompere l'esecuzione e cambiare tale parametro.
2. La scelta di utilizzare la modalità di simulazione piuttosto che quella normale influenza direttamente le prossime scelte. Si indicherà con l'opzione a) il procedimento legato alla modalità simulazione, mentre con l'opzione b) il funzionamento con gli strumenti.
3. a) Si scelga il tipo di segnale che si vuole simulare e la rispettiva ampiezza.  
b) Si scelga l'analizzatore di spettro disponibile dal controllo *VISA resource name*. Altro parametro da settare sono i punti ( *points* ) acquisiti dall'analizzatore di spettro. La scelta di tale valore deve essere fatta controllando l'opportuno indicatore dello strumento.
4. a) Viene simulata la rotazione del motore.  
b) Viene fatto ruotare il motore.  
Per entrambi l'angolo di rotazione è lo stesso specificato nella sezione *risoluzione misure*.
5. a) Viene simulata l'acquisizione di uno spettro.  
b) Viene acquisito lo spettro dall'ASA.
6. Lo spettro acquisito e il rispettivo angolo di rotazione vengono inseriti in un array. Ogni nuova rotazione e acquisizione inserisce i dati in una nuova riga, formando una matrice.

7. Se il motore ha completato un'intera rotazione la misurazione è conclusa, altrimenti si ripetono i punti 4), 5) e 6) finché la rotazione non viene eseguita per l'intero angolo giro.
8. Quando le acquisizioni spettrali per un'intera rotazione della piattaforma sono state eseguite, è possibile visualizzare il diagramma polare. Esso visualizza, per una determinata frequenza scelta, le ampiezze spettrali emesse in funzione dell'angolo raggiunto dal motore. Nella tabella mostrata in fig. 4.1 viene riportata una

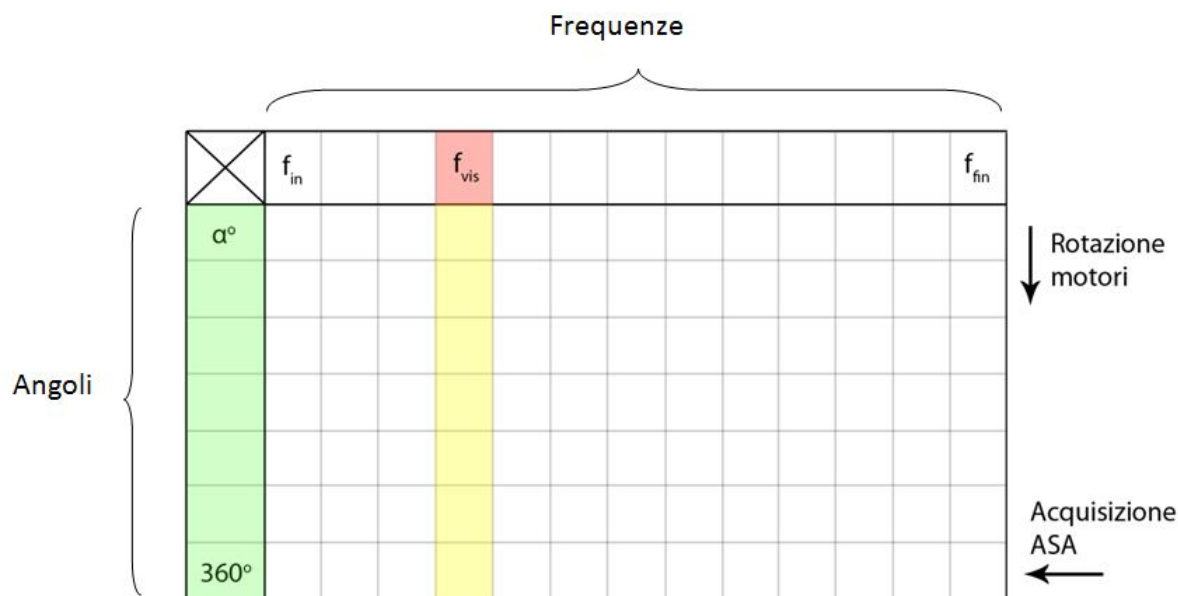


Figura 4.1: Tabella rappresentate i dati acquisiti.

rappresentazione schematica dei dati acquisiti. La colonna evidenziata in giallo contiene i dati visualizzati nel diagramma polare, mentre la cella evidenziata in rosso corrisponde alla frequenza selezionata ( $f_{vis}$ ). Spostando tale frequenza la colonna da visualizzare si sposterà a destra o a sinistra (rispettivamente verso  $f_{fin}$  o verso  $f_{in}$ ).

