

**ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA**

**Corso di laurea in Ingegneria Informatica (9254)  
Sede di Bologna**

**Anno Accademico 2021/2022**

---

# **RELAZIONE DI FINE TIROCINIO**

*Studente:* Lucia Gasperini (matricola nr. 0000921439)

*Sede del tirocinio:*  
DEI - Laboratorio Misure  
Viale del Risorgimento 2, Bologna 40126

**Tutor Accademico:**  
Prof. Roberto Tinarelli

## Sommario

<b>INTRODUZIONE.....</b>	<b>3</b>
<b>OBIETTIVO .....</b>	<b>3</b>
<b>BREVE INTRODUZIONE A LABVIEW .....</b>	<b>4</b>
Programmazione G.....	4
Dettagli dei VI.....	4
<b>CODICE LABVIEW.....</b>	<b>6</b>
<b>GENERATORE DI FUNZIONI .....</b>	<b>6</b>
FUNZIONAMENTO .....	6
INTERFACCIA GRAFICA e IMPLEMENTAZIONE.....	7
COMPONENTI VISA UTILIZZATI.....	8
STRUTTURE UTILIZZATE.....	9
<b>OSCILLOSCOPIO .....</b>	<b>11</b>
FUNZIONAMENTO .....	11
COMPONENTI VISA UTILIZZATI.....	14
COMPONENTI DRIVER TEKTRONIX UTILIZZATI .....	14
PULSANTI UTILIZZATI .....	15
COMPONENTI DI ANALISI DEL SEGNALE UTILIZZATI.....	15
COMPONENTI PER LA SCRITTURA SU FILE UTILIZZATI .....	17
STRUTTURE UTILIZZATE.....	17
ESEMPIO DI OUTPUT GENERATO DA UNA SCRITTURA SU FILE .....	18
<b>CONSIDERAZIONI FINALI.....</b>	<b>18</b>
<b>BIBLIOGRAFIA.....</b>	<b>19</b>

# INTRODUZIONE

La seguente relazione ha per oggetto un tirocinio svolto presso il Laboratorio di Misure del dipartimento di Ingegneria dell'Università degli studi di Bologna.

Il tirocinio è stato condotto nel periodo di tempo che intercorre tra marzo 2022 e giugno 2022, per una durata complessiva di 150 ore, corrispondenti a 6 crediti formativi.

## OBIETTIVO

La finalità dell'attività è stata quella di acquisire le competenze e le abilità tali da garantire utilizzo e padronanza del software LabView in modo da poter creare un'interfaccia grafica per il controllo di strumenti.

La strumentazione utilizzata è stata un generatore di funzioni nel modello DG1022z della Rigol (figura 1) e un oscilloscopio nel modello MSO58 della Tektronix (figura 2).



Figura 1

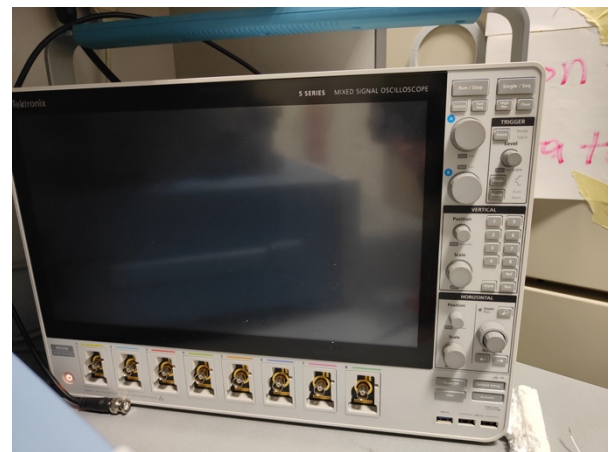


Figura 2

Gli obiettivi del progetto possono essere così riassunti:

- Studio individuale del funzionamento dell'applicativo LabView
- Studio del funzionamento della strumentazione presente in laboratorio
- Creazione di un codice per l'interfacciamento con il generatore in modo da poterlo controllare tramite LabView
- Creazione di un codice per l'interfacciamento con l'oscilloscopio in modo da poterlo controllare tramite LabView
- Nell'oscilloscopio, tramite software, deve essere possibile:
  - modificare la scala dei tempi (asse orizzontale)
  - modificare la scala delle tensioni (asse verticale)
  - determinare la modalità di acquisizione (continua o one shot)
  - configurare il trigger
  - trasferire su file la sequenza di punti acquisiti.

# BREVE INTRODUZIONE A LABVIEW

LabVIEW (abbreviazione di Laboratory Virtual Instrumentation Engineering Workbench) è l'ambiente di sviluppo integrato per il linguaggio di programmazione visuale di National Instruments. Tale linguaggio grafico viene chiamato Linguaggio G.

Il progetto nasce nel 1983 dalla necessità della National Instruments di disporre di un software di programmazione grafica, con il quale testare rapidamente gli apparati hardware prodotti da tale industria statunitense.

Realizzato originariamente per Apple Macintosh nel 1986, LabVIEW viene principalmente utilizzato per programmi di acquisizione e analisi dati, controllo di processi, generazione di rapporti, o più generalmente per tutto ciò che concerne l'automazione industriale, su diverse piattaforme come Windows, Linux, macOS e controllori National Instruments.

## Programmazione G

Il linguaggio di programmazione usato in LabVIEW si distingue dai linguaggi tradizionali perché la sua sintassi non è scritta ma grafica, e per questa ragione viene chiamato G-Language (Graphic Language). Un programma o sottoprogramma G, denominato VI (Virtual Instrument), non esiste sotto forma di testo, ma può essere salvato solo come un file binario che può essere aperto e compilato solo da LabVIEW.

La definizione di strutture dati ed algoritmi avviene con icone e altri oggetti grafici, ognuno dei quali incapsula funzioni diverse, uniti da linee di collegamento, in modo da formare una sorta di diagramma di flusso. Questo tipo di linguaggio viene definito dataflow in quanto la sequenza di esecuzione è definita e rappresentata dal flusso dei dati stessi attraverso i fili monodirezionali che collegano i blocchi funzionali. Poiché i dati possono anche scorrere in parallelo attraverso blocchi e fili non consecutivi, il linguaggio può realizzare spontaneamente il multithreading senza bisogno di esplicita gestione da parte del programmatore.

## Dettagli dei VI

Nell'ambiente di sviluppo, i VI sono composti da tre parti principali:

- il pannello frontale
- lo schema a blocchi
- il riquadro connettori

Il **pannello frontale** è l'interfaccia utente del VI. Si realizza con controlli e indicatori, che costituiscono i terminali interattivi d'ingresso e d'uscita, rispettivamente. Essi sono ben più numerosi e complessi dei widget normalmente forniti dal sistema operativo. I controlli sono matrici, manopole, potenziometri, pulsanti, quadranti e molti altri; simulano i dispositivi d'ingresso degli strumenti e forniscono dati allo schema a blocchi del VI. Gli indicatori sono grafici, tabelle, LED, termometri e molti altri; simulano i dispositivi d'uscita degli strumenti e visualizzano i dati che lo schema a blocchi acquisisce o genera.

