

Università Degli Studi Di Cassino e Del Lazio Meridionale

Dipartimento DIEI
Area Di Ingegneria



Corso Di Laurea Magistrale in "Ingegneria Informatica"

Relazione progetto del corso di Sistemi Di Misura Distribuiti

"Sistema di misura distribuito per misure di frequenza ed intervalli di tempo attraverso contatore digitale"

Gruppo 2:

- Capodiferro Francesco
- Di Carlo Luca
- Fierimonte Marco
- Marabese Emilia

Sommario

INTRODUZIONE	3
STRUMENTAZIONE UTILIZZATA	4
CLIENT: MANUALE UTENTE	6
CLIENT: MANUALE PROGRAMMATORE	10
SERVER: MANUALE PROGRAMMATORE	16

INTRODUZIONE

La traccia del progetto richiedeva la realizzazione di un sistema di misura distribuito che consentisse il controllo da remoto di un generatore di forme d'onda e di un contatore digitale. Attraverso questi strumenti si doveva consentire la misurazione di: frequenza, periodo, tempo di salita, tempo di discesa ed intervallo di tempo tra due eventi (indicato con T_{a-b}).

Il progetto è stato realizzato interamente durante le ore di lezione, nel LAMI (Laboratorio Misure Industriali). Sono stati utilizzati un generatore di forme d'onda ed un contatore digitale collegati in parallelo tra loro e collegati ad un pc tramite un'interfaccia IEEE488.

Per mezzo dei software Labview 2012 e NI MAX si è riusciti ad implementare una connessione TCP/IP tra un client ed un server che realizzasse la comunicazione tra i due strumenti e l'utente. Il sistema era costituito da due parti: un lato client ed un lato server.

Il lato client prevedeva un'interfaccia user-friendly che consentisse in maniera intuitiva all'utente di pilotare i due strumenti da remoto, con la possibilità di variare alcuni parametri.

Il lato server prevedeva, invece, un'interfaccia minimale volta soltanto ad avere un controllo sullo stato della trasmissione ed era completamente priva di elementi interattivi.

Inoltre, l'idea implementata era quella di discriminare due categorie di utenti: un utente "autorizzato", ovvero in possesso di password, il quale aveva pieno accesso alle funzionalità del programma, ed un utente "guest", il quale aveva accesso esclusivamente alla sezione di misurazione del programma e non a quella di controllo ed impostazione della forma d'onda.

Si è resa necessaria l'implementazione di un partitore di tensione come "rete di sicurezza" volta a proteggere il contatore da errori nell'impostazione della forma d'onda, riducendo il rischio di rottura di quest'ultimo. Questa rete è stata frapposta tra il generatore ed il contatore che quindi andava a misurare la tensione ai capi di una delle due resistenze costituenti il partitore.

STRUMENTAZIONE UTILIZZATA

Gli strumenti a disposizione del nostro gruppo per la realizzazione del progetto erano:

- GENERATORE DI FORME D'ONDA
 - Modello: Agilent 33120A
 - Massima Ampiezza: 10 VPP
 - Massime Frequenze:
 - Seno: 15 MHz
 - Onda Quadra: 15 MHz
 - Onda Triangolare: 100 KHz
 - Interfacce Di Comunicazione: GPIB 488 e RS232
- CONTATORE DIGITALE A DUE CANALI
 - Modello: Agilent 53132A
- RESISTENZE
- CONNETTORI BNC
- SCHEDA DI COMUNICAZIONE
 - Modello: National Instruments GPIB-USB-HS
- LABVIEW 2012 STUDENT EDITION
- NI MAX
- PERSONAL COMPUTERS

Di seguito è mostrata la strumentazione a disposizione.



Figura 1 Generatore forme d'onda e contatore digitale

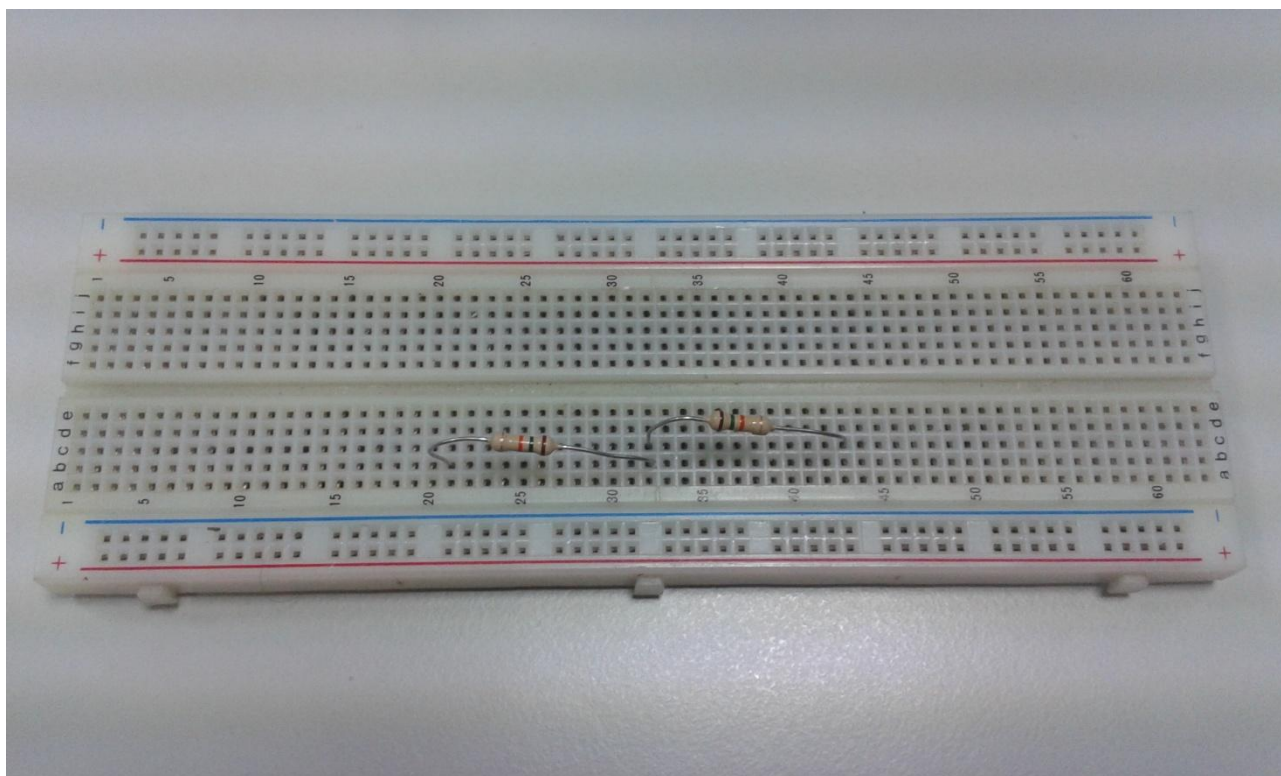


Figura 2 Partitore di tensione



Figura 3 Controller GPIB-USB-HS

CLIENT: MANUALE UTENTE

Il client mette a disposizione un'interfaccia utente che consente di settare diverse tipologie di parametri. In particolar modo essa consente il settaggio della forma d'onda erogata dal generatore, per mezzo dei parametri: *waveform*, *amplitude*, *frequency* e *offset*, e la scelta del tipo di misurazione da effettuare, per mezzo dei parametri: *measurement*, *trigger level* e *slope*. Inoltre da questo pannello è possibile inserire la password per accedere come utente "autorizzato".