Riassumendo: ciascuna riga contiene le ampiezze acquisite dall' ASA (o simulate) mentre ciascuna colonna, contiene le ampiezze per una stessa frequenza, a diversi angoli di rotazione.

9. Quando viene visualizzato il diagramma polare le misurazioni sono state completate. È possibile salvare il diagramma polare con i dati corrispondenti premendo il tasto *save* nel pannello frontale. I dati inseriti nel report sono:
  - gli angoli di rotazione;
  - le rispettive ampiezze per una frequenza scelta.

In pratica ciò che contiene il report sarà una tabella contenente le colonne evidenziate in verde e in giallo in fig. 4.1. I dati saranno relativi alla frequenza evidenziata in rosso e pertanto questa comparirà nel titolo del documento. Infine vi sarà l'immagine del diagramma polare. Di seguito si mostra un report generato dal VI di LabVIEW.

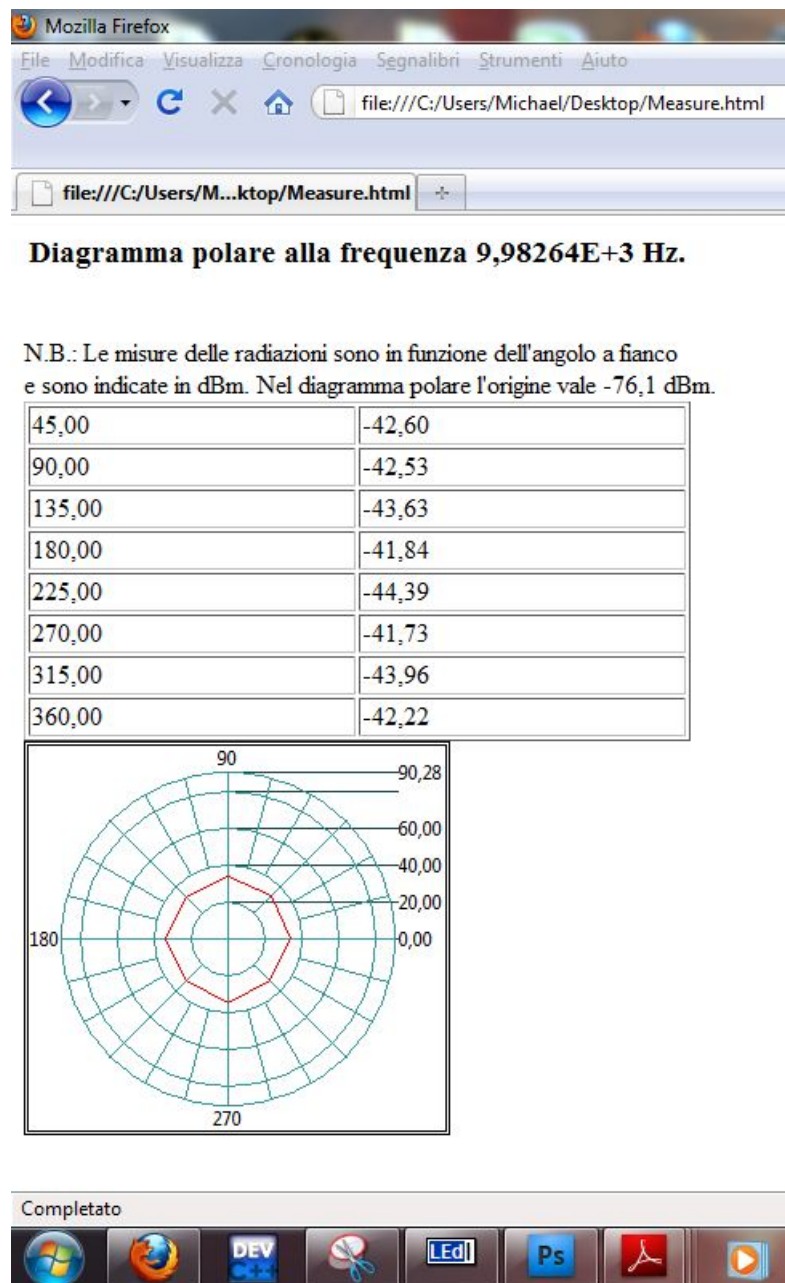


Figura 4.2: Un esempio di report generato automaticamente premendo il tasto *Save*.