Lo **schema a blocchi** è il diagramma di flusso che rappresenta il codice sorgente, in formato grafico. Gli oggetti del pannello frontale appaiono come terminali di ingresso o uscita nello schema a blocchi. Gli oggetti dello schema a blocchi comprendono:

- terminali
- funzioni
- costanti
- strutture
- chiamate ad altri VI (subVI)
- fili di collegamento
- commenti testuali

Le funzioni sono esse stesse dei VI, anche se non hanno un loro pannello frontale e un loro schema a blocchi. Possono avere un numero indefinito di ingressi e di uscite come ogni VI.

I fili di collegamento possono trasportare, teoricamente, qualunque mole di dati di qualunque tipo, anche aggregati definiti dal programmatore. I fili possono essere di diverso spessore e colore per permettere una facile identificazione dei dati che vi scorrono: ad esempio i numeri interi scorrono su fili blu, i numeri decimali su fili arancioni le stringhe su fili rosa.

Lo schema a blocchi può essere reso visibile anche durante l'esecuzione, cosa molto utile in fase di debug, in quanto esiste la possibilità di visualizzare con un'animazione al rallentatore il movimento dei dati lungo i fili e il loro valore momentaneo.

Ogni VI può essere a sua volta utilizzato come subVI e comparire all'interno dello schema a blocchi di altri VI come una qualsiasi funzione, e come tale può avere ingressi e uscite a cui collegare le linee di flusso. Il **riquadro connettori** serve appunto a definire qual è l'aspetto del VI quando appare come subVI in uno schema a blocchi di un VI più ampio: che facciata ha l'icona, ma soprattutto come e dove vanno collegate le linee per permettere il passaggio dei dati. In genere, con pochi click, ogni controllo può essere associato a un ingresso e ogni indicatore può essere associato a un'uscita.

# CODICE LABVIEW

## GENERATORE DI FUNZIONI

Il seguente codice permette all'utilizzatore di scegliere tra diverse funzioni standard di output e di settare l'ampiezza, la frequenza e l'offset dell'onda.


Nella realizzazione del codice sono stati utilizzati driver Visa generici permettendo, in questo modo, l'utilizzo di generatori diversi da quello presente nel laboratorio.

## FUNZIONAMENTO

- Collegare lo strumento alla corrente e accenderlo
- Collegare lo strumento tramite porta USB al computer
- Selezionare dal "VISA resource name" il nome del generatore con il quale ci si vuole interfacciare.

Il formato standard di nominativo che viene visualizzato è qualcosa di questo tipo:

`"USB0::0x1AB1::0x0588::DS1ED131303966::INSTR"`

- Premere il pulsante di start da LabView per avviare il programma 
- Selezionare il tipo di onda desiderata
- Impostare i valori di ampiezza, offset e frequenza desiderati nelle altre combo box
- Abilitare una delle due uscite in base al canale a cui è collegato l'oscilloscopio
- Premere il pulsante di configurazione per inviare la nuova configurazione allo strumento

NB: Bisogna premere il pulsante di configurazione ogni volta che si fa un cambiamento per riconfigurare lo strumento.

- Premere il pulsante di stop quando si è terminato. Questo permette di chiudere la sessione VISA e ripulire ogni errore che può essersi generato.

## INTERFACCIA GRAFICA e IMPLEMENTAZIONE

Nella figura 3 è possibile vedere l'interfaccia utente (front panel) con la quale è possibile controllare il generatore e modificare i parametri.

Viene data la possibilità di selezionare il dispositivo collegato e il tipo di onda che si desidera con relativa ampiezza, offset e frequenza. Inoltre è possibile selezionare in quale canale abilitare l'output.

Mentre nella figura 4 viene mostrato il back panel, ovvero la parte di LabView che mostra il codice che compone il programma.

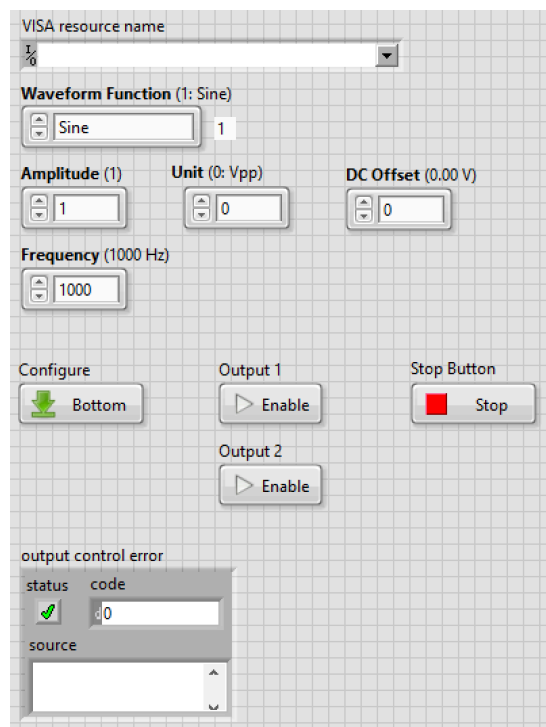


Figura 3

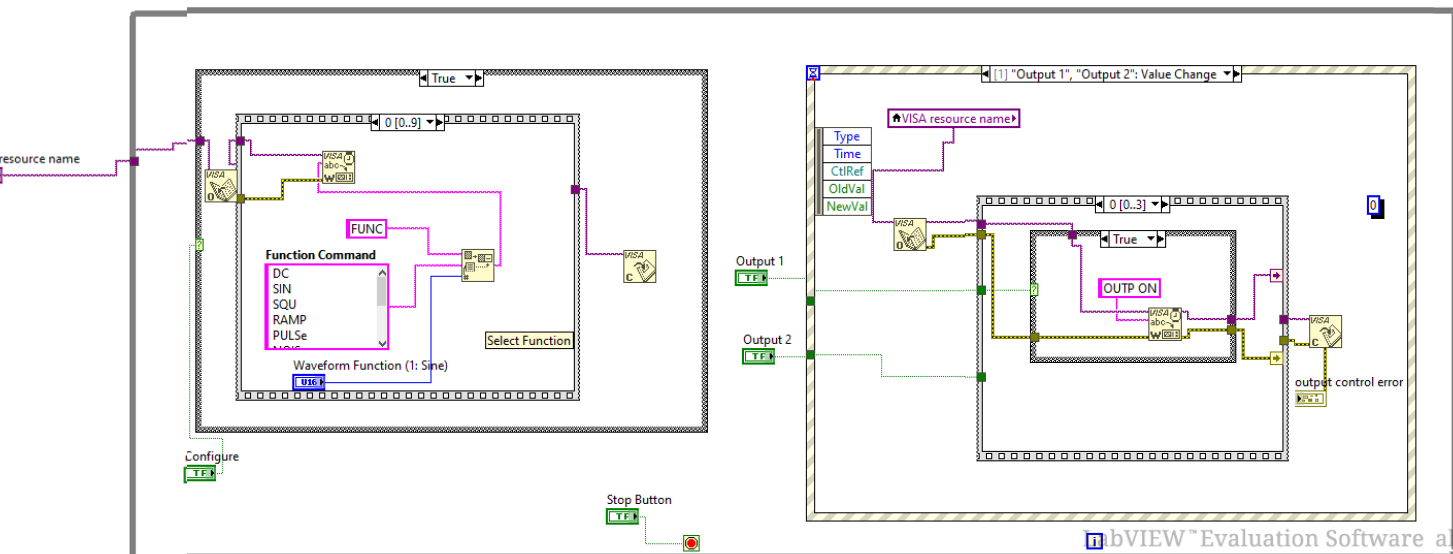


Figura 4

Il blocco a sinistra permette la selezione del tipo di onda da generare nel caso 0 (figura 4), il voltaggio nel caso 2 (figura 5), la frequenza nel caso 4 (figura 5), l'ampiezza nel caso 6 (figura 7) e l'offset nel caso 8 (figura 8).