Sull'interfaccia utente è possibile individuare due tipologie di elementi: *controlli* e *indicatori*. I controlli sono dei campi interattivi attraverso i quali l'utente può impostare le proprie scelte. Gli indicatori invece sono dei visualizzatori che consentono all'utente di visualizzare eventuali risposte del programma.

Sono presenti 3 diversi settori: uno per il Login, uno per l'impostazione del generatore ed un altro per la gestione del contatore. Di seguito verranno presentate nel dettaglio.

LOGIN PAGE

The screenshot shows a web interface titled "CONTROL PANEL" in large white letters on a blue background. Below the title are three tabs: "LOGIN PAGE", "GENERATOR SET", and "COUNTER SET". The "LOGIN PAGE" tab is active. Inside the tab, there is a light blue box containing several input fields and labels: "CLIENT PORT NUMBER" with a dropdown menu showing "0", "INSERT PASSWORD" with a text input field, "PASSWORD STATUS" with a green indicator light, and "CHECK MESSAGE" with a text input field. At the bottom of the interface, there is a cyan box with the text: "INFO: Inserire i dati richiesti, impostare il generatore, il contatore ed infine premere START MEASURE."

In quest'area è possibile loggarsi come utente autorizzato ed è possibile impostare il numero di porta del server con il quale si vuole comunicare.

- **Client Port Number:** qui va inserito il numero di porta del server. E' un parametro necessario per instaurare la comunicazione TCP/IP.
- **Insert Password:** questo controllo serve per inserire la password, in modo tale da potersi loggare come utenti "autorizzati". Se la password sarà assente o errata l'utente verrà abilitato come "guest" e potrà soltanto scegliere il tipo di misurazione da effettuare.
- **Password status:** è un indicatore che rappresenta lo stato della password. Se il LED verde è acceso, vuol dire che la verifica della password ha avuto esito positivo e quindi l'utente è riconosciuto come "autorizzato", altrimenti nel campo **message** verrà visualizzato un messaggio di errore.

GENERATOR SET

The screenshot displays a web interface titled "CONTROL PANEL" with a blue background. At the top, there are three tabs: "LOGIN PAGE", "GENERATOR SET" (which is active), and "COUNTER SET". Below the tabs, the "GENERATOR SET" section contains a "WAVEFORM" dropdown menu and three input fields labeled "AMPLITUDE", "FREQUENCY", and "OFFSET". Each of these three fields has a small circular icon with a triangle inside, positioned to the left of the input box. The "AMPLITUDE" and "FREQUENCY" fields currently show the value "0". At the bottom of the panel, there is an "INFO" box with the text: "Scegliere il tipo di forma d'onda, l'ampiezza, la frequenza e l'offset."

In questo settore è possibile settare i parametri della forma d'onda che poi verrà inviata al circuito sul quale viene misurata la tensione.

- **Waveform:** questo menù a tendina consente di scegliere una tra le diverse forme d'onda a disposizione (sinusoide, onda quadra ed onda triangolare). Nel caso in cui non venisse specificato il tipo di forma d'onda, verrà impostata di default la sinusoide.
- **Amplitude:** in questo controllo va specificata l'ampiezza della forma d'onda desiderata.
- **Frequency:** in questo controllo va specificata la frequenza della forma d'onda desiderata (n.b. non è possibile utilizzare formati alfanumerici, ad es. 10k).
- **Offset:** in questo controllo è possibile specificare l'offset iniziale della forma d'onda desiderata.

COUNTER SET

The image shows a web-based control panel titled "CONTROL PANEL" in large white letters on a blue background. Below the title are three tabs: "LOGIN PAGE", "GENERATOR SET", and "COUNTER SET", with the latter being the active tab. The interface is divided into several sections:

- TRIGGER LEVEL** and **TRIGGER SLOPE**: Each has a text input field and a dropdown arrow.
- MEASUREMENT**: A text input field with a dropdown arrow.
- START MEASURE**: A text input field.
- END MEASURE**: A text input field.
- IF YOU HAVE CHOICED Ta->b**: A section header for conditional settings.
- TRIGGER LEVEL EVEN B**: A text input field.
- TRIGGER SLOPE EVEB B**: A text input field with a dropdown arrow.
- START MEASURE**: A grey button.
- MEASURED VALUE**: A text input field.
- INFO:** A cyan box at the bottom containing the text "Inserire i dati richiesti."

In questo settore è possibile impostare i parametri del contatore, avviare la misurazione e leggerne il risultato. Si noti che c'è una sezione opzionale che deve essere impostata solo nel caso in cui si scelga come misura da effettuare T_{a-b} .

Per avviare la misurazione è necessario premere il pulsante *Start Measure*, dopo aver settato tutti i parametri.

- **Trigger Level:** questo controllo consente l'impostazione del livello di trigger del segnale da misurare.
- **Trigger Slope:** questo controllo consente la scelta della pendenza con cui il segnale deve attraversare il livello di trigger, affinché questo risulti verificato.
- **Measurement:** questo menu a tendina consente la scelta di una tra le possibili misurazioni implementate.
- **Start Measure:** questo indicatore segnala il corretto avvio della misurazione per mezzo di un messaggio che riceve dal server.
- **End Measure:** questo indicatore segnala la corretta terminazione della misurazione per mezzo di un messaggio che riceve dal server.
- **Measured Value:** questo indicatore visualizza il risultato della misurazione inviatogli dal server.

Nel caso in cui si è scelta come misurazione T_{a-b} :

- **Trigger Level Even B:** questo controllo consente l'impostazione del secondo livello di trigger del segnale da misurare.
- **Trigger Slope Even B:** questo controllo consente la scelta della pendenza con cui il segnale deve attraversare il secondo livello di trigger, affinché questo risulti verificato.

