



Rapporti Tecnici INAF INAF Technical Reports

Number	274
Publication Year	2023
Acceptance in OA@INAF	2023-05-22T09:13:21Z
Title	Software per acquisizione segnali Total Power polarimetrici a larga banda
Authors	MACCAFERRI, ANDREA; FIORENTINI, Matteo; ORLATI, ANDREA
Affiliation of first author	IRA Bologna
Handle	http://hdl.handle.net/20.500.12386/34164 ; https://doi.org/10.20371/INAF/TechRep/274

Fiorentini Matteo ^[1], Maccaferri Andrea ^[1], Orlati Andrea ^[1],

Software per acquisizione segnali Total Power polarimetrici a larga banda

Aprile 2023

[1] INAF, Istituto di Radioastronomia di Bologna, Via Fiorentina, 3513, Medicina (BO)

Sommario

Introduzione	2
Setup hardware	2
Moduli NI 9205.....	2
Controller NI cRIO-9056	4
Alimentatore NI PS-15.....	5
Server	5
Configurazione software di acquisizione.....	6
Configurazione software FPGA.....	6
Sincronizzazione	8
Acquisizione del segnale.....	9
Configurazione canali di ingresso.....	11
Configurazione software Real Time	12
DMA FIFO.....	13
Controllo acquisizione.....	13
Creazione pacchetti dati.....	14
Invio dati e comunicazione con l'utente.....	14
LED e indicatori di debug.....	15
Configurazione software Data Client	17
Comandi impostabili dall'utente	19

Introduzione

Il presente documento ha l'obiettivo di descrivere la struttura e le funzionalità del sistema realizzato al fine di eseguire l'acquisizione dei segnali Total Power all'interno del progetto di Distribuzione IF presso il ricevitore di SRT. Sono presentati dapprima i componenti hardware del sistema, basato sull'utilizzo del controller CRIO-9056 insieme ai moduli di acquisizione NI 9205 commercializzati dalla National Instruments. Successivamente viene illustrato il software che rende possibile l'acquisizione, progettato in ambiente LabVIEW.

Setup hardware

Si riporta in Tabella 1 l'elenco degli strumenti che compongono il sistema di acquisizione insieme a una breve descrizione delle loro caratteristiche di interesse per il progetto in esame. In Figura 1 lo schema del setup realizzato, con i livelli funzionali del sistema indicati a fianco.

Tipo	Quantità	Codice	Descrizione
Modulo di acquisizione	19	NI 9205	32 canali analogici single-ended
Controller	3	NI cRIO-9056 <ul style="list-style-type: none"> • 01F484B0 • 01F484B1 • 01F48494 	Cestello di connessione e controllo moduli di acquisizione
Alim. Chassis	2	NI PS-15	Alimentazione controller
Server	1	Dell PowerEdge 350 F3W3N	Esecuzione software Data Client

TABELLA 1 – HARDWARE UTILIZZATO

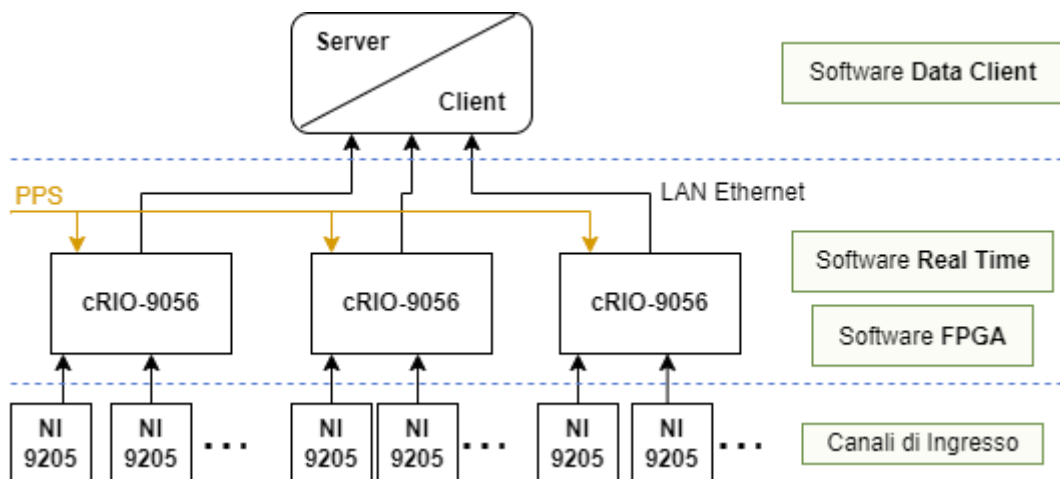


FIGURA 1 – SETUP SISTEMA ACQUISIZIONE

Moduli NI 9205

Il modulo NI 9205 è un modulo di tipo C-Series compatibile con ogni sistema CompactDAQ o CompactRIO commercializzato dalla National Instruments.

Esso presenta 32 canali di ingresso analogici, oltre ad un ingresso ed una uscita digitali, tramite un connettore a 37 pin. La connessione agli ingressi è, nella versione scelta, di tipo DSUB. I canali accettano tensioni nei

range ± 200 mV, ± 1 V, ± 5 V, ± 10 V e possono sopportare sovratensioni fino a ± 30 V, misurate rispetto all'ingresso COM. Nella configurazione DSUB la massima tensione accettabile verso terra è di 60 Vdc, con isolamento di categoria I.

In ogni modulo i canali condividono una terra comune, facente capo all'ingresso COM, che è isolata dagli altri moduli presenti nel medesimo cestello. In ogni modulo è presente un amplificatore, chiamato in Figura 1 "PGIA" (Programmable Gain Instrumentation Amplifier) e un convertitore analogico-digitale "ADC" a 16 bit con massima sample rate di 250 KHz, a cui tutti i canali del modulo vengono connessi in sequenza attraverso un multiplexer, "MUX". Le altre denominazioni presenti in Figura 2 verranno chiarite in Tabella 2.

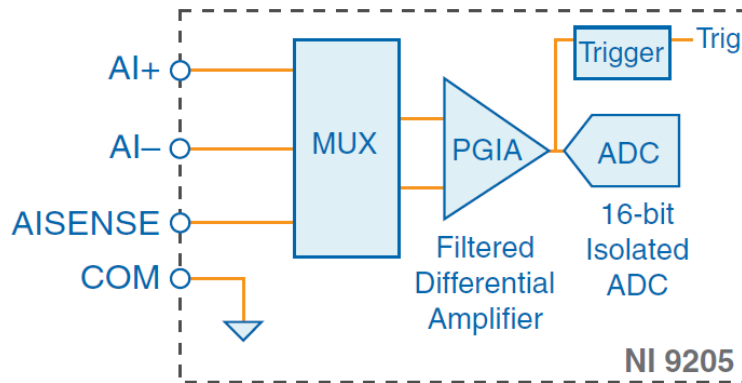


FIGURA 2 - CIRCUITO INTERNO RELATIVO AD UN CANALE ANALOGICO

Si riporta in Figura 3 la piedinatura del modulo di acquisizione e ne si fornisce una breve descrizione in Tabella 2.

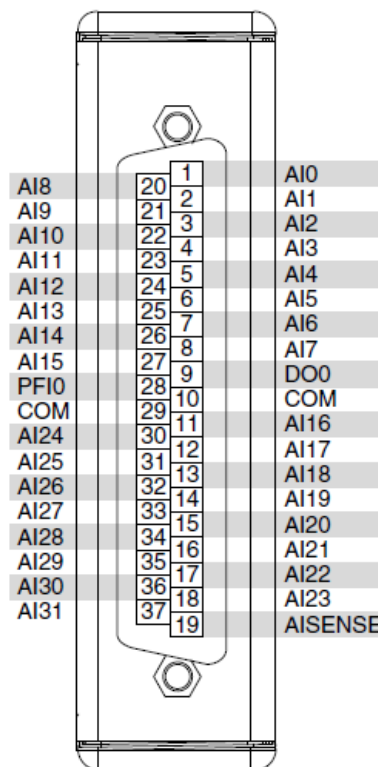


FIGURA 3 - PINOUT INGRESSI MODULO 9205

AI	Canale di ingresso analogico
AISENSE	Riferimento per i canali AI in modalità NRSE
COM	Ingresso connesso internamente al riferimento di terra
DO	Canale di uscita digitale
PFI	Interfaccia utilizzabile come ingresso digitale

TABELLA 2 - DESCRIZIONE CANALI DI INGRESSO

Nella configurazione utilizzata per il progetto in esame vengono acquisiti 32 segnali analogici di tipo single-ended in modalità NRSE (Non-Referenced Single-Ended), di cui si riporta uno schema elettrico di esempio in Figura 4. In configurazione NRSE, il modulo misura ogni segnale in ingresso rispetto all'AISENSE, collegato all'ingresso negativo dell'amplificatore PGIA. Questo tipo di collegamento migliora la reiezione del rumore rispetto ad una configurazione single ended, avvicinandosi alle prestazioni di una configurazione a differenziale puro, senza richiedere l'utilizzo di 2 ingressi per ciascun segnale da acquisire, il che renderebbe necessario l'uso di un numero doppio di moduli.