Nelle casistiche non citate è stato utilizzato una funzione di wait (figura 9).

Il blocco a destra (figura 4) permette di attivare uno canale (tra i due a disposizione nel modello di generatore utilizzato) di uscita al quale verrà collegato l'oscilloscopio.

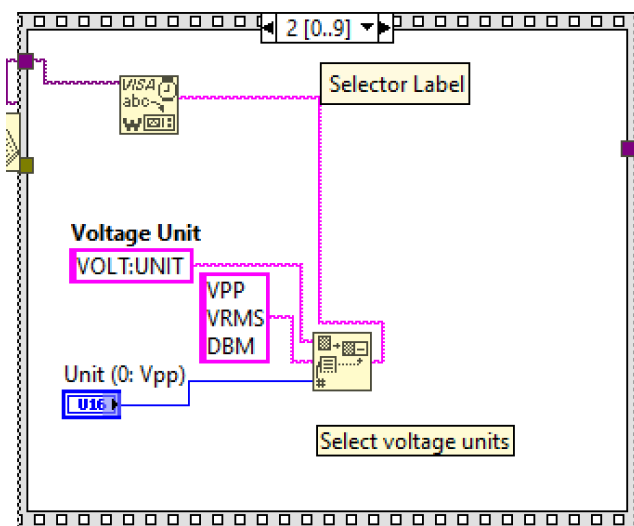


Figura 5

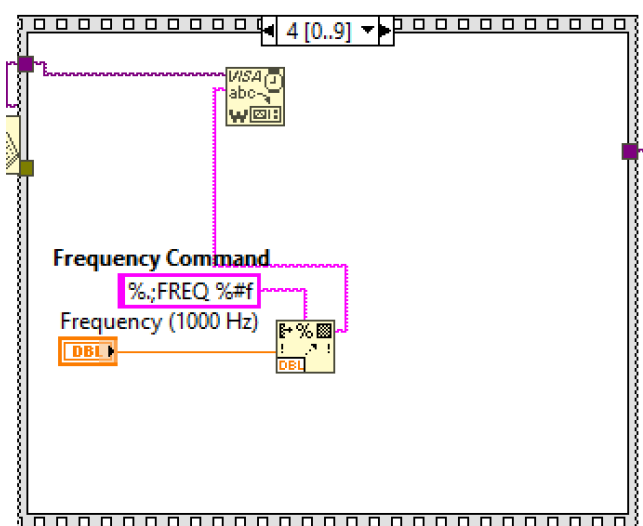


Figura 6

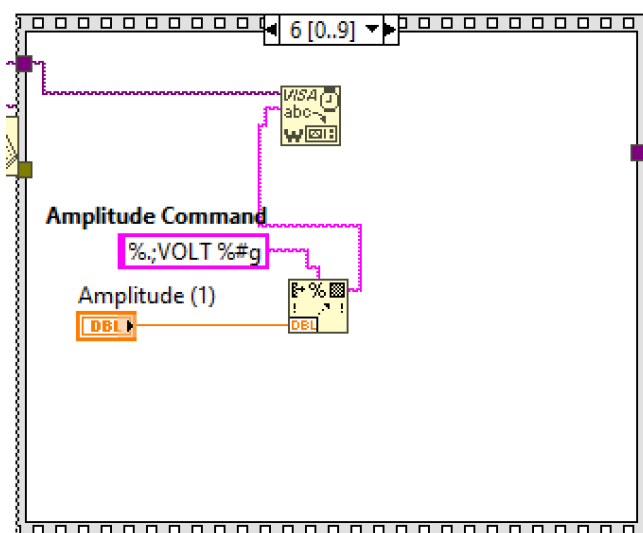


Figura 7

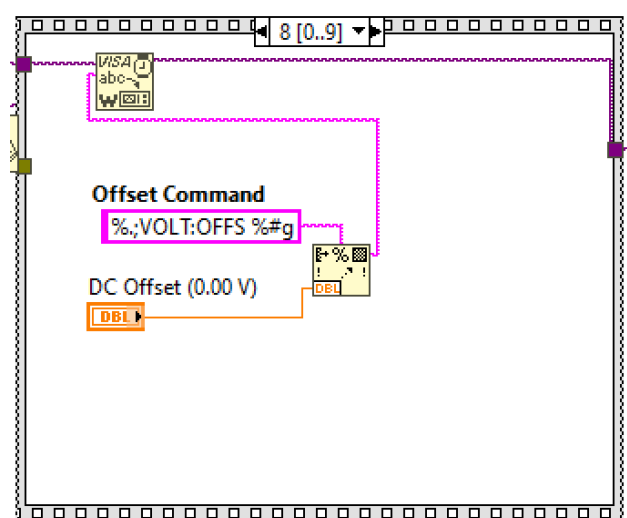


Figura 8

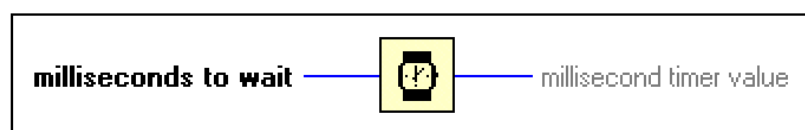


Figura 9

## COMPONENTI VISA UTILIZZATI

Di seguito sono riportati i principali componenti utilizzati:

**OPEN FUNCTION** — Apre una sessione sul dispositivo specificato dal nome della risorsa VISA e restituisce un identificatore di sessione che può essere utilizzato per chiamare qualsiasi altra operazione di quel dispositivo.

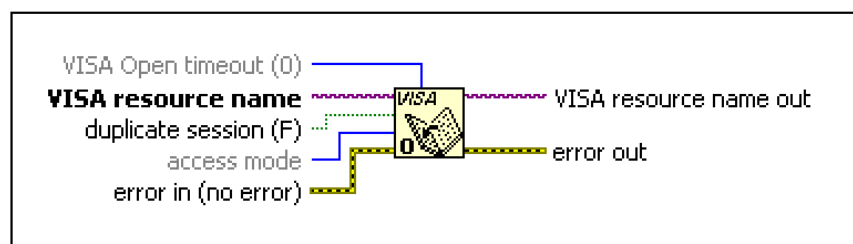


Figura 10



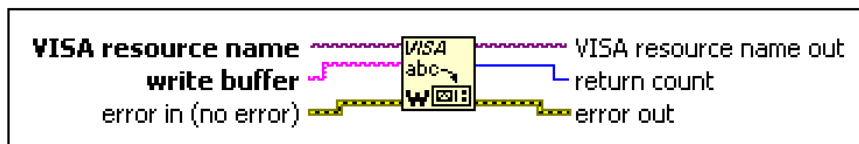


Figura 11

**WRITE FUNCTION** — Scrive i dati dal buffer di scrittura al dispositivo o all'interfaccia specificata dal nome della risorsa VISA.