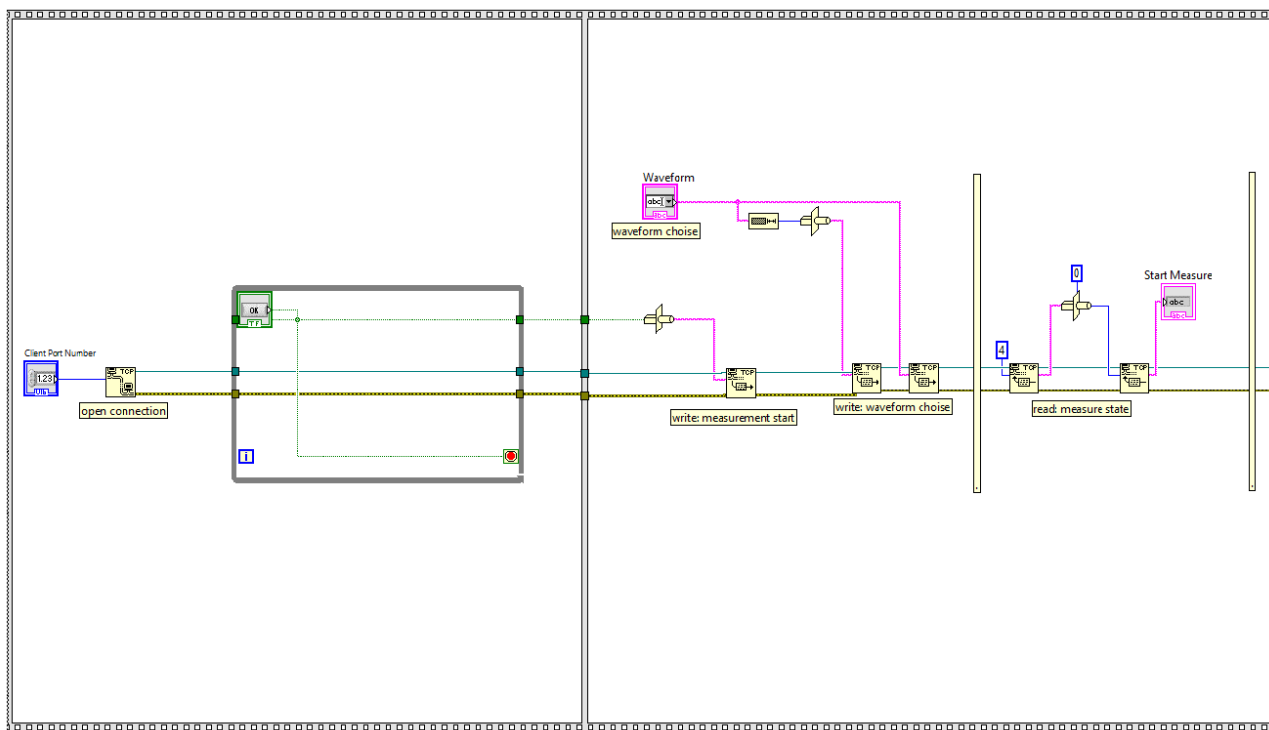
Una volta eseguito, il programma è in grado di effettuare una sola misurazione, dopodiché termina. Dunque nel caso si vogliano effettuare più misurazioni è necessario lanciare più volte il programma.

CLIENT: MANUALE PROGRAMMATTORE

Questa sezione si pone l'obiettivo di illustrare, in maniera quanto più chiara ed immediata possibile, la struttura del programma realizzato. L'ambiente di sviluppo utilizzato, come già detto, è il LabVIEW 2012. Esso consente la realizzazione di file nel formato VI (Virtual Instruments). Queste VI sono composte da due macrosezioni: un *front panel*, che è la parte di interfacciamento del programma ed è quella vista nel manuale utente, ed un *block diagram*, che rappresenta la sezione di codice (sebbene grafico) realizzato dal programmatore.

Ora si vedrà nel dettaglio il block diagram. Vista la sua complessità e lunghezza, è stato suddiviso in diverse parti e queste verranno discusse singolarmente.

Ricordiamo la funzionalità implementata dal lato client: ricevere password e verificare le autorizzazioni per l'utente, ricevere in ingresso i parametri della forma d'onda da generare e le impostazioni del contatore digitale ed inviarle al server, e ricevere dallo stesso il risultato della misurazione.



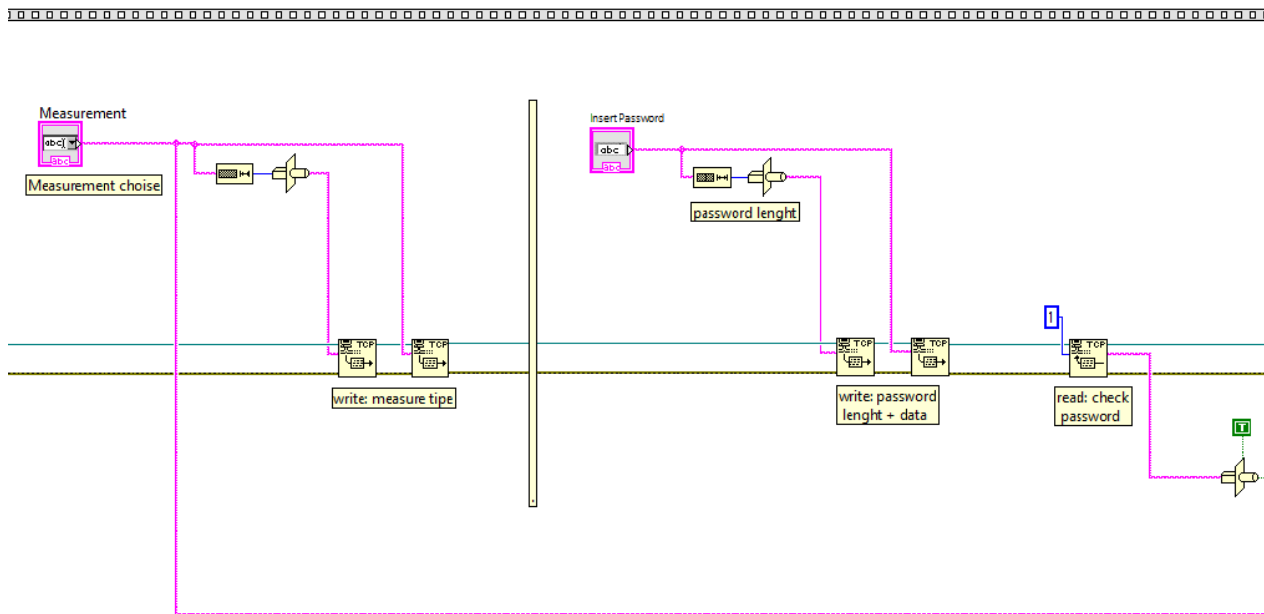
Inizialmente si è provveduto ad aprire una connessione TCP/IP tra il client ed il server. A tal proposito si è reso necessario inserire un controllo con il quale settare il numero di porta. La *while structure* successiva contiene al suo interno il pulsante di *Start measure* ed ha lo scopo di temporizzare l'esecuzione del client: una volta avviato, infatti, sarà possibile settare tutti i parametri ma finché non verrà premuto il tasto di start il programma rimarrà nel while in questione. Il controllo tramite pulsante infatti associa a quest'ultimo un valore booleano che è false finché non

viene premuto, dopodiché commuta il suo valore booleano sul *true*. La condizione presente nel ciclo infatti è del tipo *Stop if true*, il che implica che finché il valore della variabile booleana associata al pulsante rimarrà *false*, il programma stazionerà nel *while*, quando invece passerà a *true*, esso uscirà dal ciclo e proseguirà la sua esecuzione. Da notare la presenza di una *flat sequence*: questa serve a temporizzare l'esecuzione delle istruzioni. Infatti senza di questa poteva accadere che, sebbene i valori scelti venissero inviati al server solo dopo aver premuto il tasto di avvio, essi venissero letti direttamente all'avvio del programma. In tal modo venivano resi inutili i settaggi degli strumenti fatti dopo l'avvio del programma stesso. Con questa *flat*, invece, il programma andrà a leggere i valori contenuti negli indicatori solo dopo la pressione del tasto di start. La *flat sequence* implementata ha solo due frame, una contenente la pressione del tasto e una che contiene tutto il resto del programma.

Dopo l'avvio della misurazione è presente un *TCP/IP write* necessario per inviare al server la comunicazione di avvio della misurazione.