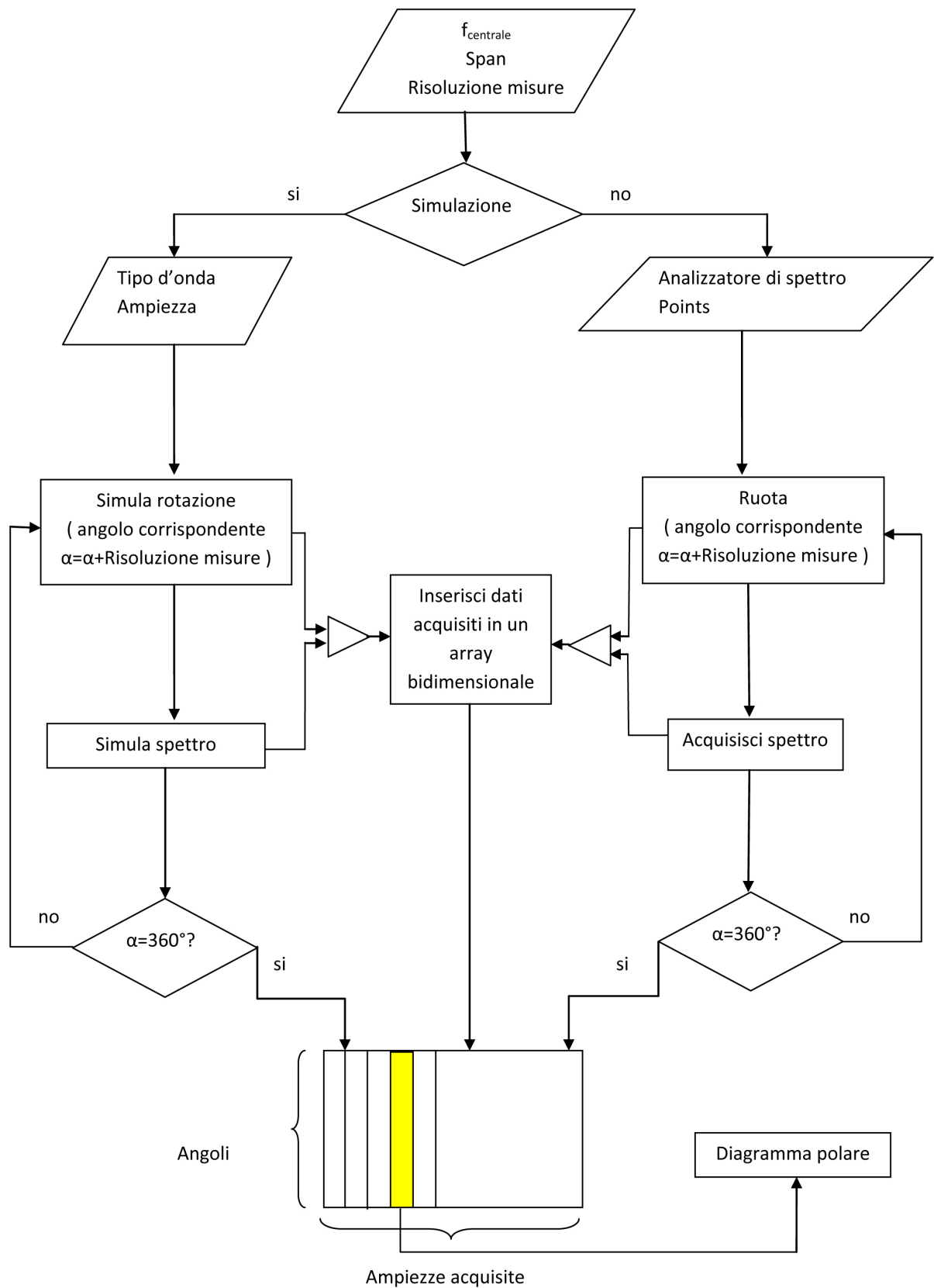


Figura 4.3: Schematizzazione del funzionamento logico del programma.



## Capitolo 5

# Conclusioni

Lo strumento virtuale realizzato consente all'operatore di gestire l'intero sistema di misura automaticamente. Il pannello frontale semplifica le scelte che l'utente può eseguire sull'analizzatore di spettro, limitandole alle sole necessarie per l'esecuzione della misura, e riesce ad astrarre ad un livello più alto il funzionamento dei singoli strumenti utilizzati. Ulteriori miglioramenti ed eventuali sviluppi possono essere applicati al VI realizzato. In questo capitolo se ne vedranno alcuni esempi.

### 5.1 Utilizzo con apparecchio adatto allo spostamento della piattaforma

Il sistema che avrebbe dovuto muovere la piattaforma sottostante la sorgente da analizzare, presentato in fig. 3.7, non si è rivelato funzionante. In particolare gli ingranaggi adatti al prolungamento dell'albero del motore non ruotavano e pertanto l'utilizzo di tale apparecchio è stato escluso, non essendo possibile sostituire in tempo utile i motori passo-passo già presenti con altri, in grado di sviluppare una coppia maggiore.