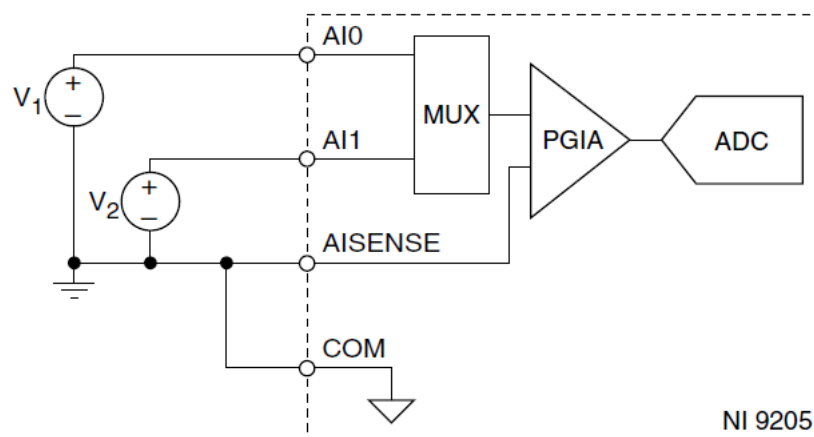


FIGURA 4 - COLLEGAMENTO DI DUE SEGNALI DI INGRESSO

Controller NI cRIO-9056

Il cRIO-9056 è un controller per applicazioni di controllo e acquisizione di dati. È dotato di un processore dual-core Intel Atom da 1.33 GHz, 4 GB di archiviazione, 1 GB di memoria RAM, una scheda FPGA Artix-7 e otto slot per il collegamento di moduli C-Series. In Figura 5 si ha un esempio della connessione dei moduli di ingresso al controller.

Sono presenti una porta Gigabit Ethernet, una porta USB 3.1 *host* e una porta USB 2.0 *device* di tipo C, una linea di trigger e uno slot per scheda microSD per archiviare i dati in locale. Il controller è dotato due LED programmabili, USER FPGA e USER1, che si possono pilotare programmando, rispettivamente, la scheda FPGA e il processore Real Time. La porta PFI 0, (Programmable Function Interface), che consiste in un connettore SMB, può essere usata come ingresso o uscita di sincronizzazione. Nel sistema in esame vi viene collegato il segnale esterno di PPS, come descritto nel paragrafo relativo alla sincronizzazione della FPGA.

Il controller utilizza il sistema operativo NI Linux Real-Time, che può controllare ingressi e uscite attraverso i driver NI-DAQmx o per mezzo del software FPGA LabVIEW. Il controller si fa carico dell'esecuzione del software a livello Real Time e FPGA, illustrati nel successivo capitolo.

La sincronizzazione in tempo, le temporizzazioni e le comunicazioni deterministiche sulla rete, di default, possono essere realizzate per mezzo del networking TSN, ma nella nostra applicazione si è preferito utilizzare il protocollo NTP.



FIGURA 5 - ESEMPIO DI CONNESSIONE MODULI A CONTROLLER

Alimentatore NI PS-15

Il PS-15 è un alimentatore industriale montabile su guida DIN, con una potenza di uscita di 120 W con una temperatura di funzionamento compresa tra -25 ° C e +60 ° C. Accetta in ingresso tensioni fra 100 e 240 Vac, selezionati in automatico. Attraverso l'accensione del LED "DC OK" viene indicata la disponibilità di alimentazione DC ai terminali di uscita.

Server

La macchina utilizzata come server è dotata di un processore a 64 bit Xeon, 16 GB di memoria RAM, 600 GB di hard disk. Sul server viene eseguito il software Data Client, descritto in seguito

Configurazione software di acquisizione

Il controller cRIO-9056 è dotato di un processore Real Time (RT) e di una scheda FPGA programmabile contenuta nello chassis, cui sono collegati moduli di ingresso e uscita. Il processore RT offre un comportamento prevedibile ed affidabile, mentre la scheda FPGA permette di eseguire molto velocemente le istruzioni più semplici, risultando adatta ad ospitare gli algoritmi di campionamento dei segnali in ingresso ai canali analogici.

L'acquisizione del segnale viene dunque realizzata predisponendo il software su entrambi i livelli (FPGA e cpu RT), il controllo da parte dell'Utente delle varie operazioni viene invece svolto da un programma eseguito sulla macchina server, indicato come "Data Client". La programmazione del sistema di acquisizione complessivo avviene quindi su tre livelli: FPGA, Real Time e Data Client. Su ogni controller cRIO sono caricati entrambi i programmi, il primo che si traduce nella configurazione della FPGA, o per meglio dire nella sintesi delle funzioni che deve svolgere la FPGA, il secondo che gira sulla CPU Real Time che manda i dati acquisiti via rete al Data Client.

I programmi caricati sui tre controller sono identici nella struttura e nelle variabili utilizzate, dove le uniche differenze sono legate al numero di moduli previsti per essere collegati ai diversi controller. Il Data Client è invece unico e si occupa di controllare l'acquisizione e a sua volta di interfacciarsi con il software di gestione della osservazione e del telescopio da cui riceve le informazioni sulla modalità osservativa desiderata, e a cui invia i dati acquisiti.

Configurazione software FPGA

La scheda FPGA è stata programmata utilizzando il software NI LabVIEW FPGA Module, incluso nel pacchetto LabVIEW Embedded Control and Monitoring Suite. Oltre al software fornito nel pacchetto, è stato necessario aggiungere la libreria esterna FPGA Timekeeper 1.1b0 per poter eseguire le procedure di sincronizzazione, come descritto al paragrafo seguente. In Figura 6 è riportato il diagramma di flusso che descrive il funzionamento del programma.