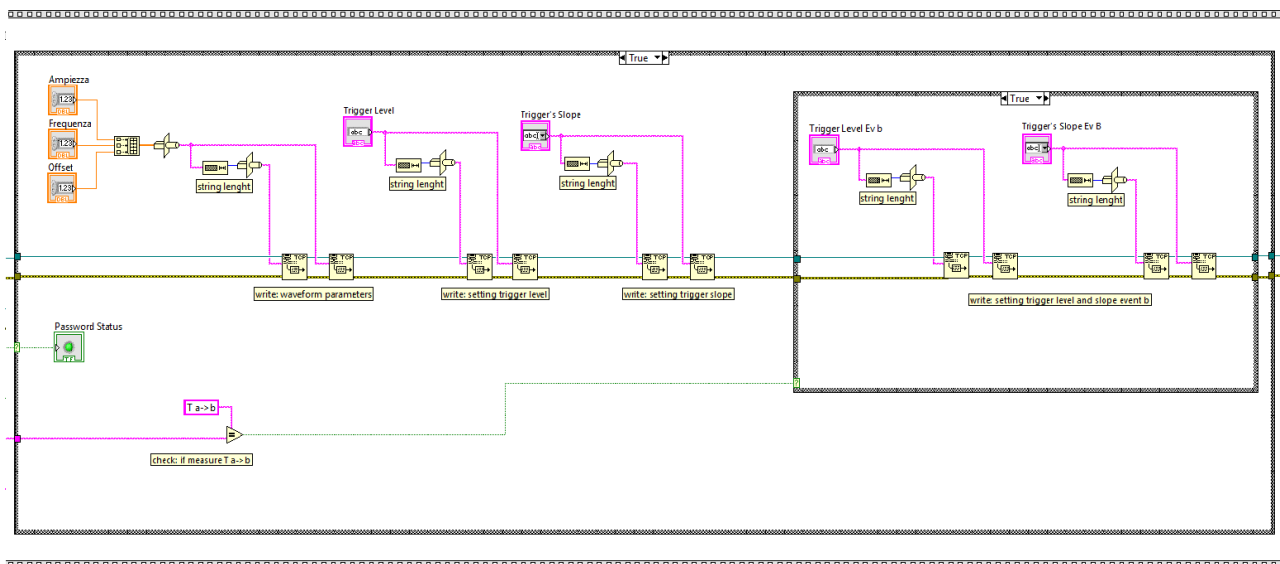
Lo step successivo è la scelta della forma d'onda. Tramite il controllo *waveform* viene letta la forma d'onda selezionata ed il valore viene inviato al server tramite un'operazione di *TCP/IP write*. La forma d'onda selezionata è inviata in formato stringa e pertanto si è reso necessario specificare al server il numero di caratteri da leggere. Dunque si è utilizzato un secondo *write*, anteposto a quello della forma d'onda, che indicasse al server quanti bytes avrebbe dovuto leggere nell'operazione successiva. A tal proposito si è utilizzato il blocco di *string lenght* che, ricevendo una stringa in ingresso, ne restituisce la lunghezza effettiva. Tuttavia si è reso necessario convertire questo valore numerico (intero) in formato stringa: questo perché nella comunicazione (read - write) *TCP/IP* è possibile inviare solo questo formato di dati. Per effettuare questa conversione si è utilizzato un blocco di *type cast*.

Successivamente è presente un'operazione di *TCP/IP read* che riceve dal server un messaggio di corretto avvio della comunicazione. Questa richiede, dualmente a come accadeva in precedenza per la scrittura, due operazioni di *read*: una per ricevere la lunghezza del messaggio da leggere (dovendo leggere un intero sappiamo già che sono sufficienti 4 byte) ed un'altra per leggere l'intero messaggio, alla quale va dato come parametro il risultato della precedente.

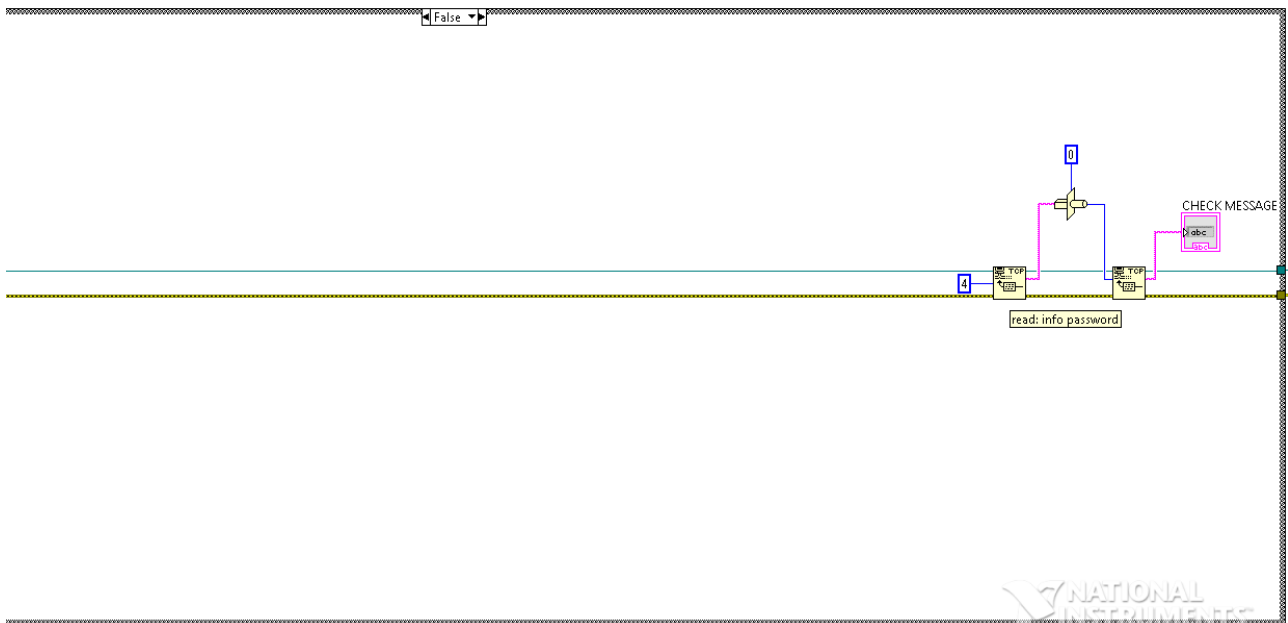


Si procede con la scelta del tipo di misurazione e con l'inserimento della password, che avvengono in maniera del tutto analoga a come avveniva la scelta del tipo di forma d'onda. Si noti che, benché siano due operazioni diverse, presentano la medesima struttura e quindi gli stessi elementi per implementarle. Graficamente l'unica differenza nella loro realizzazione sta nel fatto che per scegliere il tipo di misurazione (così come il tipo di forma d'onda) si è utilizzato un menù ring, ovvero un controllo che presenta un menu a tendina con opzioni definite dall'utente, mentre per la password si è utilizzato un semplice controllo testuale, che consente di inserire una qualunque stringa.

