

Progetto GIANO

Inventario e memorandum del software di controllo.

E. Rossetti

Dipartimento di Astronomia, Università di Bologna

e-mail: emauel.rossetti@unibo.it

(versione: 7/10/2010)

Sommario

Nel presente memo vengono elencati i vari moduli software relativi ai controlli della criogenia e dei movimenti criogenici installati (ed operativi) sulla workstation di controllo dello strumento GIANO. Tutte le informazioni qui contenute descrivono lo stato dell'arte del software installato alla data del presente documento.

Per ogni modulo forniremo una breve descrizione per un corretto utilizzo.

Introduzione

Tutto il software ICS (Instrument Control Software) di GIANO deve essere installato in una directory di lavoro, scelta dall'utente, attraverso la definizione obbligatoria della variabile d'ambiente **GIANO_ICS**.

Sul PC del laboratorio di Arcetri (ebenezer.arcetri.astro.it), alla data della stesura del presente documento, e' stata utilizzata la seguente definizione (nel file `$HOME/.cshrc`):

```
setenv GIANO_ICS /home/giano/ICS
```

All'interno della cartella **\$GIANO_ICS** si ramificano altre due sotto-directory:

- **IS1** (per il monitoraggio delle temperature e le comunicazioni con il PLC)
- **IS2** (per le interfacce con i motori criogenici).

Una terza sotto-directory, denominata IS3 era prevista all'inizio del progetto, ed avrebbe dovuto contenere i moduli dedicati alla gestione delle pre-ottiche dello strumento. Al momento pero' tale sotto-directory non esiste in quanto mancano completamente le specifiche (e l'hardware) per questo di tipo di interfaccia. Nell'immediato futuro, quando sara' chiara la struttura delle pre-ottiche di GIANO, si dovranno scrivere i relativi moduli di controllo nella sotto-directory IS3.

IS1

Nel ramo **IS1** sono presenti le seguenti sotto-cartelle:

- **bin** – Contiene tutti gli script eseguibili di IS1 e, per comodità di utilizzo, questa sottodirectory dovrebbe essere aggiunta alla variabile d'ambiente PATH (ad es. in *\$HOME/.cshrc*):

setenv PATH \$PATH":"\$GIANO_ICS/IS1/bin

Sul PC del laboratorio di Arcetri questo PATH è stato correttamente definito per cui è possibile lanciare, da linea di comando, un qualsiasi script di IS1 partendo da una qualunque directory.

Se si vuole di far partire il processo di comunicazione e di acquisizione automatica del solo PLC è sufficiente digitare:

IS1plc start &

mentre per fermare la medesima acquisizione:

IS1plc stop

Nel caso invece si voglia acquisire soltanto la telemetria dei 32 sensori di temperatura (dai 4 LakeShore) bisogna digitare:

IS1temp start &

e per terminare l'acquisizione:

IS1temp stop

Uno dei requirement di GIANO è di avere sempre in background attivi i due processi sopra citati. Per cui è stato creato un unico script incaricato di far partire/fermare i due processi contemporaneamente (cioè: IS1plc + IS1temp).

Per LANCIARE, in background, sia l'acquisizione dal PLC che l'acquisizione dei 32 sensori di Temperature è sufficiente digitare:

IS1start &

mentre per FERMARE/CHIUDERE i due processi digitare:

IS1stop

Per visualizzare su pannello (GUI) l'intera telemetria del PLC bisogna digitare:

monitorPLC &

Questo pannello si auto-aggiorna ogni 5 secondi (IS1plc deve essere in esecuzione) ed è suddiviso in 4 sezioni: 3 in sola lettura ed una in lettura + scrittura (sul frame avente per background il colore giallo). Di default tutti i bottoni "Edit", per la sola scrittura, sono disabilitati per motivi di sicurezza. Per poter cambiare/aggiornare uno di questi parametri è necessario inserire la password ("giano2007") nella finestrella "Login editor" in basso a sinistra del pannello. Al termine dell'aggiornamento di uno o più parametri uscire sempre dall'editor cliccando sul bottone "Logout editor".

Si raccomanda di non utilizzare questo editor se non si sa esattamente che cosa si sta facendo; inoltre va tenuto presente che l'aggiornamento di un parametro non è mai immediato in quando il meccanismo di scrittura va a sincronizzarsi sempre con i cicli di lettura, per cui solitamente bisogna attendere circa 5 secondi prima che il nuovo

valore di un dato parametro venga inviato al PLC (il nuovo valore sarà visibile sulla GUI al prossimo ciclo di lettura).