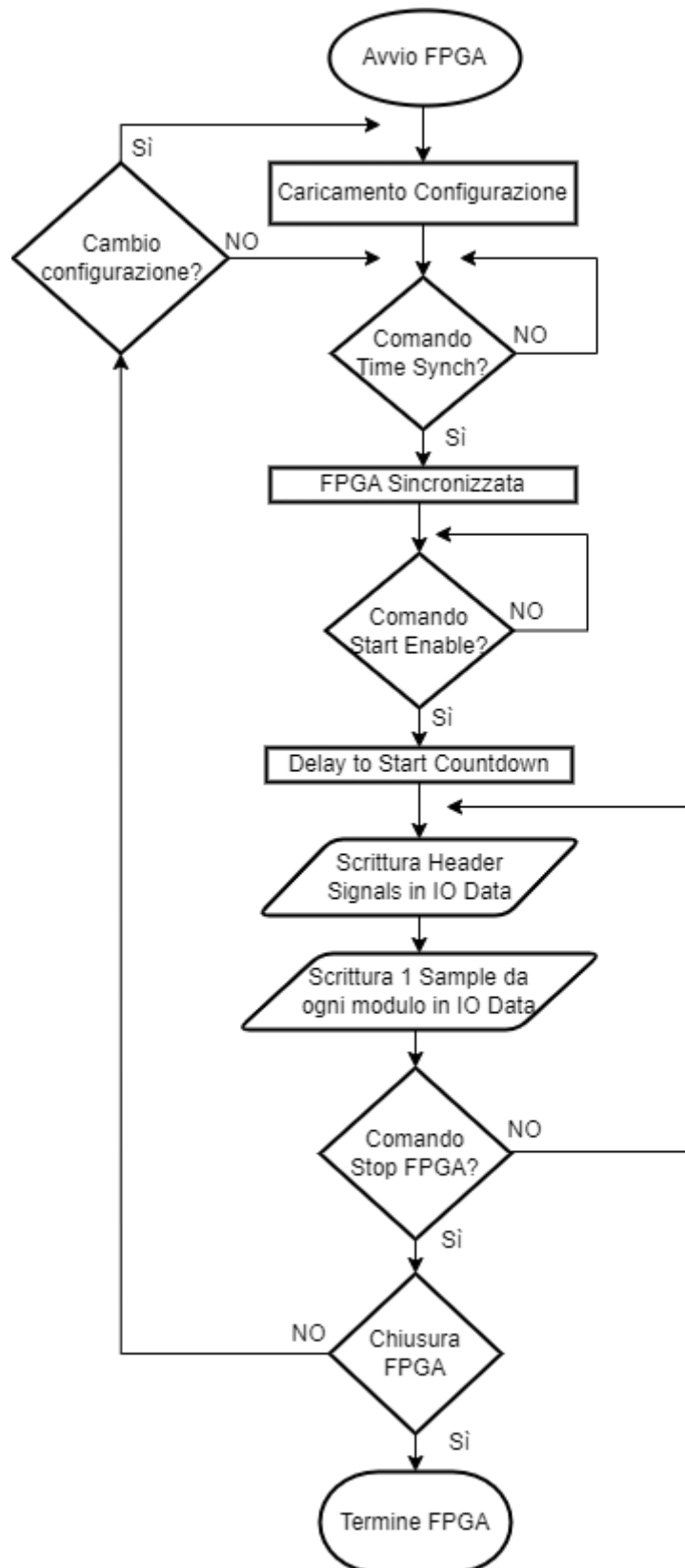


FIGURA 6 - FLOWCHART ACQUISIZIONE FPGA

Sincronizzazione

Il processore Real Time e la scheda FPGA sono componenti temporizzati, sincronizzarli è quindi necessario per compiere l'acquisizione dei segnali.

La strategia di sincronizzazione adottata nel sistema in esame si basa sulla ridondanza di segnali di riferimento temporali, ed è schematizzata nel diagramma di Figura 7.

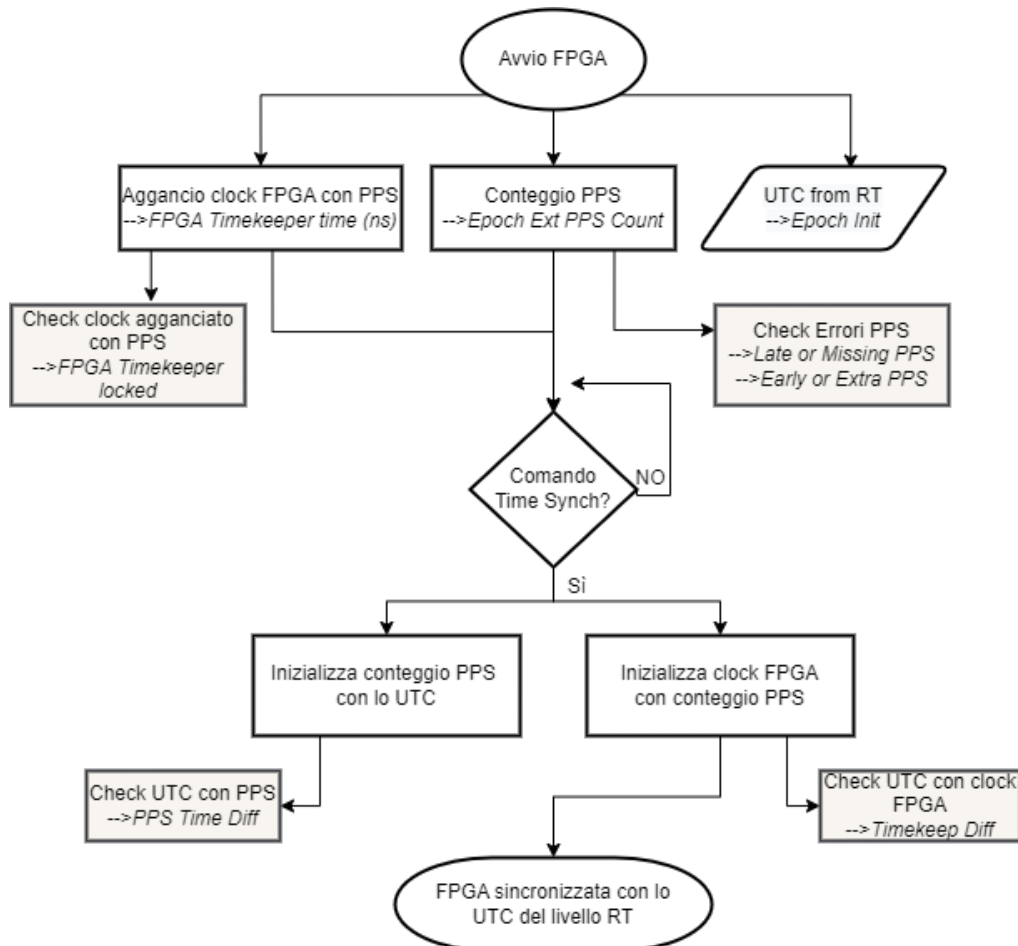


FIGURA 7 - FLOWCHART SINCRONIZZAZIONE FPGA

Il primo segnale di tempo è costituito dal clock interno alla scheda FPGA, che misura il tempo trascorso dall'accensione della scheda, con precisione del nanosecondo. Tale misura è memorizzata nella variabile *FPGA Timekeeper time (ns)*. Il secondo segnale è il *Pulse per Second (PPS o 1PPS)* di stazione, che fornisce una indicazione precisa dell'inizio di un secondo, connesso alla porta PFI 0 presente sul chassis del controller. Gli impulsi di PPS ricevuti vengono contati e il valore viene scritto nella variabile *Epoch Ext PPS Count*.

Attraverso opportuni algoritmi si aggancia il tempo del clock interno con il segnale PPS. Questo è necessario per assicurarsi che, nel tempo, i clock interni alle tre diverse schede FPGA non derivino fra di loro. Il buon esito di questa procedura è segnalato dall'accensione del LED, sul Front Panel del programma FPGA, associato alla variabile *FPGA Timekeeper locked*.

Infine vi è il segnale del tempo UTC, che a livello RT è ottenuto dalla rete utilizzando il protocollo NTP. L'informazione è memorizzata nella variabile *Unix Epoch UTC* e poi trasferita alla FPGA, dove viene utilizzato come *Epoch Init*, nella fase di inizializzazione.

Il contatore di PPS viene infatti inizializzato al tempo UTC, ovvero al valore di *Epoch Init*, mediante il comando di *Time Synch*. Il *Time Synch* inoltre inizializza il *FPGA Timekeeper time (ns)* con il valore di *Epoch Ext PPS*

Count. A seguito di questo comando, quindi, la variabile *Epoch Ext PPS Count* contiene l'attuale tempo UTC e viene incrementato dal fronte del segnale esterno PPS, mentre il *FPGA Timekeeper time (ns)* riporta la misura del tempo incrementata dal clock della FPGA.

In condizioni di corretto funzionamento, tutte e tre le variabili *Unix Epoch UTC*, *Epoch Ext PPS Count* e *FPGA Timekeeper time (ns)* contengono dunque la stessa informazione, con accuratezza dell'ordine delle decine di nanosecondo. Questa ridondanza è utile per evidenziare un errore di ricezione di un segnale, o di deriva nel tempo del clock interno. Gli indicatori *offset from time reference* e *FPGA Timekeeper locked* permettono di valutare se il clock FPGA continua ad essere agganciato al PPS. Il segnale di PPS viene a sua volta sottoposto a controllo, ed eventuali errori vengono segnalati dall'accensione sul Front Panel dei LED *Late or Missing PPS* e *Early or Extra PPS*.

A livello Real Time, inoltre, per mezzo delle variabili *PPS Time Diff* e *Timekeep Diff* si può verificare lo sfasamento del tempo misurato rispettivamente dal PPS e dal clock FPGA con il tempo UTC ottenuto da rete. Dalla valutazione dell'insieme di tali indicatori l'Utente può così accertarsi se FPGA e controller RT siano sincronizzati tra loro, o dove sia il problema in caso contrario. Il Front Panel è mostrato in Figura 9. (Questi parametri sono trasmessi anche al Data Client, per essere mantenuti sotto controllo e inseriti nei dati acquisiti.)

Il comando di *Time Synch* permette dunque di sincronizzare i livelli FPGA e Real Time, ed è pertanto un passo necessario ad ogni accensione del controller per poter procedere con l'acquisizione. In seguito, tale comando risulta necessario solo se il tempo scandito dal PPS si discosta dal tempo UTC a causa di errori nel segnale di PPS.

Il LED presente sullo chassis del controller, "USER FPGA", lampeggia comandato dalla FPGA ad ogni picco di PPS ricevuto.

Acquisizione del segnale

L'algoritmo di acquisizione del segnale consiste nell'esecuzione di un ciclo *while* a intervalli di tempo corrispondenti al periodo di campionamento richiesto. Tale periodo si imposta per mezzo della variabile *Sample Period*, il cui minimo previsto è di 1 ms, corrispondente ad una frequenza di campionamento massima di 1 KHz.