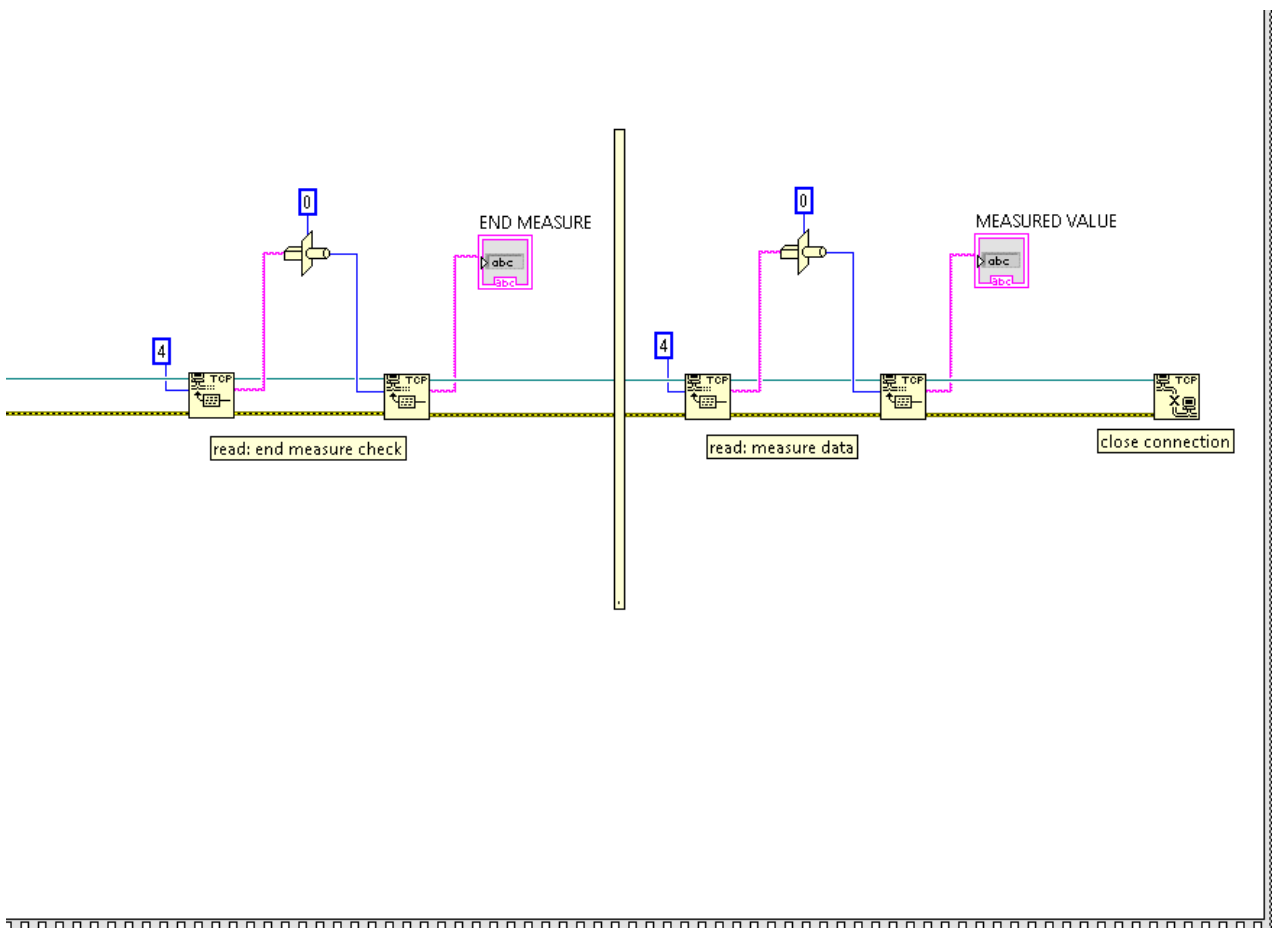
Una volta inviata la password al server, questo restituirà un valore booleano che conterrà l'esito della verifica della stessa. Se la password risulterà verificata, questo sarà true, altrimenti false. Solo che il TCP/IP prevede esclusivamente il passaggio di stringhe dunque bisogna riconvertire tale valore in booleano e lo si fa tramite il type cast.



La sezione vista sin'ora era comune sia ad un utente "autorizzato", sia ad un utente "guest". Ora dopo il check password si discriminano due casi: se un utente è autorizzato (dunque password=true) esso potrà modificare anche i parametri della forma d'onda e del contatore, altrimenti potrà solo misurare dei valori di default. Allora per implementare quest'idea si utilizza una *case structure*, la quale prevede due diversi frame, ovvero due diversi andamenti del programma, a seconda del valore della variabile booleana che funge da variabile di controllo. Nel caso true sono presenti 3 controlli booleani attraverso i quali è possibile inserire i valori di ampiezza, frequenza ed offset della forma d'onda. Con questi viene costruito un array il quale viene poi convertito in stringa per poter essere inviato al server. Si è scelto di accorpare i 3 valori in un unico array per limitare il numero di read (ed i corrispondenti write). Con le stesse tecniche viste fin'ora si procede con l'inserimento del livello di trigger e della pendenza del livello di trigger. Se l'utente seleziona la misura T_{a-b} è necessario inserire un secondo livello di trigger con la rispettiva pendenza, per individuare l'evento b. Questa situazione viene raccolta in una case structure in cui la condizione di verifica è data dall'esito del confronto tra la misura scelta ed una stringa costante contenente Ta-b, mediante l'operatore di uguaglianza.



Nel caso false invece, come si può vedere, la parte di setting dei parametri è sostituita da un'unica operazione di read, la quale serve a ricevere dal server un messaggio di errore nella verifica della password.



Al termine di entrambi i casi il server invia un messaggio al client per comunicare la chiusura dell'operazione di misura, e questo avviene mediante una read.

Infine è presente un ultimo read che riceve dal server il risultato della misurazione e lo pone nell'indicatore MEASURED VALUE, visibile sul front panel.

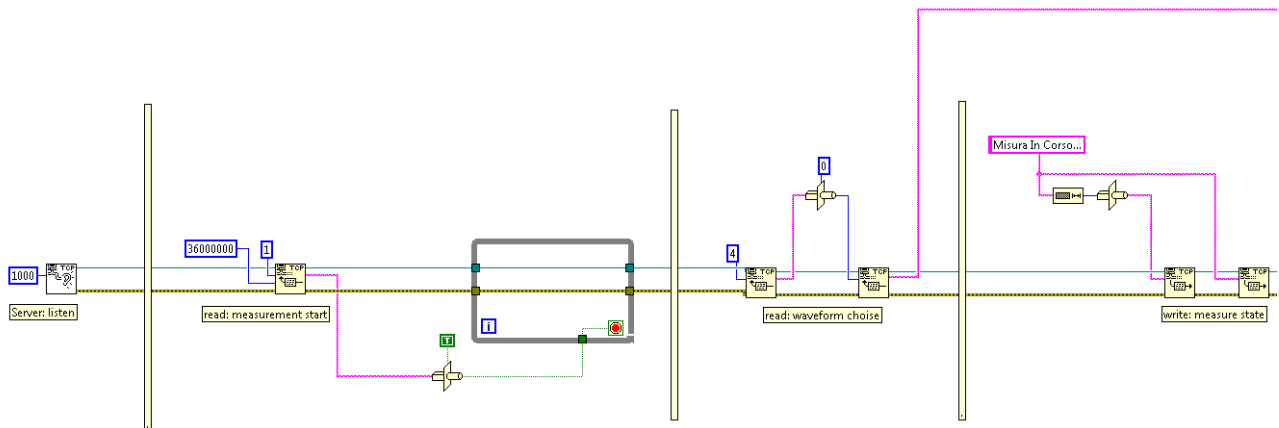
Al termine delle operazione è presente il blocco *close connection* che chiude la connessione TCP/IP con il server.

SERVER: MANUALE PROGRAMMATTORE

Il server presenta un front panel privo di controllori e indicatori.

In questa sezione si va a descrivere nel dettaglio il block diagram del lato server.