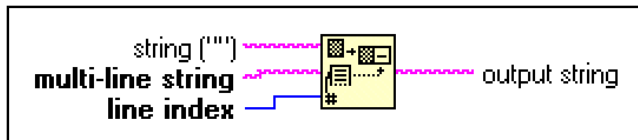


Figura 12

**PICK LINE FUNCTION** — Sceglie una riga da una stringa a più righe e aggiunge quella riga alla stringa. Il riquadro del connettore mostra i tipi di dati predefiniti per questa funzione.

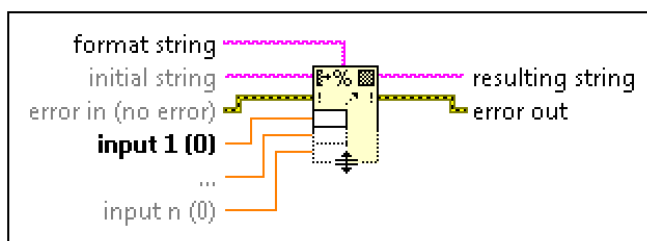


Figura 13

**FORMAT INTO STRING FUNCTION** — Formatta stringhe, percorsi, tipi enumerati, timestamp, booleani o dati numerici come testo.

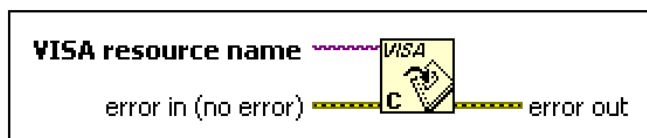


Figura 14

**VISA CLOSE FUNCTION** — Chiude una sessione del dispositivo o un oggetto evento specificato dal nome della risorsa VISA.

## STRUTTURE UTILIZZATE

L'implementazione è stata fatta innestando diverse strutture: un while loop (figura 15) più esterno che permette l'esecuzione del codice fino alla pressione del pulsante di stop, una Case Structure (figura 16) a sinistra con una Flat Sequence Structure (figura 17) come sottodiagramma innestato e una Event Structure (figura 18) a destra con una Case Structure (figura 16) innestata a una Flat Sequence Structure (figura 17).

Inoltre vengono usati dei registri a scorrimento nel ciclo While per trasferire valori da un'iterazione del ciclo alla successiva.



Un registro a scorrimento appare come una coppia di terminali, mostrati a sinistra, con direzione opposta l'uno rispetto all'altro sui lati verticali della cornice del ciclo. Il terminale di destra contiene una freccia verso l'alto e memorizza i dati alla fine dell'iterazione. LabVIEW trasferisce i dati collegati al lato destro del registro alla successiva iterazione.

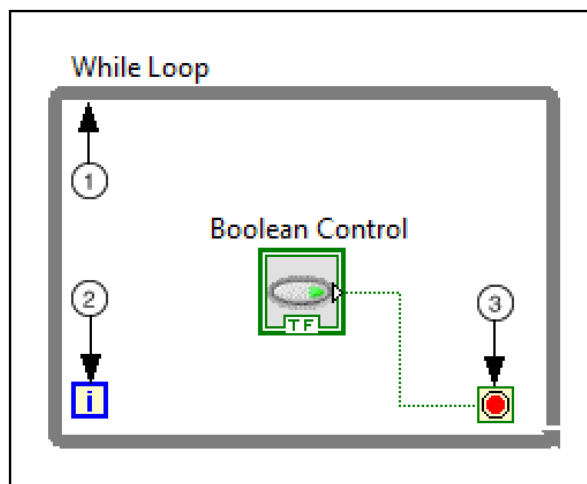


Figura 15

- ① **Sottodiagramma** — Contiene il codice che il while loop esegue una volta per ogni iterazione.
- ② **Terminale di iterazione** — Fornisce il conteggio delle iterazioni del ciclo corrente. Il conteggio del ciclo inizia sempre da zero per la prima iterazione.
- ③ **Terminale di condizione** — valuta un valore di input booleano per determinare se continuare a eseguire il ciclo While. È possibile condizionare la continuazione del ciclo con un pulsante, nel caso del generatore un pulsante di stop.

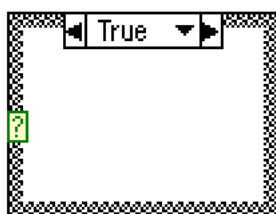


Figura 16

**CASE STRUCTURE** — Contiene uno o più sottodiagrammi, o casi, esattamente uno dei quali viene eseguito quando viene eseguita la struttura. Il valore collegato al selettore caso determina quale caso eseguire. Tramite il label di selezione in alto è possibile visualizzare i valori per i quali viene eseguito il caso associato.

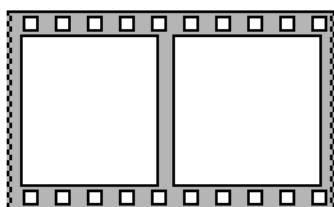


Figura 17

**FLAT SEQUENCE STRUCTURE** — È costituita da uno o più diagrammi secondari, o frame, che vengono eseguiti in sequenza. I frame in una struttura Flat Sequence vengono eseguiti da sinistra a destra e quando tutti i valori dei dati collegati a un frame sono disponibili. I dati lasciano ogni frame al termine dell'esecuzione del frame. Ciò significa che l'input di un frame può dipendere dall'output di un altro frame.

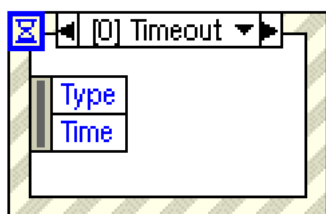


Figura 18

**EVENT STRUCTURE** — Si tratta di una struttura attende fino a quando si verifica un evento, quindi esegue il caso appropriato per gestire tale evento (nel caso del generatore consiste nella selezione del canale 1 o 2). La struttura Event ha uno o più sottodiagrammi, o casi di eventi, esattamente uno dei quali viene eseguito quando la struttura viene eseguita per gestire un evento. Questa struttura può scadere durante l'attesa della notifica di un

evento. Collega un valore al terminale Timeout in alto a sinistra della struttura Evento per specificare il numero di millisecondi che la struttura Evento attende per un evento. Il valore predefinito è -1, che indica di non scadere mai.

## OSCILLOSCOPIO

Il seguente codice permette all'utilizzatore regolare il range di rappresentazione dell'onda generata dal generatore di funzioni, il livello di trigger, la base dei tempi, la posizione e l'ending point.

### FUNZIONAMENTO

- Collegare lo strumento alla corrente e accenderlo
- Collegare lo strumento tramite porta USB al computer
- Selezionare dal "VISA resource name" il nome del generatore con il quale ci si vuole interfacciare.
- Premere il pulsante di start da LabView per avviare il programma ➡
- Selezionare il tipo di onda desiderata
- Impostare i valori desiderati
- Premere il pulsante di stop quando si è terminato. Questo permette di chiudere la sessione VISA e ripulire ogni errore che può essersi generato.