Ad ogni campionamento idealmente si vorrebbero acquisire tutti i 608 segnali contemporaneamente, purtroppo un sistema di acquisizione di 608 canali simultanei sarebbe troppo ingombrante e costoso, si è quindi optato per l'utilizzo di moduli che permettono di acquisire per mezzo di un multiplexer i 32 canali in sequenza da 0 a 31. È possibile acquisire un nuovo canale al massimo ogni 4 microsecondi, e considerando il settling time necessario al multiplexer per assestarsi, si è preferito utilizzare come periodo di scansione 8 microsecondi.

Avendo a disposizione diversi moduli è possibile acquisire in parallelo il segnale dei vari moduli, quindi dopo 256 microsecondi (8 μ s x 32 canali) si saranno acquisiti tutti i canali desiderati.

È possibile modificare il periodo di scansione del multiplexer, indicato con *Minimum Time Between Conversions*, attraverso le proprietà dei moduli, accessibili nella finestra del Project Explorer come mostrato in Figura 10 - proprietà dei moduli nel Project Explorer.

Ad ogni iterazione vengono inoltre scritte in memoria 3 informazioni relative all'istante in cui vengono acquisiti i campioni: i valori delle già presentate variabili *Epoch Ext PPS Count* e *FPGA Timekeeper time (ns)*, e del *Acq Loop Counter*, che consiste nell'iterazione del ciclo di acquisizione in cui avviene il campionamento. Queste informazioni vengono nell'insieme indicate come *Header Signals*, in quanto vengono scritte prima dei

campioni, e permettono di conoscere l'istante esatto di campionamento dei dati di un certo pacchetto. Uno schema dell'acquisizione dei dati è mostrato in Figura 8.

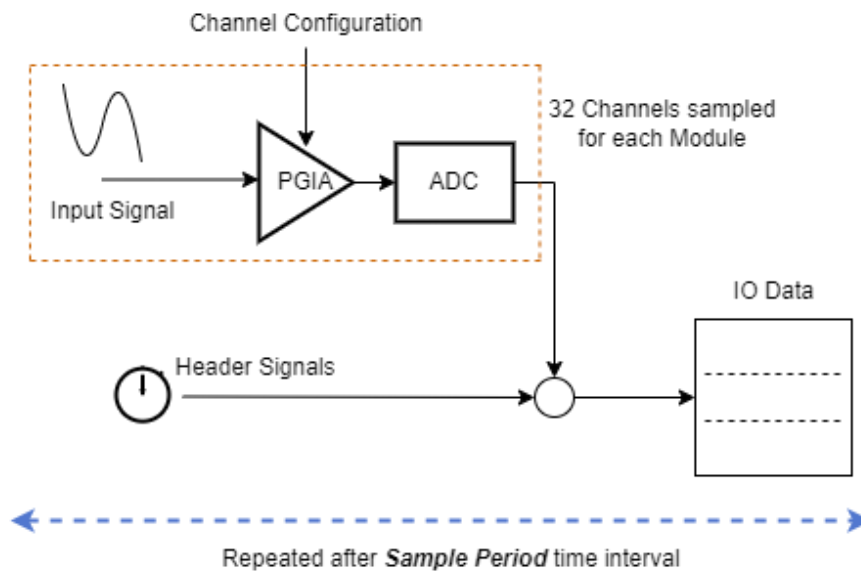


FIGURA 8 - CAMPIONAMENTO E SCRITTURA IN MEMORIA

L'inizio dell'acquisizione è attivato dal comando *Start Enable*: una volta ricevuto dalla FPGA, il ciclo di acquisizione comincia all'impulso di PPS successivo allo scadere di un countdown di lunghezza impostabile per mezzo della variabile *Delay to Start*. Essa è di default pari a 1 secondo, l'acquisizione comincia quindi all'impulso di PPS successivo allo *Start Enable*.

L'acquisizione termina al comando di *Stop*, che fa concludere il ciclo di acquisizione e riporta il programma nella condizione di ricevere un nuovo *Start Enable*. In Figura 9 si mostra il Front Panel con i pulsanti di comando e gli indicatori riportati anche a livello Real Time, che perciò verranno illustrati nel capitolo ad esso relativo.

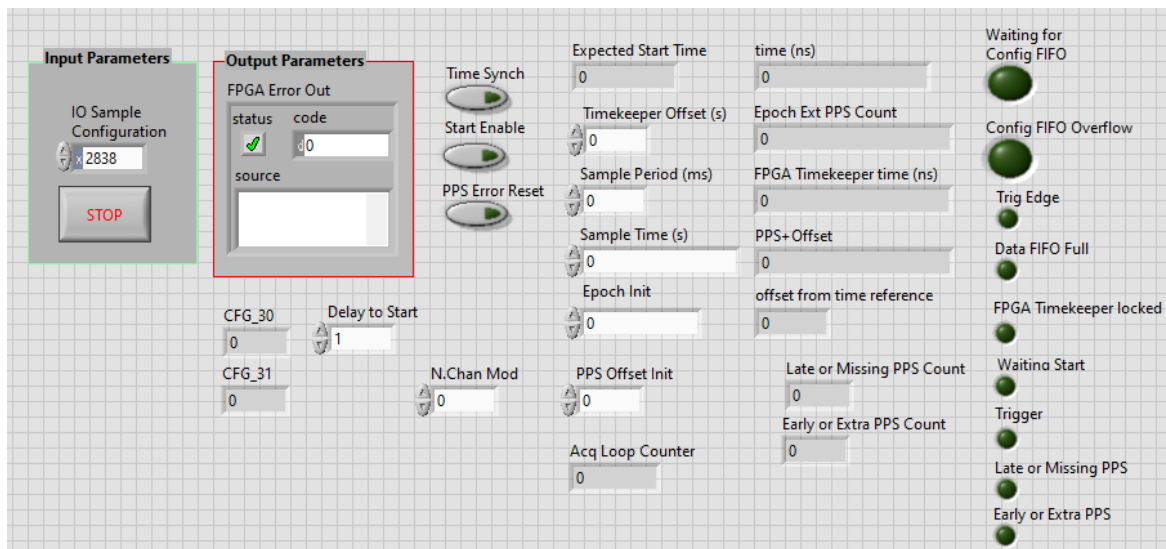


FIGURA 9 - FRONT PANEL PROGRAMMA FPGA

Configurazione canali di ingresso

Al fine di acquisire correttamente è importante assicurarsi che i moduli NI 9205 abbiano l'adeguata configurazione, che è stata fissata attraverso le proprietà dei moduli accessibili nel Project Explorer e può anche venire reimpostata ad ogni nuova acquisizione per mezzo dei controlli presenti nel programma Real Time. La configurazione è preimpostata per essere $\pm 5\text{ V}$ e NRSE, il tipo di collegamento scelto per i segnali single-ended. L'accesso alle proprietà dei moduli è mostrato in Figura 10.

Attraverso le proprietà dei moduli si può inoltre scegliere la *Calibration Mode*. Scegliendo "Calibrated" si ottengono dei campioni in Volt e loro frazioni, mentre selezionando "Raw" i campioni sono da intendere come conteggi, successivamente da convertire nell'unità di misura desiderata in base alla conoscenza dei parametri dell'ADC. Nella modalità Raw, che è quella prevista per il progetto in esame, i campioni acquisiti sono memorizzati in variabili di tipo I16, per questo motivo la FIFO *IO Data* è stata impostata per contenere variabili dello stesso tipo.

Oltre ai canali di ingresso, è stato configurato il canale di output digitale, DO, del modulo nello slot 0 di ogni controller per generare un segnale che diventa alto ad inizio di ogni campionamento e termina con l'acquisizione dell'ultimo canale del suddetto campionamento. Questo permette, visualizzando per mezzo di un oscilloscopio l'uscita di tale DO, di verificare sia la frequenza di campionamento che la durata del tempo necessario per campionare tutti i canali richiesti.