Anche in questo caso, vista la lunghezza complessiva, il codice è stato suddiviso in sezioni le quali verranno discusse singolarmente.



NATIONAL
INSTRUMENTS

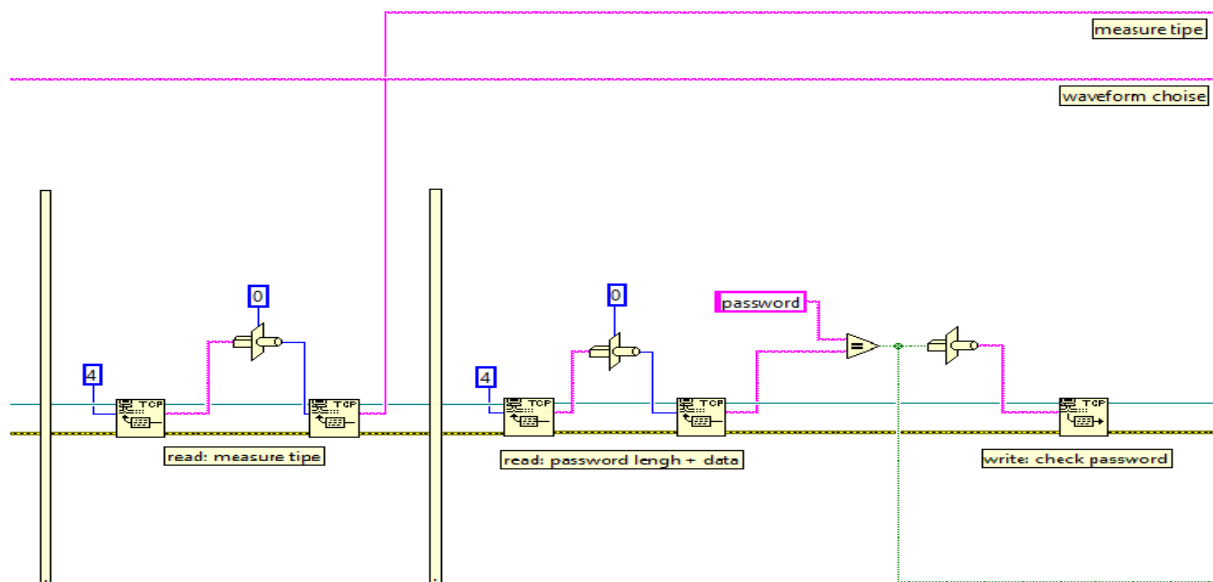
La chiamata dal client avviene mediante un blocco *TCP/IP listen*, il quale crea un ascolto e un'attesa per la connessione TCP. Al server è stato assegnato il numero di porta 1000, che dovrà essere specificato nel client per stabilire la connessione.

Una volta aperta la connessione il server attende che l'utente prema il pulsante di *Start Measurement*. Per fare questo si utilizza un ciclo *While* dal quale si esce, proseguendo l'esecuzione del programma, solo quando la costante booleana in ingresso diventa *False*.

Dopo l'avvio della misurazione il server legge il tipo di forma d'onda, da generare attraverso il generatore connesso tramite porta seriale RS-232, attraverso il blocco *Read TCP/IP*.

Il read riceve in ingresso una stringa dal client. Dal momento che il read è relativo ad un'operazione di write del client, deve presentarne la stessa struttura: un primo read va a leggere il numero di bit da leggere ed il secondo bit legge il messaggio e come parametro *bytes to read* sfrutta il risultato della read precedente.

Il server invia un messaggio di misurazione in corso, che viene visualizzato nel Front Panel del client. Questo viene fatto tramite il blocco di *Write TCP/IP*, con le stesse modalità con cui venivano effettuati i write nel client.

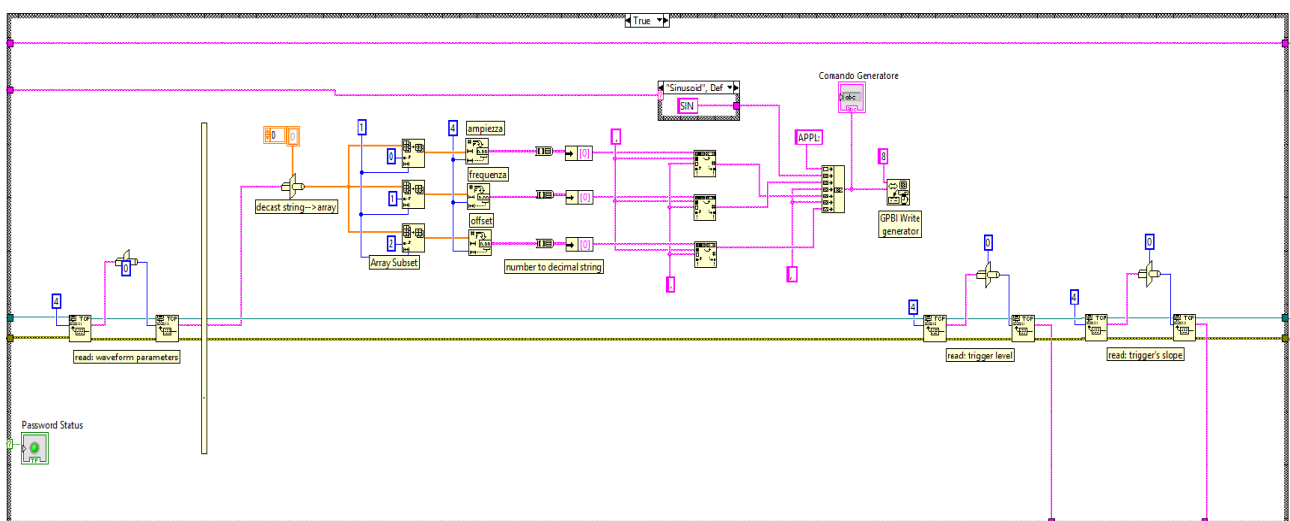


Dopo l'avvio della misura il server riceve il tipo di misurazione da effettuare, sempre tramite operazioni di read.

Successivamente viene letta e controllata la password inserita dall'utente. In base all'esito di questo confronto si discrimina l'andamento del programma per un utente "autorizzato" e per un utente "guest".

L'uscita del blocco di confronto è un valore booleano che abilita uno dei due case mostrati nelle immagini successive.