Nella figura 19 è possibile vedere l'interfaccia utente con la quale è possibile controllare il generatore e modificare i parametri.

Mentre nella figura 20 viene mostrato il back panel, ovvero la parte di LabView che mostra il codice che compone il programma.

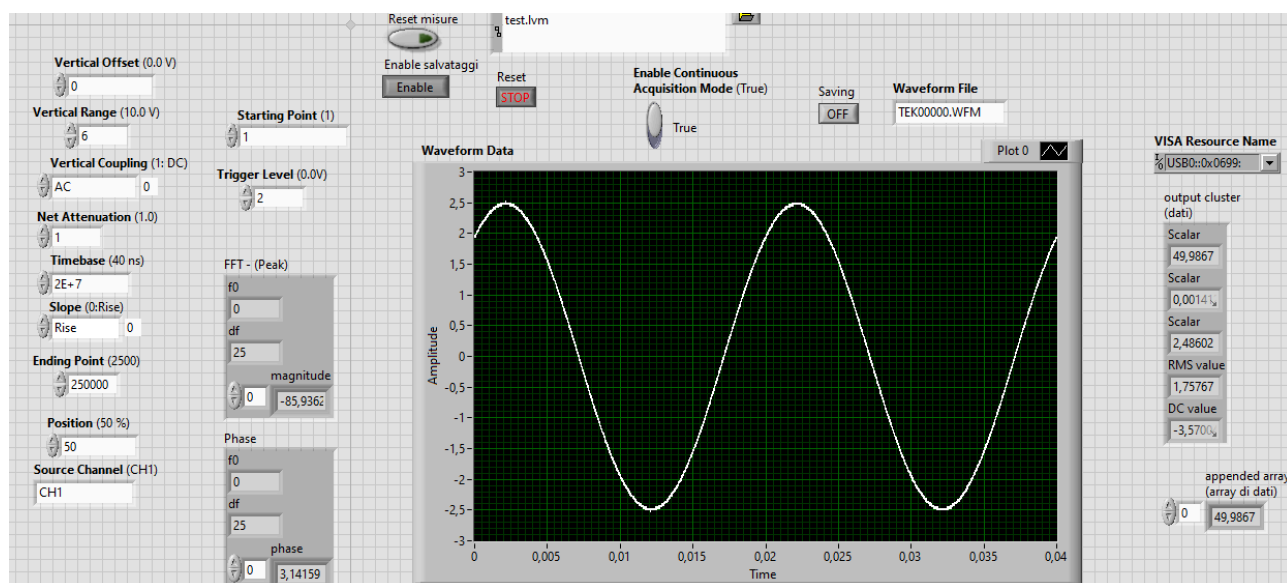


Figura 19

L'interfaccia dell'oscilloscopio permette, dopo aver collegato lo strumento, di modificare i vari parametri, tra cui il tipo di onda, la frequenza e la base dei tempi. All'avvio dell'interfaccia sarà possibile vedere le modifiche sullo strumento stesso.

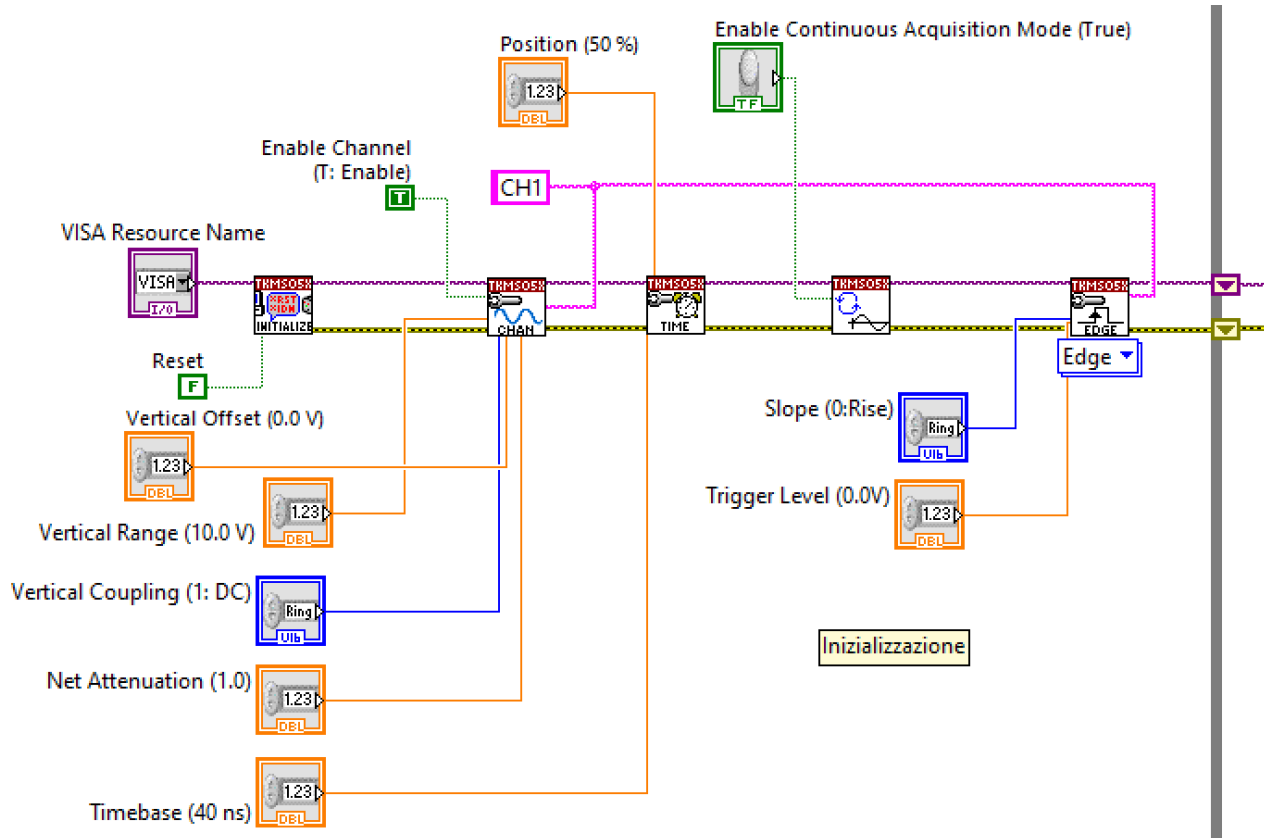
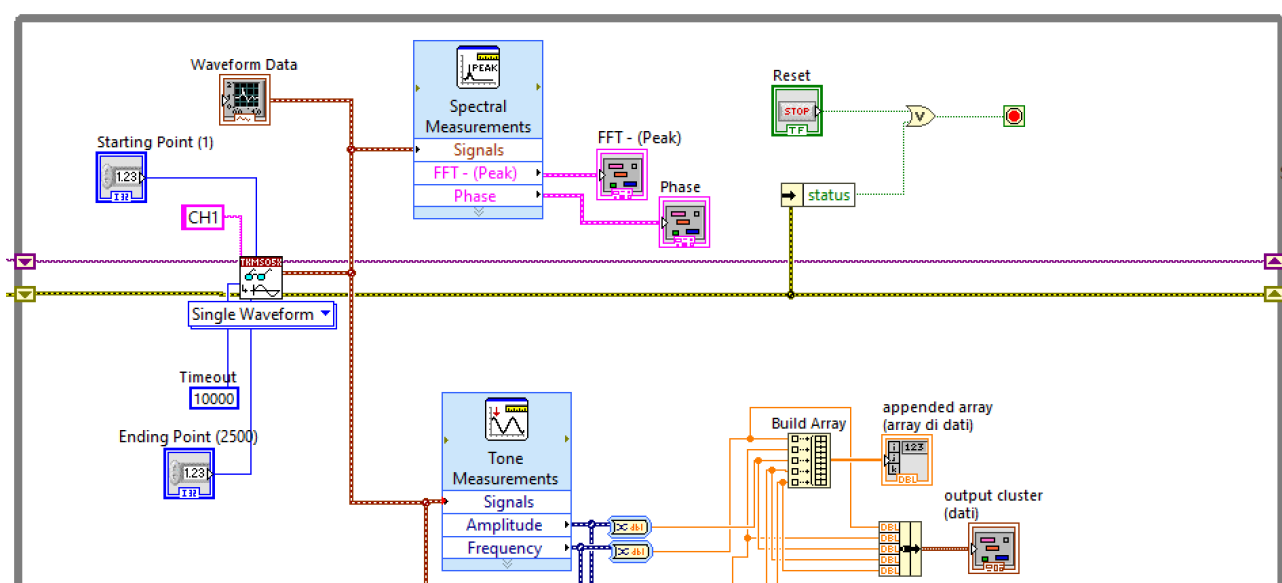