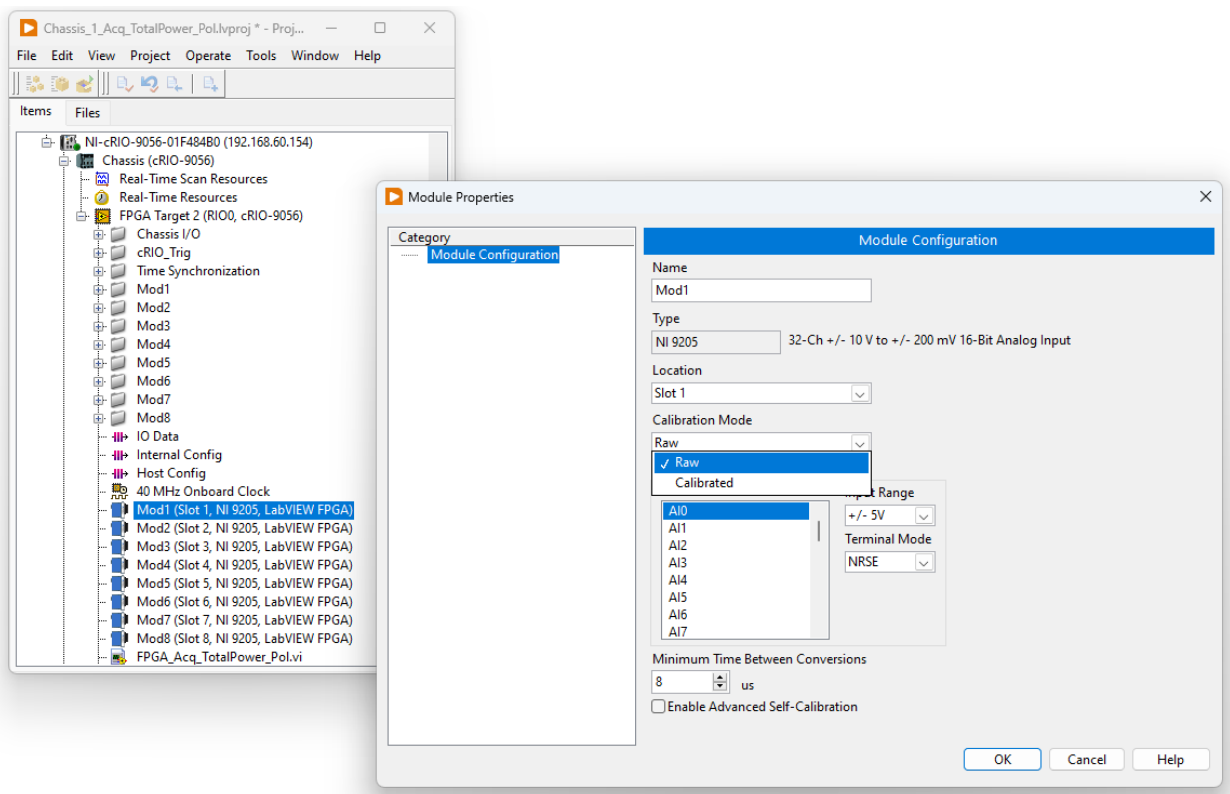


FIGURA 10 - PROPRIETÀ DEI MODULI NEL PROJECT EXPLORER

Configurazione software Real Time

La programmazione del software eseguito dal processore Real Time è stata realizzata utilizzando il software NI LabVIEW Real-Time Module, versione 2021 SP1 f1, 32 bit. Il compito principale svolto a questo livello è quello di governare l'esecuzione del programma FPGA, inviando ad esso i comandi e ricevendo informazioni e dati di acquisizione. Attraverso una comunicazione TCP/IP con il programma Data Client, il software RT permette all'Utente di scambiare comandi e dati con la FPGA. In Figura 11 il diagramma di flusso che illustra l'utilizzo del software Real Time.

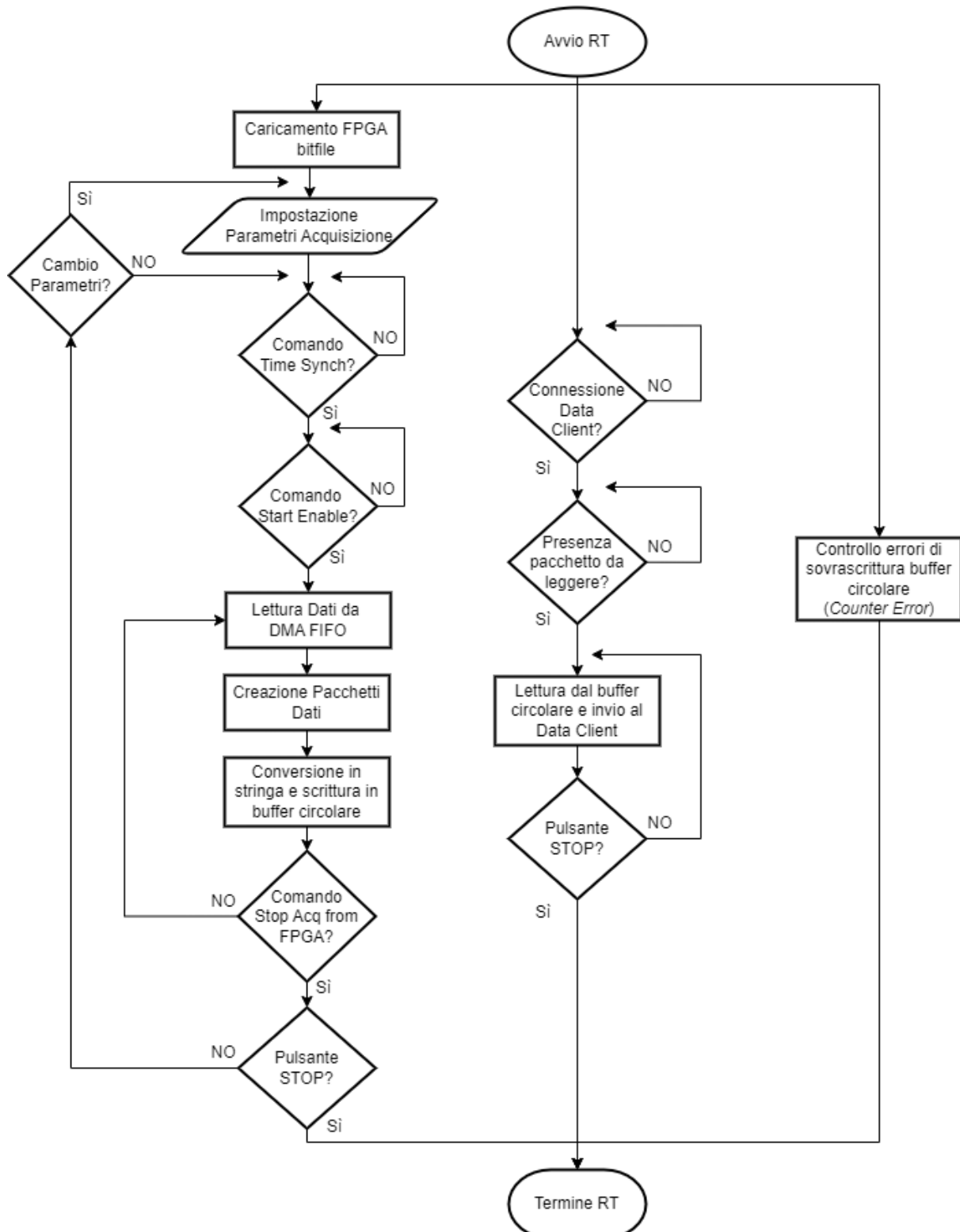


FIGURA 11 - FLOWCHART SOFTWARE REAL TIME

DMA FIFO

Il DMA, Direct Memory Access, è un metodo di trasferimento dati che non richiede l'intervento della CPU per trasferire dati fra il FPGA target e la memoria dati dell'Host computer.

Un canale DMA consiste in un buffer di tipo FIFO: la FPGA provvede a scrivere i propri dati in sequenza all'interno della FIFO e la CPU Real Time, quando questa ha all'interno un numero sufficiente di dati, provvede ad estrarli con lo stesso ordine con cui sono stati scritti dalla FPGA, ovvero secondo il principio di una struttura FIFO. Uno schema del trasferimento è illustrato in Figura 12.

A livello FPGA, ad ogni campionamento i dati acquisiti vengono impacchettati assieme all'header che ne identifica il tempo di campionamento e permette poi offline la verifica di corretto sincronismo, e quindi vengono inseriti direttamente nella FIFO.

Il processo DMA, in presenza di dati nel buffer FIFO, li rende disponibili alla CPU Real Time. I dati ricevuti vengono quindi salvati temporaneamente in un ulteriore buffer circolare la cui dimensione può essere impostata dall'Utente attraverso la variabile *N Page Circular Buffer*. Un secondo processo attivo a livello Real Time si occupa quindi di verificare il sincronismo e la corretta sequenza dei pacchetti dati ricevuti e di inviarli via rete al Data Client che sovrintende alla gestione delle osservazioni e al salvataggio dei dati.