Iniziamo con il caso true:



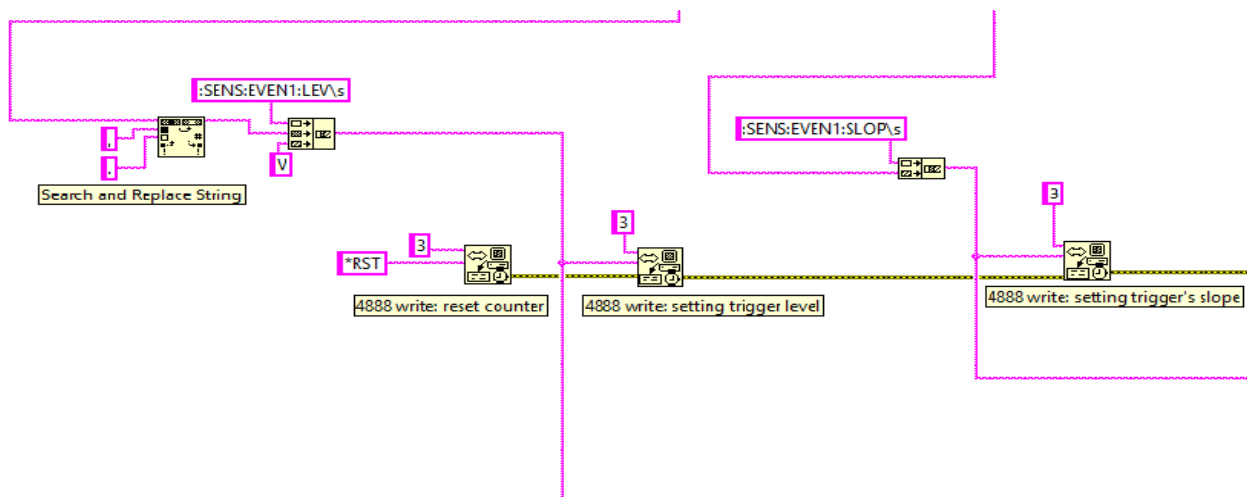
Vengono letti tutti i parametri inseriti dall'utente, ovvero ampiezza, frequenza e offset. Dato che i valori vengono inviati in un array, viene usato il blocco *Array Subset*. In questo modo vengono estratte le righe rappresentati un dato, le quali vengono selezionate per mezzo di una costante, e convertite in stringhe che poi andranno a costituire parte del comando inviato al generatore.

Una volta ottenute le stringhe contenenti il dato, si è resa necessaria una loro elaborazione: infatti poteva accadere che i valori numerici fossero immessi con la virgola come separatore per le cifre decimali. Questa eventualità non è supportata dal generatore che considera il punto come carattere di separazione tra le cifre decimali e la virgola come separatore fra i vari caratteri. Allora si è ricercata la virgola all'interno della stringa e la si è sostituita con il punto. Questa operazione è stata interamente implementata mediante il blocco *Search and replace*.

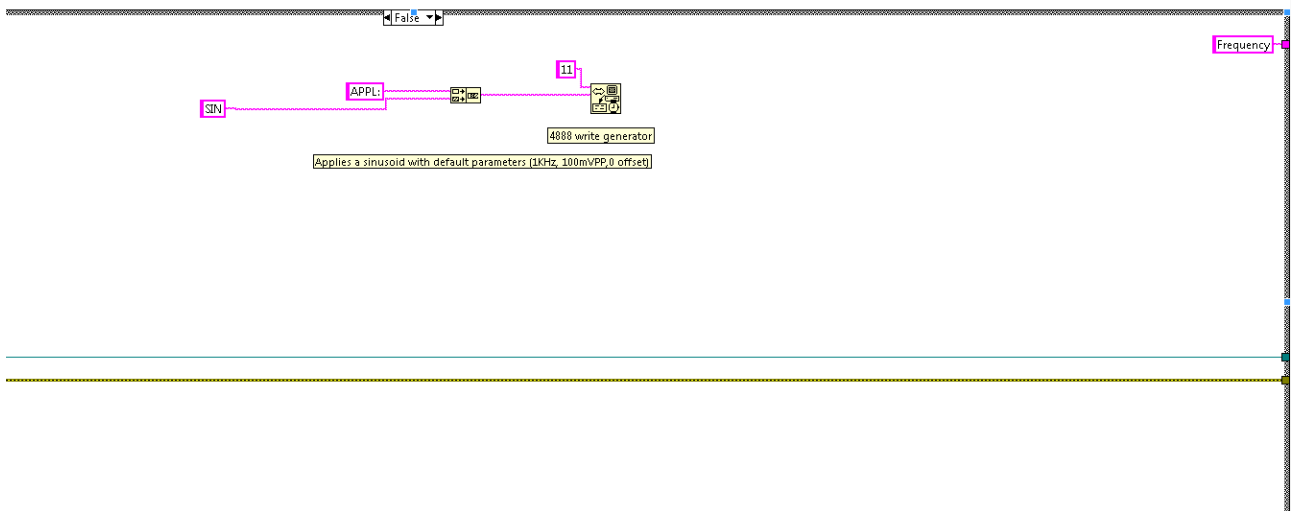
Per completare il comando si è implementato un ulteriore case che riceveva in ingresso il tipo di forma d'onda selezionata ed in base alla scelta restituiva il comando da dover inviare al generatore.

Una volta ottenute ed elaborate tutte le sottostringhe costituenti il comando si è proceduto a concatenarle in un'unica stringa mediante il blocco *concatenate string*.

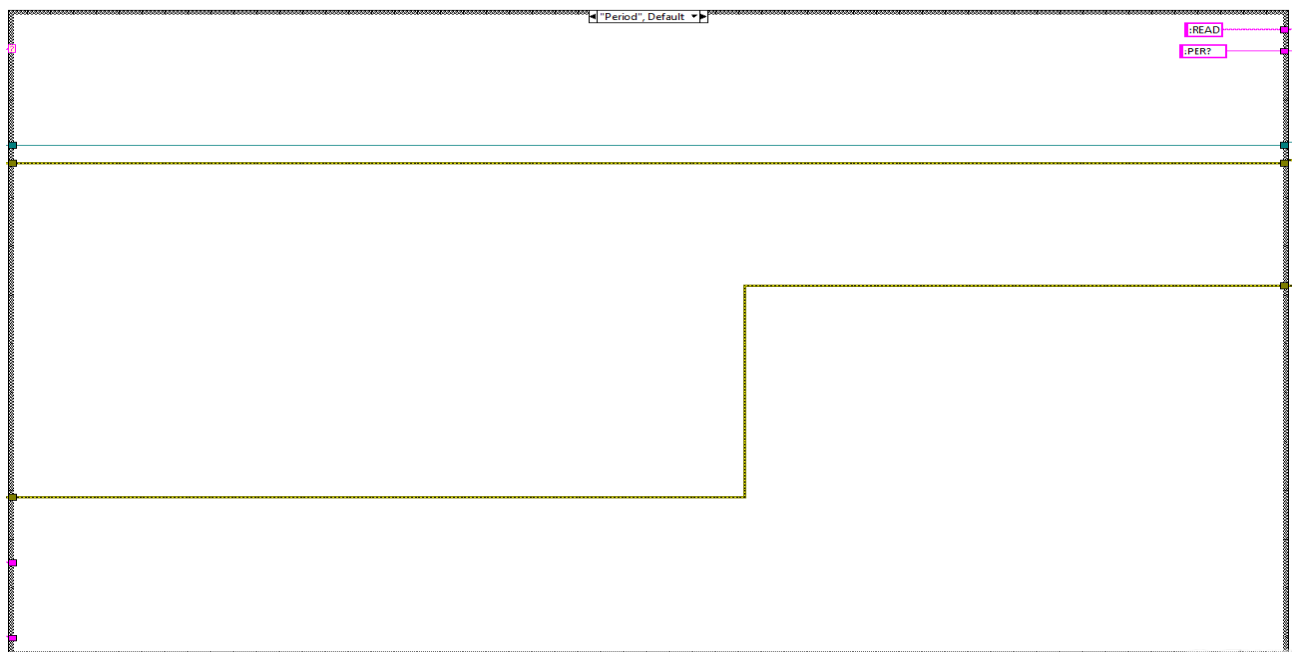
Il comando veniva inviato al generatore tramite il blocco *Write GPIB*, nel quale andava specificato l'indirizzo del generatore (in questo caso 8).