Sarebbe quindi opportuno, in un sviluppo futuro di questo lavoro di tesi, realizzare un nuovo apparecchio funzionante, che gestisca adeguatamente la piattaforma, senza causare interferenze con la sorgente sotto esame. Si dovrà quindi tarare opportunamente la rotazione della piattaforma in base alla rotazione del rotore: essendo l'azione del motore mediata da diversi ingranaggi prima di raggiungere la piattaforma, il rapporto *step-angolo di rotazione piattaforma* potrebbe subire un cambiamento rispetto al rapporto diretto esistente nel programma sviluppato in questo lavoro di tesi, pertanto andrebbe apportato un opportuno cambiamento.

Supponendo inoltre che l'apparecchio realizzato permetta lo spostamento della piattaforma lungo più direzioni, sarebbe opportuno implementare nel VI la possibilità di effettuare ulteriori misurazioni anche in funzione dello spostamento della piattaforma lungo le altre direzioni. Infine si potrebbe pensare alla possibilità di avere diagrammi polari selezionabili sia in funzione della frequenza sia in funzione dello spostamento lungo uno o più assi di spostamento. I dati acquisiti da questo sistema, supponendo che la piattaforma si muova lungo una sola direzione, sarebbero acquisiti in un array tri-dimensionale: la generazione di un diagramma polare completo, per un'intera rotazione della piattaforma, sarebbe ripetuto durante lo spostamento lungo la direzione di spostamento.