Figura 20.1

Appena viene selezionato lo strumento collegato tramite l'apposito VISA Resource Name viene abilitato il canale alla lettura ed è possibile impostare i valori di offset verticale, di range verticale, di attenuazione e selezionare la modalità tra AC e DC. Inoltre è possibile regolare la base dei tempi e impostare, o meno, la modalità di acquisizione continua e un eventuale trigger.



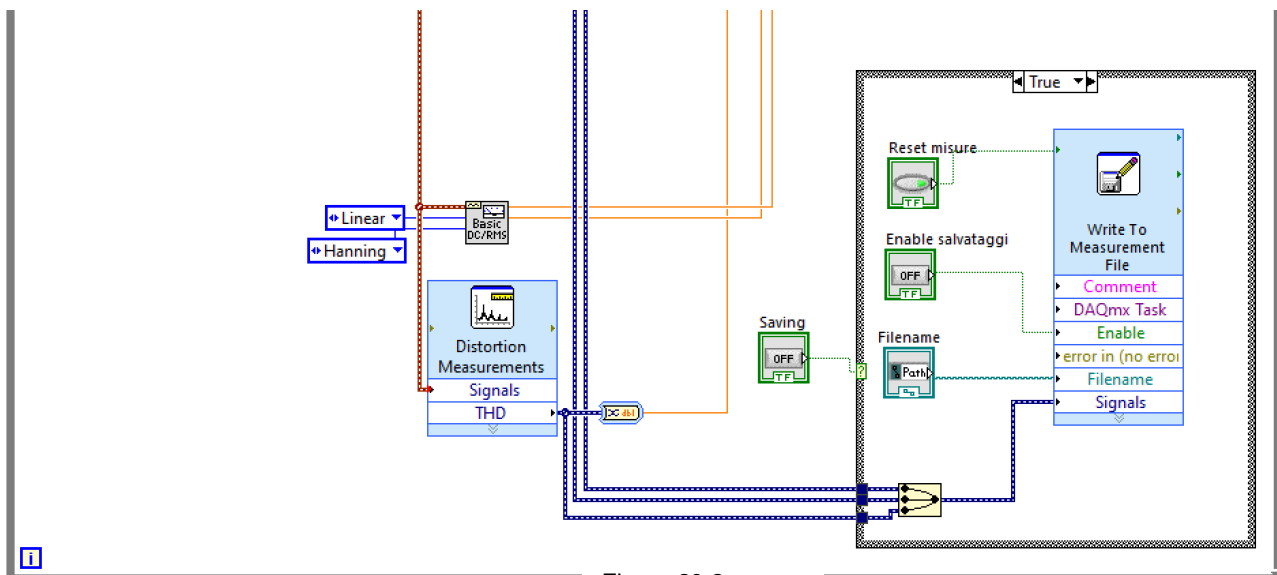


Figura 20.2

Il rilevamento della forma d'onda può essere fatto anche solo in un range selezionato dall'interfaccia in modo da ridurre l'intervallo di dati analizzato. Tramite un waveformData è possibile visualizzare l'andamento dell'onda generata. Eventualmente, a seguito della pressione del pulsante di salvataggio, i dati possono essere salvati in un file grazie al componente "Write To Measurement File".

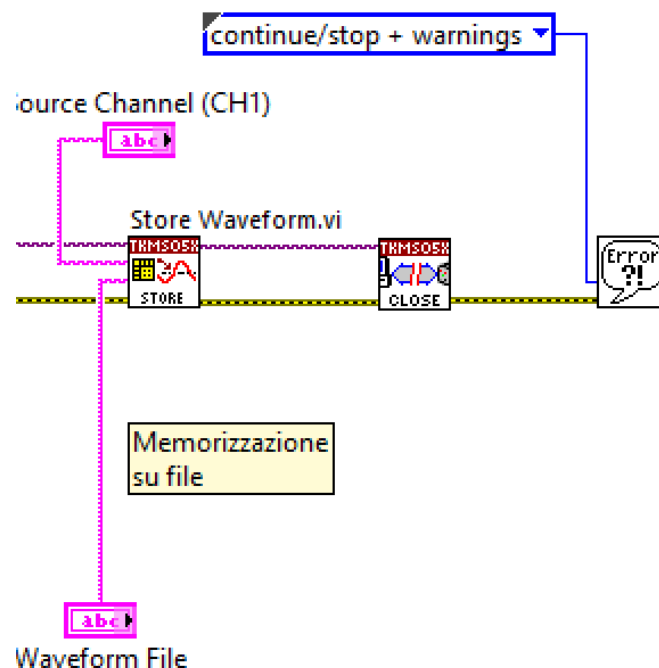


Figura 20.3

Nella parte finale del codice viene memorizzata la forma d'onda e viene chiusa la comunicazione lanciando eventuali messaggi di errore.

## COMPONENTI VISA UTILIZZATI

Nell'oscilloscopio è stato usato un Visa Resource Name per poter leggere dallo strumento.

VISA Resource Name



Figura 21

## COMPONENTI DRIVER TEKTRONIX UTILIZZATI

Di seguito sono riportati i principali componenti Tektronix utilizzati:



Figura 22

Inizializza il canale di comunicazione dello strumento. Questo VI può anche eseguire una richiesta di identificazione e un'operazione di reset e può svolgere tutte le operazioni necessarie per porre lo strumento nel suo stato acceso di default o un altro stato specificato.



Figura 23

Componente che permette la configurazione del canale con la relativa impostazione dei parametri, tra cui il range verticale e l'attenuazione.



Figura 24

Componente che permette la configurazione della base dei tempi dell'oscilloscopio.