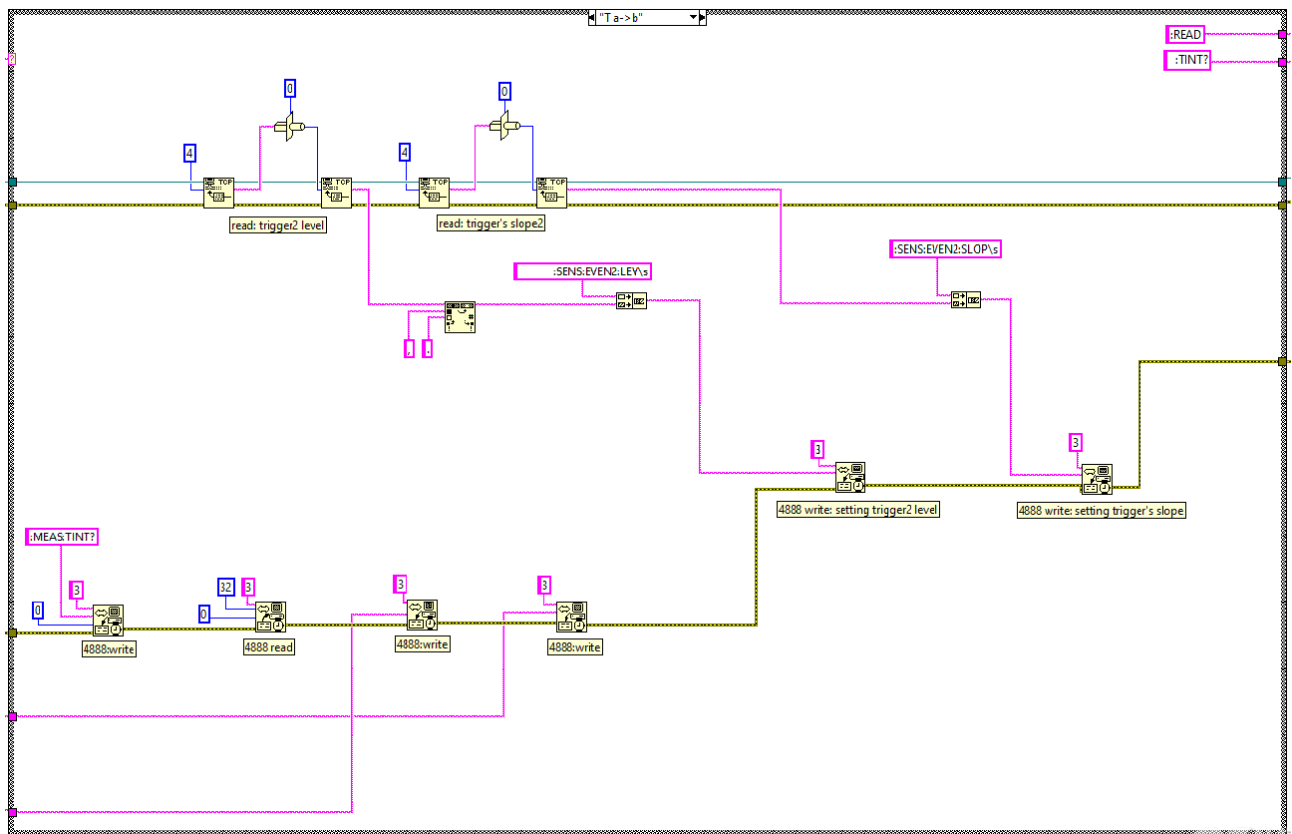
Di seguito venivano letti il valore e la pendenza del trigger, scelti nel client, ed impostati nel contatore mediante un opportuno comando. Questo veniva fatto tramite il blocco *GPIB Write*, il quale invia il comando :SENS:EVENT1:LEV/s per il livello di trigger, e :SENS:EVENT1:SLOP/s per la pendenza. In questa operazione andava specificato l'indirizzo del generatore (che nel nostro caso era 3).



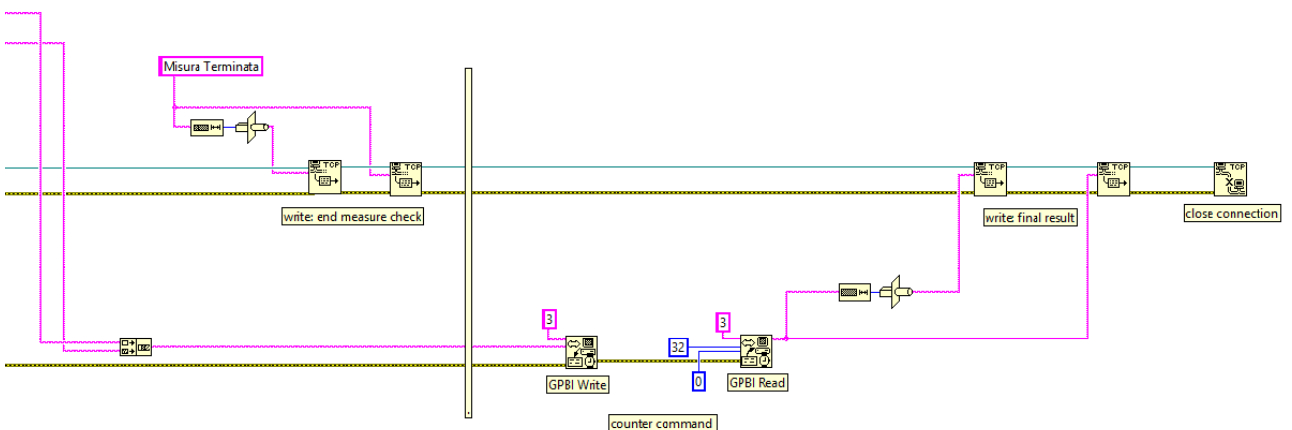
Nel caso false, invece, non veniva settato alcun parametro, ma veniva inviato solo un messaggio di errore, e generatore e contatore venivano impostati su parametri di default. Anche il livello di trigger e la pendenza sono messi di default a 0 e POS.



Successivamente si è implementato un case che fornisce una sintassi diversa per il comando da inviare al contatore, a seconda del tipo di misurazione scelta. Di default veniva eseguita la misurazione di periodo.



Nel caso di misurazioni di intervallo di tempo tra due eventi (T_{a-b}), si è implementata una procedura di impostazione del contatore diversa dalle altre misurazioni. Infatti per eseguire questa misurazione era necessario abilitare entrambi i canali del contatore. A tal proposito si è inviato preliminarmente il comando :MEAS:TINT che eseguiva la misura di intervalli di tempo con il livello di trigger impostato al 50% del segnale, ma che impostava il contatore ad un setting di default nel quale venivano anche abilitati entrambi i canali. Dopodiché si andava a cambiare soltanto i livelli di trigger con le relative pendenze, che erano l'unico parametro rilevante in questa misura, e si faceva la query standard per mezzo del comando READ:TINT.



Infine, attraverso il blocco di *write GPIB*, si è inviato il comando al contatore ed attraverso un successivo blocco di *read GPIB* si è ottenuto il risultato della misurazione che è stato poi inviato al client per mezzo del blocco di *write TCP/IP*.

Infine si è inviato al client il messaggio di fine misurazione e si è chiusa la connessione TCP/IP.