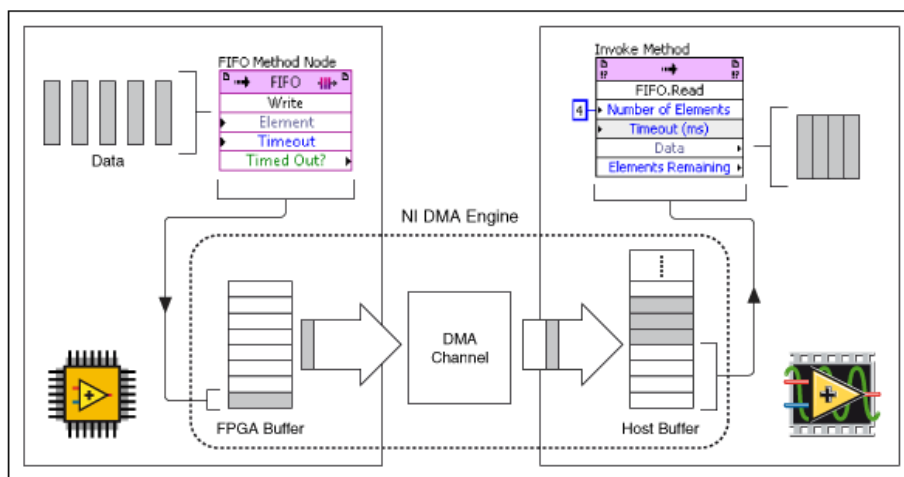


FIGURA 12 - TRASFERIMENTO DATI ATTRAVERSO CANALE DMA

Controllo acquisizione

Il programma Real Time si occupa innanzitutto di aprire il collegamento con il programma FPGA caricandone il bitfile, ovvero il file contenente tutte le informazioni sulla logica interna del programma, generato ad ogni compilazione. Tale programma permette inoltre di collegare il segnale di PPS, che arriva al terminale PFI 0 dello chassis del controller alla scheda FPGA, dirigendolo verso il terminale di trigger interno *cRIO_Trig5*.

Vi è la possibilità di inviare al programma FPGA le impostazioni di configurazione dei canali di ingresso selezionabili nel Front Panel attraverso il pulsante *Update Module Configuration* di Figura 13. Mediante il Project Explorer è possibile preimpostare la configurazione dei moduli NI 9205 che verrà utilizzata dal programma Real Time.

Il controllo dell'acquisizione prevede che l'Utente per mezzo del Data Client imposti i comandi e i parametri a livello Real Time, e le configurazioni opportune da trasferire a livello FPGA. Il Data Client ha a disposizione gli stessi comandi previsti a livello FPGA, escluso il comando di *Stop*, che qui è chiamato *Stop Acq from FPGA*. Come descritto precedentemente, quindi, una volta sincronizzati FPGA e RT con il comando di *Time Synch*, si

può avviare l'acquisizione con lo *Start Enable* e terminarla con uno *Stop Acq from FPGA*. Lo *Start Enable* determina l'inizio del trasferimento di dati attraverso il canale DMA.

Il Front Panel del Real Time visualizza le informazioni relative allo stato dell'esecuzione del software FPGA, e permette di variare i parametri con cui avviene la creazione dei pacchetti di dati da inviare all'Utente del Data Client.

Creazione pacchetti dati

I campioni acquisiti vengono trasferiti, attraverso il canale DMA, nel buffer del programma Real Time che li formatta in stringhe e li trasferisce in un buffer circolare per il trasferimento verso il Data Client.

Il buffer contenente i dati provenienti dalla DMA FIFO *IO Data* viene monitorato dal programma Real Time fino al raggiungimento della dimensione necessaria alla formattazione. La dimensione è determinata dai seguenti parametri, impostabili dall'utente o determinati dalla configurazione hardware, mostrati nel Front Panel di Figura 13:

- *N.Chan Mod*: numero di canali di ingresso per ogni modulo NI 9205, uguale a 32 nel caso in esame.
- *N Modules*: numero di moduli connessi al controller, dipende dalla configurazione scelta per il sistema, attualmente 7 per i primi due controller e 5 per il terzo. (Si prevede al massimo di dover acquisire 8 bande per ogni feed, un massimo di 19 feed, 4 parametri per banda, totale 608 canali.
- *N Header Signals*: numero di segnali che portano l'informazione contenuta negli Header Signals, che consistono nei valori delle variabili *Epoch Ext PPS Count*, *Acq Loop Counter* e *FPGA Timekeeper time (ns)*, come descritto al paragrafo "Acquisizione del segnale".
- *Sample Period (ms)*: periodo di campionamento impostato dall'utente, previsto per un minimo di 1 ms fino ad un massimo di 40 ms.
- *Bundle rate (ms)*: intervallo di tempo, in millisecondi, che definisce la grandezza del pacchetto. Definiti gli altri parametri, questo intervallo determina il numero di elementi che devono essere letti dalla FIFO.

Ad esempio, con 1 ms di *Sample Period* e 100 ms di *Bundle Rate*, ogni pacchetto (*Block*) sarà formato da 100 acquisizioni. Nel caso si acquisiscano 32 canali da 8 moduli, si hanno 256 segnali di ingresso. Sommandovi gli 8 *Header Signals*, si ha un totale di 264 elementi per ogni acquisizione. Un pacchetto sarà così formato da 26,400 elementi.

Questo è il tipo di calcolo che viene svolto per determinare quanti elementi devono essere presenti nel buffer FIFO. Il risultato è memorizzato nella variabile *Elements per Block*.

Invio dati e comunicazione con l'utente

La comunicazione tra il programma Real Time e il Data Client, con cui l'Utente si interfaccia, avviene attraverso una connessione di tipo TCP/IP. L'invio di dati all'Utente e la ricezione dei suoi comandi avvengono all'interno di due cicli separati, chiamati rispettivamente *Data Clients Loop* e *Main Data Control Clients Loop*, per indicare i diversi ruoli svolti.

Il *Data Clients Loop* provvede a prelevare i pacchetti scritti nel buffer circolare in forma di stringhe e ad inviarli al Data Client, confrontando gli indici di scrittura e lettura del buffer stesso. La creazione del buffer circolare è una soluzione che ha lo scopo di consentire la contemporaneità di acquisizione e di invio dati al client.

Il *Main Data Control Clients Loop* consente la ricezione dei comandi e dei parametri dal Data Client e l'invio delle informazioni sullo stato dell'acquisizione e sul tempo UTC del Real Time verso il Data Client.

È inoltre presente un terzo ciclo, chiamato *Additional External Control Clients Loop* ed eseguito in parallelo ai precedenti, che permette la connessione di ulteriori programmi client esterni. Questa possibilità è stata prevista nel caso si verifici un malfunzionamento dovuto ad una manovra errata da parte del Data Client, che interrompa la connessione TCP/IP prima che l'acquisizione venga chiusa in modo corretto. Attraverso la connessione di un ulteriore client esterno, strutturato in modo simile al Data Client, si può attivare il comando "DISCONNECT" con una procedura di "emergenza" che termina i *Data Clients Loop* e *Main Data Control Clients Loop* in modo corretto liberando le risorse impegnate e permettendo così di riutilizzarle nuovamente da parte del Data Client.

LED e indicatori di debug

Sul Front Panel del programma sulla CPU RT sono presenti indicatori che forniscono informazioni di debug sullo stato di esecuzione del programma Real Time e FPGA, come riportato in Figura 13. Gli indicatori diagnostici di maggior interesse sono:

- *Config FIFO Overflow*: indica un overflow nella FIFO che contiene i dati di configurazione.
- *Waiting for Config FIFO*: indica che i dati di configurazione non sono ancora stati caricati a livello FPGA.
- *Data FIFO Full*: indica che il buffer livello FPGA del canale DMA FIFO è andato in overflow.
- *FPGA error out*: riporta le informazioni relative ad un eventuale errore insorto nell'esecuzione del programma FPGA.
- *FPGA Timekeeper locked*: LED che indica che il clock FPGA è stato correttamente agganciato al segnale esterno di PPS.
- *Waiting Start*: indica che il ciclo di acquisizione a livello FPGA non è ancora iniziato.
- *Trigger*: LED che lampeggia alla ricezione di un picco del segnale di PPS.
- *Late or Missing PPS*: indica la ricezione di un impulso PPS in più rispetto a quello previsto.
- *Early or Extra PPS*: indica la ricezione di un impulso PPS in meno rispetto a quello previsto.
- *Late or Missing PPS Count*: contatore degli errori di tipo *Late or Missing PPS*.
- *Early or Extra PPS Count*: contatore degli errori di tipo *Early or Extra PPS*.
- *PPS Error Reset*: pulsante che resetta i contatori di errore relativi al segnale di PPS.