Nella fig. 5.1 la colonna evidenziata rimane il diagramma polare visualizzato. In questa implementazione il diagramma cambierà sia se si sposta la frequenza per la quale voglia-

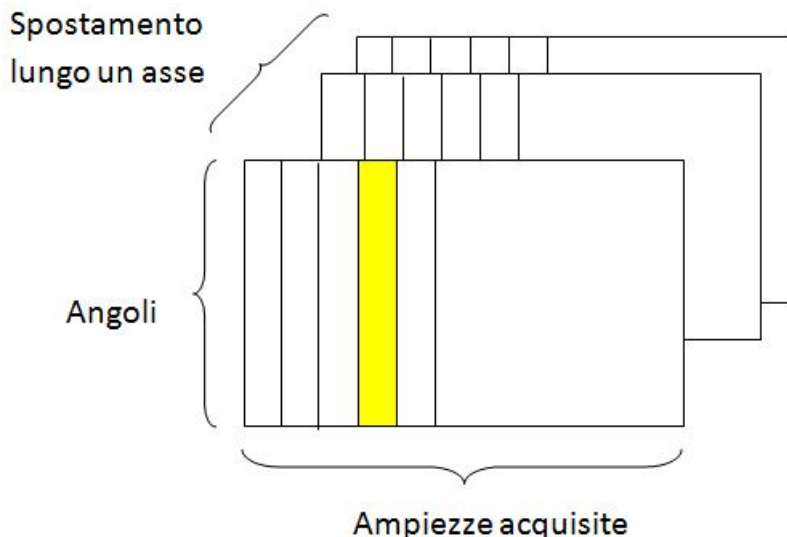


Figura 5.1: Dati acquisiti per un'eventuale sviluppo del programma.

mo che venga mostrato, sia se si decide di visualizzarlo ad una distanza ad esempio di 5, 10 o 15 cm dalla fine corsa di un ipotetico spostamento della piattaforma.

La risoluzione, ovvero la distanza tra le misurazioni, comporta un aumento delle misurazioni da effettuare e un incremento del tempo totale. Riprendendo l'eq. 3.3:

$$T_{tot} = T_{elaborazione} \cdot (T_{rotazione} + T_{sweep}) \cdot \text{risoluzione misure} \quad (5.1)$$

dove “risoluzione misure” è il *prodotto* tra le misure necessarie per completare la rotazione e quelle per completare lo spostamento lungo l'asse di spostamento.

## 5.2 Modalità simulazione avanzata

Un eventuale sviluppo del programma potrebbe riguardare la possibilità di modificare altri parametri nella modalità simulazione, come ad esempio la frequenza di campionamento, implementandoli nel pannello frontale.

Tale modifica dovrebbe essere opportunamente ponderata per non appesantire il pannello frontale e ad ogni modo dovrebbe essere giustificata dallo scopo dello strumento virtuale.

## 5.3 Possibilità di visualizzare lo spettro acquisito per diversi angoli

La traccia che visualizza lo spettro acquisito dall'ASA viene sovrascritta ogni volta che viene effettuata una nuova acquisizione.

Potrebbe essere utile implementare la possibilità di visualizzare, selezionando uno degli angoli di rotazione per i quali sono state effettuate le misure, il grafico spettrale corrispondente.

Si potrebbe ad esempio implementare un controllo sul pannello frontale adatto a selezionare un angolo di rotazione. In questo modo si potrebbe modificare sia il diagramma polare, variando la frequenza da visualizzare, sia il grafico che mostra lo spettro, variando l'angolo di acquisizione.



# Bibliografia

- [1] Agilent Spectrum Analysis Basics: application note 150.
- [2] LabVIEW Help.
- [3] User guide and specifications NI USB-6008/6009.
- [4] Wikipedia english alla voce *stepper motor*.
- [5] G.Giorgi: Slides del corso “misure per l’automazione e la produzione industriale”
- [6] Lavoro di tesi di Alessandro Inglese.
- [7] N.Benvenuto,R.Corvaja,T.Erseghe,N.Laurenti: Communication Systems.
- [8] Dispense di misure per l’automazione e la produzione industriale (prima parte):  
L.Benetazzo, C.Narduzzi.
- [9] Halliday, Resnick, Krane: Fisica 2.
- [10] L.Malesani, P.Mattavelli, G.Spiazzini: Elettronica industriale.
- [11] J.Rabaey,A. Chandrakasan, B.Nilolić: Circuiti integrati digitali.
- [12] Lavoro di tesi di F. Masieri.
- [13] M.Guarnieri,A.Stella: Principi ed applicazioni di elettrotecnica.