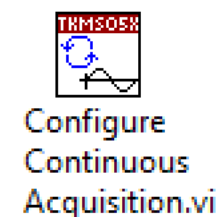


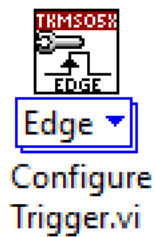
Figura 25

Componente che permette, tramite un pulsante collegato, l'attivazione della modalità di acquisizione continua.



Figura 26

Permette la lettura della forma d'onda da un determinato canale. In questo progetto è stato settato un tempo massimo di 10000ms per non avere tempi eccessivi, ma può essere modificato. Inoltre possono essere impostati i punti limiti "starting point" e "ending point".



Componente che permette di effettuare la configurazione del trigger. In questo progetto è stata settata in “modalità edge” ma sono possibili altre modalità riportate a lato. Inoltre è possibile settare la pendenza (rise/fall) tramite l'apposito selettore.

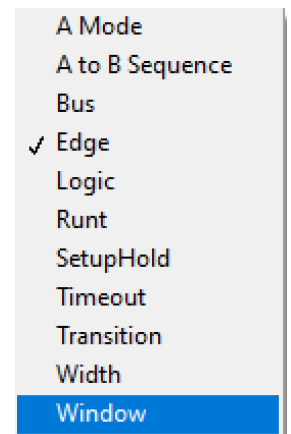


Figura 27

Figura 28



Termina la comunicazione con il canale dello strumento.

Figura 29

## PULSANTI UTILIZZATI



Figura 30

Questo pulsante, collegato al terminale di condizione del while loop, è stato utilizzato per stoppare l'andamento dell'onda in modo da poterla analizzare più facilmente.



Figura 31

Si tratta di un pulsante con due modalità (on/off) usato per attivare il salvataggio delle misure su file.

## COMPONENTI DI ANALISI DEL SEGNALE UTILIZZATI

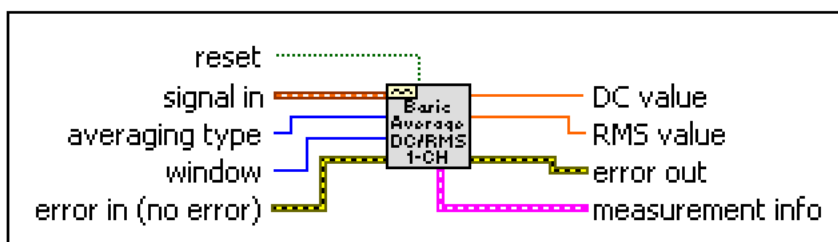


Figura 32

Calcola i valori CC e RMS di una forma d'onda di ingresso o di una matrice di forme d'onda.

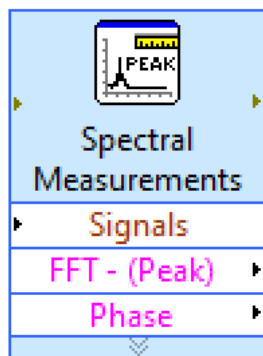


Figura 33

Esegue misurazioni spettrali basate su FFT, come lo spettro di magnitudine media, lo spettro di potenza e lo spettro di fase su un segnale.

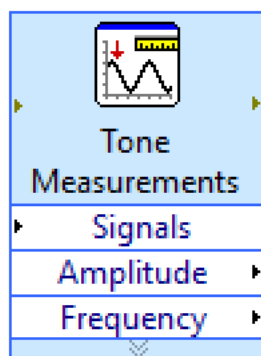


Figura 34

Trova il tono singolo con l'ampiezza più alta o ricerca un intervallo di frequenza specificato per trovare il tono singolo con l'ampiezza più alta. Serve anche trovare la frequenza e la fase per un singolo tono.

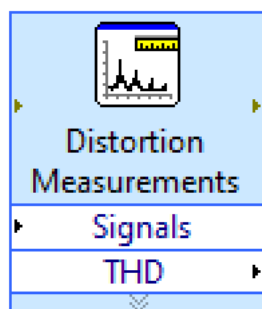


Figura 35

Esegue misurazioni della distorsione su un segnale, come l'analisi del tono, la distorsione armonica totale (THD) e il segnale in rumore e distorsione (SINAD).



## COMPONENTI PER LA SCRITTURA SU FILE UTILIZZATI

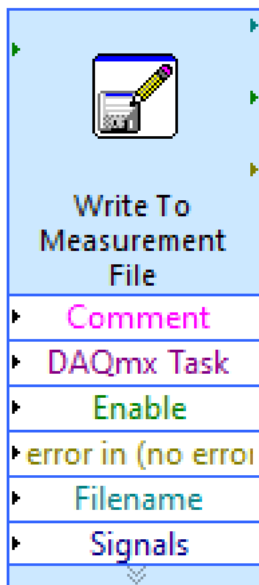


Figura 36

Scrive i dati in file di misurazione basati su testo (.lvm), file di misurazione binari (.tdm o .tdms) o file di Microsoft Excel (.xlsx). In questo caso è stato scelto il salvataggio in formato lvm per comodità (visibile un esempio nella figura 39).

## STRUTTURE UTILIZZATE

Le strutture utilizzate sono un While Loop (figura 37) per la gestione del ciclo che circonda l'intera parte centrale del codice e una Case Structure (figura 38) per la gestione della scrittura su file in caso di pressione del pulsante di salvataggio.

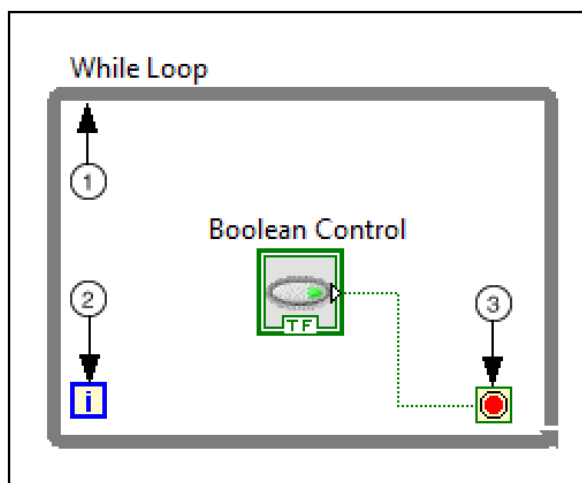


Figura 37

- ① **Sottodiagramma** — Contiene il codice che il while loop esegue una volta per ogni iterazione.
- ② **Terminale di iterazione** — Fornisce il conteggio delle iterazioni del ciclo corrente. Il conteggio del ciclo inizia sempre da zero per la prima iterazione.
- ③ **Terminale di condizione** — valuta un valore di input booleano per determinare se continuare a eseguire il ciclo While. È possibile condizionare la continuazione del ciclo con un pulsante, nel caso del generatore un pulsante di stop.