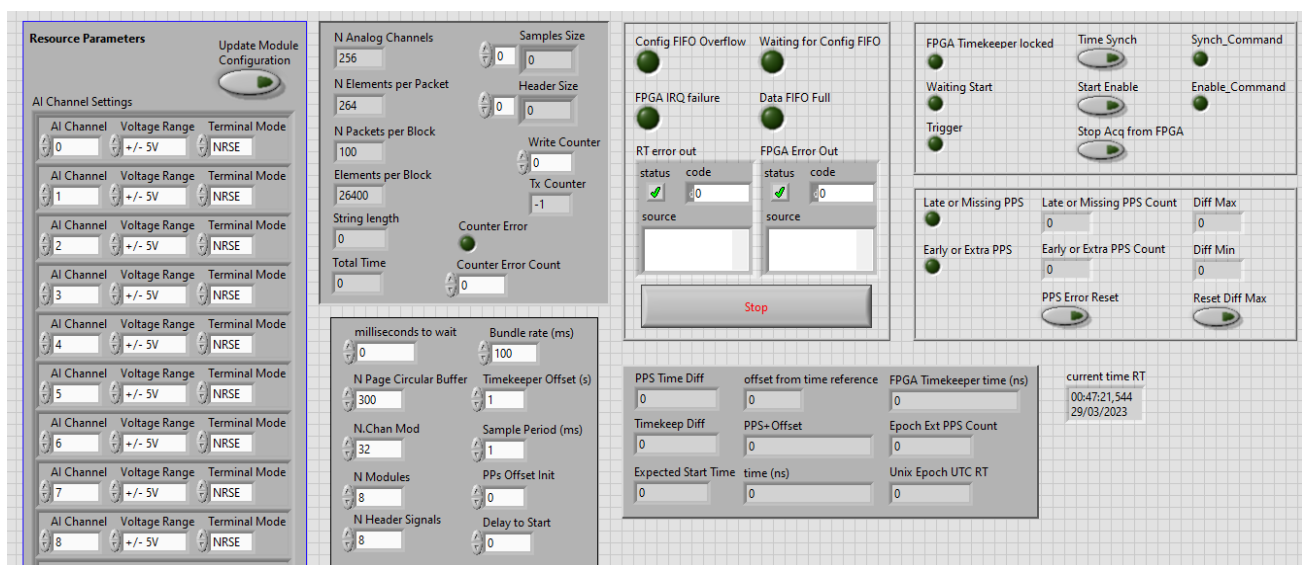


FIGURA 13 - FRONT PANEL PROGRAMMA RT

I seguenti indicatori si riferiscono all'esecuzione del software Real Time:

- *FPGA IRQ Failure*: indica un errore nel caricamento della configurazione da livello RT a FPGA.
- *Counter Error*: indica un errore di sovrascrittura nel buffer circolare.
- *Counter Error Count*: contatore degli errori di tipo *Counter Error*.
- *Block Counter*: contatore dei pacchetti inviati al Data Client.
- *Total Time*: riporta il tempo totale, in millisecondi, impiegato per svolgere tutte le istruzioni con cui si costruiscono e inseriscono nel buffer circolare i pacchetti dati.
- *Header Size*: riporta le dimensioni dell'array 2D degli *Header Signals* contenuti in un pacchetto dati.
- *Samples Size*: riporta le dimensioni dell'array 2D dei campioni contenuti in un pacchetto dati.
- *String lenght*: riporta le dimensioni in byte della stringa contenente un pacchetto dati.
- *State*: variabile *enum* che contiene i possibili stati del sistema, che l'Utente può visualizzare nel Data Client: NO TRANSFER, DATA TRANSFER, ACQ ERROR, TX ERROR. Sono illustrati nel capitolo relativo al Data Client.
- *RT error out*: riporta le informazioni relative ad un eventuale errore occorso nei cicli di controllo dell'acquisizione o nella creazione dei pacchetti dati.
- *current time RT*: indicatore che riporta la data e l'ora corrente, vista dal livello Real Time.
- *Unix Epoch UTC RT*: tempo UTC in formato Unix visto dal livello Real Time.
- *Timekeep Diff*: differenza fra lo *Unix Epoch UTC* e il *FPGA Timekeeper time (ns)*.
- *PPS Time Diff*: differenza fra lo *Unix Epoch UTC* e il *Epoch Ext PPS Count*.

È stato inoltre programmato il LED USER1 presente sullo chassis del controller cRIO-9056 per fornire un'indicazione sullo stato dell'esecuzione del software Real Time. Il LED è configurato per lampeggiare nel seguente modo:

- 1 lampeggio al secondo: Acquisizione e trasferimento dati al Data Client in corso.
- 2 lampeggi ravvicinati al secondo: Programma FPGA caricato correttamente, acquisizione non in corso.
- Lampeggi continui: Errore a livello FPGA o Real Time che impedisce l'acquisizione.

Configurazione software Data Client

Il software Data Client, che può essere eseguito sulla macchina server presente nel sistema, permette all'Utente di controllare l'esecuzione del software Real Time e quindi il processo di acquisizione svolto dalla scheda FPGA. Anche per la programmazione del Data Client si è utilizzato il software LabVIEW Professional Development System for Windows.

Nel programma sono presenti principalmente i cicli corrispondenti a quelli che a livello Real Time permettono l'invio dei dati e la comunicazione con l'utente, ovvero che si incaricano di aprire la connessione TCP attraverso le porte corrispondenti e di svolgere le funzioni di ricezione e invio parametri e comandi da parte dell'Utente.

Il programma Data Client è unico per i tre controller, quindi la ricezione dei dati avviene per mezzo di tre cicli *while* in parallelo, dove ognuno comunica con un diverso controller. Ogni ciclo è strutturato allo stesso modo e prevede essenzialmente di svolgere le operazioni inverse a quelle che hanno portato alla costruzione dei pacchetti dati nel programma Real Time. Una volta ricevuta una stringa attraverso la connessione TCP/IP, essa viene riconvertita nel bundle di due array, e gli *Header Signals* e i campioni che li compongono vengono visualizzati attraverso indicatori e grafici visualizzati sul Front Panel. Nella stessa finestra si può inoltre scegliere quali canali visualizzare per ogni controller attraverso appositi pulsanti.

Le informazioni contenute negli *Header Signals*, in particolare il tempo scandito dal PPS, vengono inoltre sottoposte a controllo per rilevare eventuali errori nella ricezione del segnale di PPS. Le informazioni relative ad eventuali errori vengono scritte su un file in formato csv mentre tutti i dati ricevuti vengono scritti in un file binario.

A seguito di una acquisizione si otterranno dunque tre file .csv chiamati `Error_Log_0`, `Error_Log_1`, `Error_Log_2`, e tre file binari il cui nome inizia con il tempo UTC dell'istante in cui si è avviata l'acquisizione, e termina con `_Chassis_0`, `_Chassi_1`, `Chassis_2`.

Il diagramma di Figura 14 descrive l'utilizzo del software.