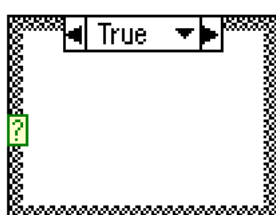
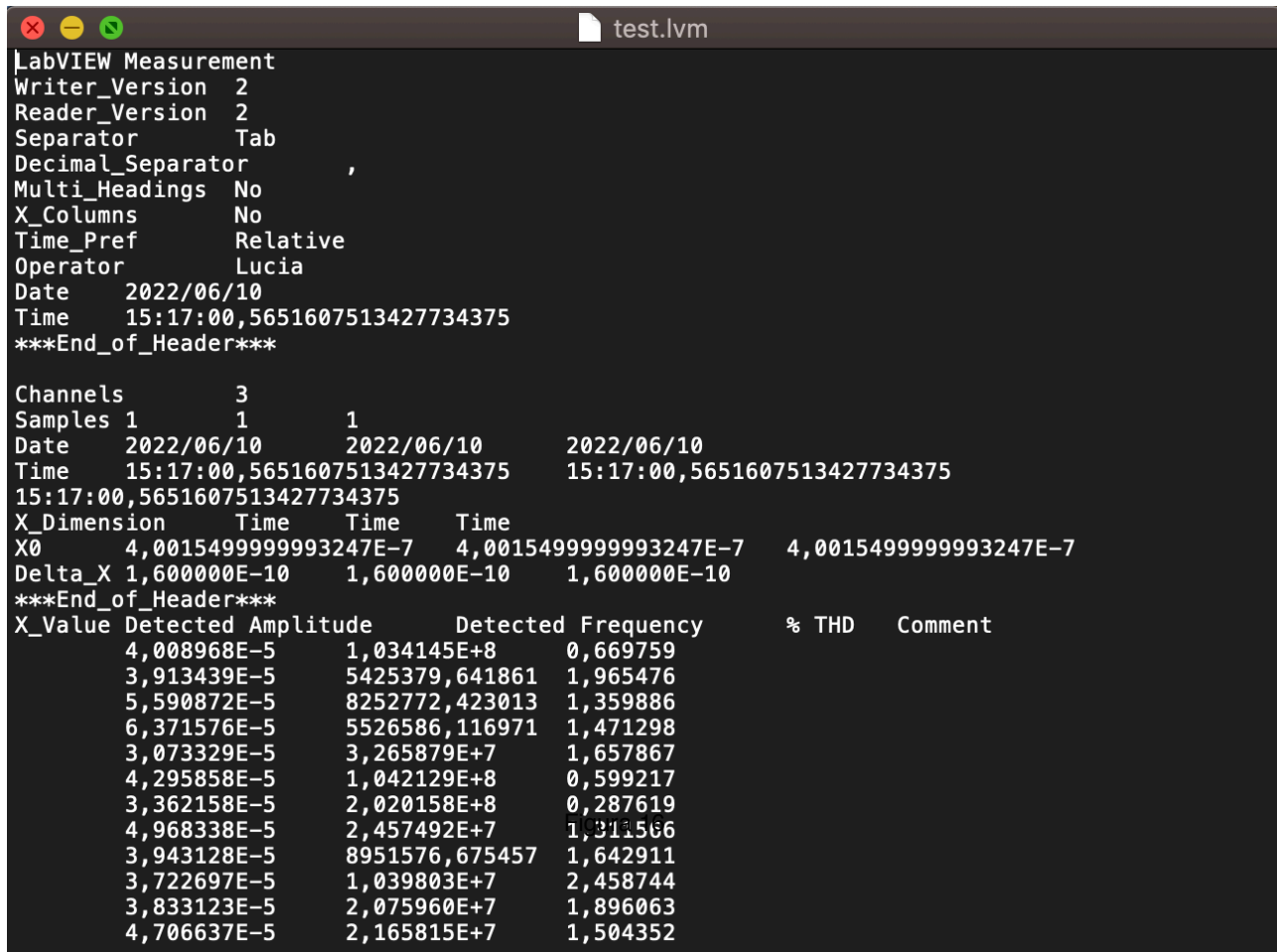


Figura 38

**CASE STRUCTURE** — Contiene uno o più sottodiagrammi, o casi, esattamente uno dei quali viene eseguito quando viene eseguita la struttura. Il valore collegato al selettore caso determina quale caso eseguire. Tramite il label di selezione in alto è possibile visualizzare i valori per i quali viene eseguito il caso associato.

## ESEMPIO DI OUTPUT GENERATO DA UNA SCRITTURA SU FILE

Nella figura seguente viene riportato un esempio di risultato di scrittura su file in cui si possono vedere le ampiezze e le frequenze misurate.



```
LabVIEW Measurement
Writer_Version 2
Reader_Version 2
Separator Tab
Decimal_Separator ,
Multi_Headings No
X_Columns No
Time_Pref Relative
Operator Lucia
Date 2022/06/10
Time 15:17:00,5651607513427734375
***End_of_Header***

Channels 3
Samples 1 1 1
Date 2022/06/10 2022/06/10 2022/06/10
Time 15:17:00,5651607513427734375 15:17:00,5651607513427734375
15:17:00,5651607513427734375
X_Dimension Time Time Time
X0 4,0015499999993247E-7 4,0015499999993247E-7 4,0015499999993247E-7
Delta_X 1,600000E-10 1,600000E-10 1,600000E-10
***End_of_Header***
X_Value Detected Amplitude Detected Frequency % THD Comment
4,008968E-5 1,034145E+8 0,669759
3,913439E-5 5425379,641861 1,965476
5,590872E-5 8252772,423013 1,359886
6,371576E-5 5526586,116971 1,471298
3,073329E-5 3,265879E+7 1,657867
4,295858E-5 1,042129E+8 0,599217
3,362158E-5 2,020158E+8 0,287619
4,968338E-5 2,457492E+7 1,311566
3,943128E-5 8951576,675457 1,642911
3,722697E-5 1,039803E+7 2,458744
3,833123E-5 2,075960E+7 1,896063
4,706637E-5 2,165815E+7 1,504352
```

Figura 39

## CONSIDERAZIONI FINALI

Durante la realizzazione di questo progetto di tirocinio ho imparato ad utilizzare diverse tecnologie che non conoscevo e che non avevo mai approfondito molto.

In primo luogo ho imparato ad utilizzare Labview, uno strumento che non conoscevo ma che è risultato essere molto interessante perché offre una moltitudine di funzioni e driver aggiuntivi per poter implementare diverse funzionalità.

Inoltre, mi sono interfacciata con il generatore di funzioni e l'oscilloscopio. Questi strumenti li avevo studiati solo in minima parte nei corsi precedenti e alle superiori, ma non li avevo mai utilizzati per "lunghi" periodi di tempo.

## BIBLIOGRAFIA

- Per scaricare LabView e i vari driver e componenti aggiuntivi necessari per la realizzazione del progetto:

<https://www.ni.com/it-it/support/downloads/software-products/download.labview.html#460283>  
<https://www.ni.com/it-it/support/downloads.html>

- Manuali utilizzati per imparare ad usare LabView:

["LabVIEW – Corso Base I"](#), edizione ottobre 2000

"Introduction to LabVIEW – Six Hour Course", edizione settembre 2003

- Corsi online usati per imparare ad usare LabView:

[https://www.halvorsen.blog/documents/teaching/courses/labview\\_automation/labview\\_basic.php](https://www.halvorsen.blog/documents/teaching/courses/labview_automation/labview_basic.php)  
<https://www.halvorsen.blog/documents/tutorials/resources/Introduction%20to%20LabVIEW.pdf>