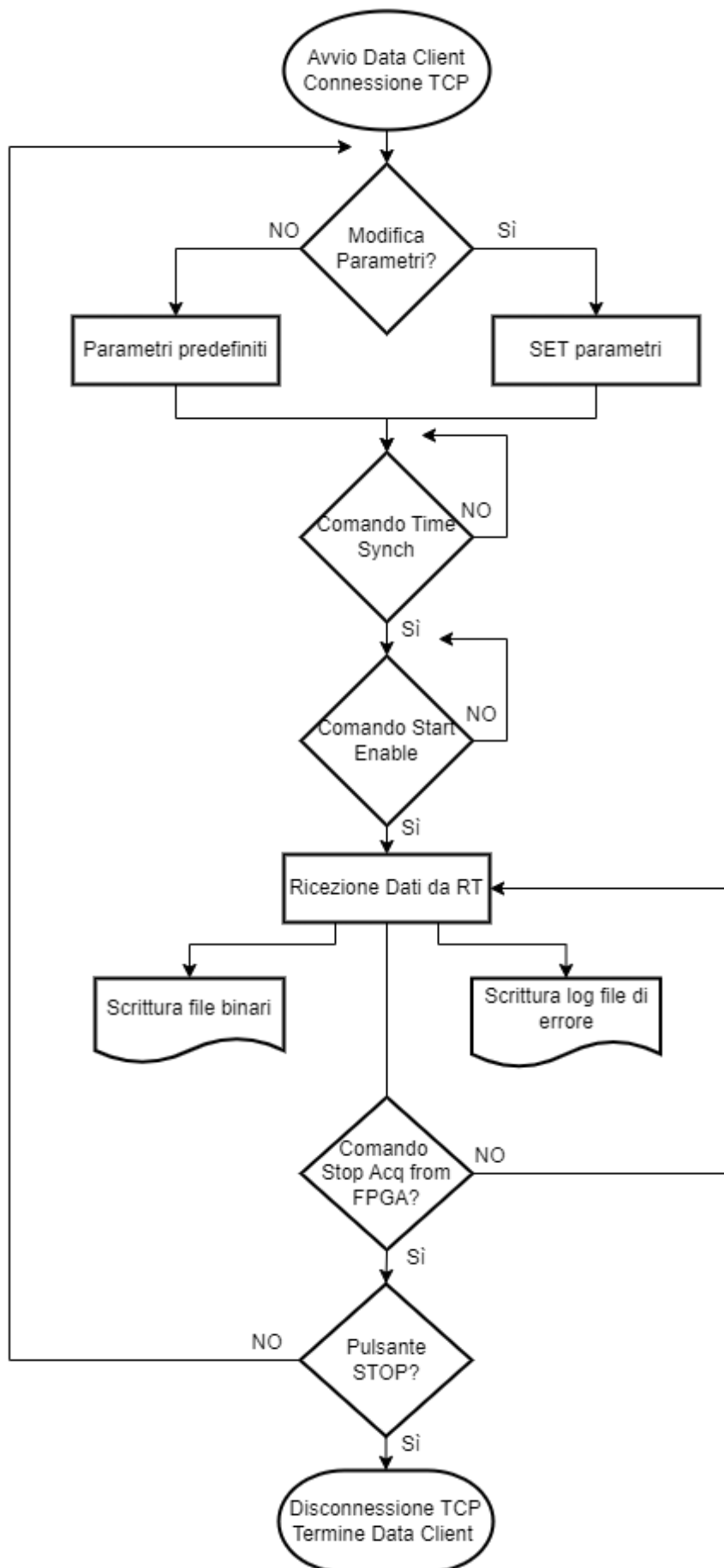


FIGURA 14 - FLOWCHART PROGRAMMA DATA CLIENT

Comandi impostabili dall'utente

La comunicazione di controllo con il programma Real Time è realizzata mediante un singolo ciclo *while* attraverso il quale i comandi vengono inviati allo stesso momento a tutti e tre i controller, mentre i messaggi di risposta da ognuno di essi sono visualizzabili separatamente. L'Utente può utilizzare i pulsanti sul Front Panel, mostrati in Figura 15, per inviare comandi.

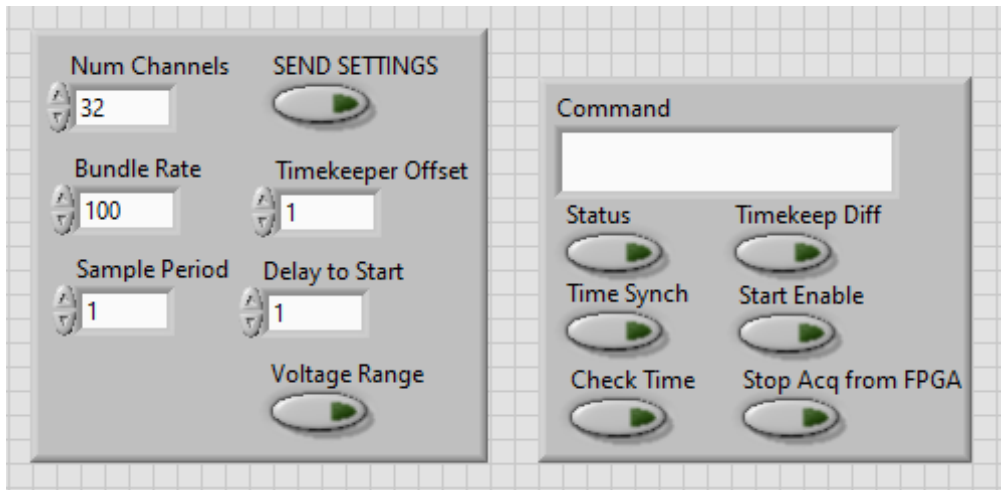


FIGURA 15 - COMANDI IMPOSTABILI DALL'UTENTE

- **Time Synchron:** si invia il comando di *Time Synchron* ai programmi Real Time sui controller, i quali poi lo trasmettono a loro volta al software sulle schede FPGA. Questo comando va inviato all'inizio della prima acquisizione per sincronizzare i controller con le schede FPGA, e in seguito solo all'occorrenza, ovvero se il tempo scandito dal PPS si discosta dal tempo UTC di riferimento a causa di errori nel segnale di PPS. L'esito positivo dell'invio del comando determina la ricezione della risposta "ACK" da parte del programma Real Time.
- **Start Enable:** come per il *Time Synchron*, si trasmette questo comando al software FPGA per cominciare l'acquisizione. Un corretto invio del comando determina la risposta "ACK"
- **Stop Acq from FPGA:** si attiva il comando di *Stop* presente nel programma FPGA e si termina così l'acquisizione. Questo comando inoltre fa terminare i cicli di ricezione dei dati e disconnette la loro connessione TCP con i corrispondenti cicli di invio dati a livello Real Time. Un corretto invio del comando determina la risposta "ACK".
- **Status:** questo comando permette di monitorare lo stato dell'acquisizione. Le risposte possibili sono:
 - "NO TRANSFER", nel caso la connessione TCP per l'invio dei dati sia stabilita ma non ci sia ancora trasferimento di dati verso il Data Client;
 - "DATA TRANSFER", nel caso il trasferimento sia in atto;
 - "ACQ ERROR", nel caso sia insorto un errore a livello FPGA o nei cicli Real Time di acquisizione e controllo;
 - "TX ERROR", nel caso sia insorto un errore nel ciclo di invio dati verso il Data Client.
- **Checktime:** attraverso questo comando si ottiene il tempo UTC in formato Unix misurato su ogni Controller. Si calcola inoltre la differenza fra ciascuno dei tempi ricevuti e il tempo visto dal Data Client, al fine di assicurarsi che il riferimento di tempo sia il medesimo.
- **Voltage Range:** con questo comando si ottiene l'informazione sul fondo scala impostato per l'acquisizione dei segnali.

In alternativa ai pulsanti presentati, l'Utente può anche digitare manualmente nella casella *Command* la parola corrispondente al comando, che è la medesima, ma solo con lettere maiuscole. Si può quindi digitare:

- TIME SYNCH
- START ENABLE
- STOP FPGA
- STATUS
- CHECKTIME

L'Utente può anche visualizzare l'elenco dei comandi accettati dal programma inviando la parola "HELP".

Si possono inoltre impostare i parametri di acquisizione per mezzo dei seguenti controlli sul Front Panel:

- **Num channels:** valore della variabile *N Chan Mod*, il valore predefinito è 32.
- **Bundle rate:** valore della variabile *Bundle rate (ms)*, di default è 100 ms.
- **Sample Period:** valore della variabile *Sample Period*. È un controllo di tipo *enum* che può assumere solo i valori 1; 2; 4; 5; 10; 20; 40 ms.
- **Delay to Start:** valore della variabile *Delay to Start*. Di default è 1, il che significa che l'acquisizione comincia al segnale di PPS successivo alla ricezione dello *Start Enable*.
- **Timekeeper Offset:** valore della variabile *Timekeeper Offset (s)*, rappresenta l'incremento da dare al tempo *Epoch Ext PPS Count* per compensare il ritardo fra l'istante in cui la sincronizzazione viene richiesta e quello in cui essa avviene. In condizioni normali corrisponde ad 1 secondo.

Una volta impostati, i valori dei parametri vengono inviati, tutti allo stesso tempo, premendo il pulsante "SEND SETTINGS".

Il comando inviato sarà composto dalla parola "SET" seguita dai valori dei parametri impostati, separati da virgole. Anche questo comando può essere in alternativa digitato dall'Utente nella casella *Command*, i parametri devono essere scritti secondo un ordine rigido, corrispondente all'ordine con cui sono stati sopra elencati. Ad esempio, un comando accettabile è "SET,32,100,1,1,2". Si sottolinea inoltre che l'invio di tali parametri non è necessario per cominciare l'acquisizione, in quanto sul programma Real Time essi saranno comunque già impostati ai loro valori predefiniti.

Tutti gli indicatori presenti sul Front Panel del Data Client vengono descritti in dettaglio nel documento "Software per acquisizione segnali Total Power polarimetrici a larga banda - Manuale Utente".