Per visualizzare su pannello i valori relativi ai 32 sensori di temperatura posizionati all'interno del criostato digitare:

monitorIS1 &

Questo pannello è stato concepito per la sola lettura e si auto-aggiorna ogni 5 secondi. Oltre alle temperature consente di visualizzare anche alcuni altri parametri principali del criostato e del Be-Tank di Giano. In basso alla GUI viene visualizzata la data e l'ora delle ultime letture utili effettuate.

Per visualizzare l'andamento temporale di una serie di parametri (ad es. gruppi di temperature, gruppi di pressioni ecc.) è possibile utilizzare il comando:

IS1_plot your_param_list &

dove *your_param_list* è un opportuno file di configurazione. In \$GIANO_ICS/IS1 ci sono alcuni esempi (commentati) di come questo file ASCII debba essere scritto. Per maggiori informazioni vedere ad esempio *temperature.plot*, *pressioni.plot*, *top.plot*, *arr.plot* ecc.

L'eseguibile *IS1_plot* consente di monitorare graficamente (plot) gruppi di parametri IS1 su finestre temporali di 1, 3 e 24 ore. Questo tool ha un limite massimo relativo alla telemetria delle ultime 24 ore (era un vecchio requirement) ed inoltre può essere utilizzato/visualizzato solo "localmente", cioè sul PC di controllo di Giano. Perciò per visualizzare su tempi più lunghi, e soprattutto per non limitarsi al solo uso locale, è preferibile utilizzare la pagina web: <http://tirgo.arcetri.astro.it/irlab/rrddaemon.php>.

- **conf** – Questa directory contiene tutti i file di configurazione per i moduli di IS1.

In ***IS1.conf*** sono memorizzate le seguenti definizioni: l'indirizzo IP della Lantronix, i numeri (porte della Lantronix) assegnati sia al PLC che alle 4 unità LakeShore. Inoltre, per ogni LakeShore, devono sempre essere assegnati anche i numeri dei sensori di temperatura effettivamente collegati/attivi.

In ***PLC_MODBUS.conf*** sono registrate tutte le informazioni delle variabili di telemetria relativa al solo PLC. Questo file non dovrebbe mai essere editato (a meno che, in futuro, non vengano aggiunti/attivati nuovi registri per il PLC). L'attuale versione di questo file si riferisce esattamente alle specifiche riportate nel documento *PLC_Modbus_V7.pdf* (rev.7 del 5/6/2009) presente nella cartella *doc* di IS1.

Il file ***WritePLC.txt*** NON è un file di configurazione ma è stato concepito, su esplicita richiesta del Project Manager, al fine di poter editare/scrivere direttamente sui registri del PLC di Giano (senza dover utilizzare l'interfaccia ***monitorPLC***). In tal modo è possibile editare al volo, anche da remoto, uno o più registri.

La situazione di default è che tutti i registri elencati in ***WritePLC.txt*** rimangano sempre commentati (segno di cancelletto “#” all'inizio di ogni riga). Se una o più righe vengono scomentate, e poi ***WritePLC.txt*** viene salvato, il daemon ***IS1plc*** andrà automaticamente a leggersi, per ogni riga scomentata, le coppie “Tag” “Valore” e provvederà a scriverne i valori sul PLC: questo fintato che la/e riga/righe

rimangono scommentate. L'utilizzo di questo file puo' quindi diventare molto pericoloso se ci si dimentica di ri-commentare una determinata riga, e di salvare nuovamente **WritePLC.txt**, una volta che un dato registro sia gia' stato aggiornato. Per questo motivo si sconsiglia ai non esperti l'utilizzo di questo file : per editare i registri del PLC utilizzare sempre l'interfaccia **monitorPLC**.

- **dat** – In questa directory vengono automaticamente scritti e aggiornati tutti i file contenenti i valori della telemetria di IS1. Sono di 3 tipi: **IS1plc.dat**, **IS1temp.dat** e **IS1plcWe.dat** ; i primi due sono file “a rotazione”, vengono cioe' mantenute sempre le ultime 8+1 versioni, con aggiornamenti ogni 24 ore.

In **IS1plc.dat** i valori, per ogni lettura, vengono sempre scritti su 42 colonne:

- 1) INTEGER: "date and time" [sec] (Unix epoch)
- 2) FLOAT: TE-301 Temperatura ambiente [K]
- 3) FLOAT: TE-302 Temperatura ambiente [K]
- 4) FLOAT: TAAV Temperatura media TE-301/302 [K]
- 5) FLOAT: TE-101 Temperatura Be-Tank [K]
- 6) FLOAT: TE-102 Temperatura Be-Tank [K]
- 7) FLOAT: TDAV Temperatura media TE-101/102 [K]
- 8) FLOAT: TE-103 Temperatura uscita gas N2 [K]
- 9) FLOAT: TE-104 Temperatura ingresso MFC [K]
- 10) FLOAT: PT-101 Pressione assoluta Be-Tank [mbar]
- 11) FLOAT: PT-201 Pressione assoluta pompa vuoto P-201 [mbar]
- 12) FLOAT: PT-202 Pressione assoluta criostato [mbar]
- 13) FLOAT: WT-100 Peso criostato [kg]
- 14) FLOAT: LV-101 Stato apertura valvola LN2 [%]
- 15) FLAOT: FCV-101_s Setpoint MFC [mc/h]
- 16) FLOAT: FCV-101_r Portata MFC [mc/h]
- 17) FLOAT: HE-101 Potenza riscaldatore elettrico [%]
- 18) FLOAT: TT-401 Temperatura scarico Cold_Trap [K]
- 19) FLOAT: DTHR Velocita' di raffr./riscald. [K/h]
- 20) FLOAT: MAXDTHR Massima velocita' raffr./riscald. [K/h]
- 21) FLOAT: Temperatura chiusura YV401 (Cold-Trap) [K]
- 22) FLOAT: MAPVLN2 Massima apertura LV-101 a caldo [%]
- 23) FLOAT: TDCRIT Soglia di temperatura limitatore LV-101 [K]
- 24) FLOAT: PCEMER Soglia di pressione avvio ciclo vuoto [mbar]
- 25) FLOAT: WEIGHT_MIN Valore minimo peso criostato [kg]
- 26) FLAOT: WEIGHT_MAX Valore massimo peso criostato [kg]
- 27) FLOAT: SET_PT101 Setpoint regolatore pressione Be-Tank [mbar]
- 28) FLOAT: SETHEPWR Set massima potenza riscaldatore elettrico [%]
- 29) FLOAT: PID_G Guadagno regolatore PID GN2
- 30) FLOAT: PID_TI Tempo integrale regolatore PID GN2 [min]
- 31) FLOAT: PID_TD Tempo derivata regolatore PID GN2 [min]
- 32) FLOAT: DT_MAX Tempo massimo tra riempimenti [sec]
- 33) FLOAT: TOPEN_MAX Tempo massimo di apertura LV-101 [sec]
- 34) FLOAT: TIMELN2V Tempo di messa a freddo linea LN2 [sec]
- 35) FLOAT: MAXVLN2 Massima apertura LV101 a freddo [%]
- 36) FLOAT: TIMEYV401 Tempo minimo tra 2 aperture YV401 [sec]

- 37) 16 bits (reg. 30070)
- 38) 16 bits (reg. 30071)
- 39) 16 bits (reg. 30072)
- 40) 16 bits (reg. 30073)
- 41) FLOAT: LIMVLN2 Limitatore valvola LV101 [%]
- 42) FLOAT: LIMHE101 Limitatore riscaldatore HE101 [%]

Le colonne da 37 a 40 contengono, ognuna, una stringa di 16 bit (*True/False*) relativi a valvole/accensioni/guasti ecc. (vedi *PLC_Modbus_V7.pdf*, nella cartella *doc* di IS1).

In *IS1temp.dat*, per ogni lettura, i valori vengono scritti su 33 colonne, di cui la prima e' il solito INTEGER "date and time" [sec] (Unix epoch), mentre le rimanenti 32 colonne contengono i valori delle 32 temperature cosi' come sono state definite ed ordinate nel file di configurazione *\$GIANO_ICS/IS1/conf/IS1.conf*.

In *IS1plcWe.dat* sono contenuti i valori dei registri relativi al peso del criostato ad inizio ed a fine riempimento (in Kg.), al tempo (in sec.) trascorso da un precedente riempimento ed al tempo di riempimento (in sec).

- **doc** – Contiene l'importante file di documentazione *PLC_Modbus_V7.pdf* e, per motivi storici, le sue precedenti versioni (anche se ormai obsolete). Inoltre contiene un breve **ChangeLog** in cui sono stati riportati i principali cambiamenti/aggiornamenti avvenuti nel corso del tempo su IS1.
- **lib** – Necessaria per la libreria grafica BLT (*libBLT24.so*), ed utilizzata da *IS1_plot*. Questa versione funziona solo su PC-Linux a 32 bit (in futuro, per versioni a 64 bit, sara' necessario procedere alla sostituzione con la libreria corretta).
- **log** – E' una directory utile nelle fasi di debugging e/o per risalire a cause di eventuali malfunzionamenti. I file sono di tre tipi: *IS1plc.log*, *IS1temp.log*, *error.log*. I primi due sono file "a rotazione", e vengono mantenute sempre le ultime 6+1 versioni, con aggiornamenti ogni 24 ore. Il terzo file viene utilizzato solo per l'eventuale raccolta di informazioni provenienti dallo *stderr*.
- **src** – Contiene alcuni file/libreria (metodo "source") utilizzati dagli script eseguibili presenti nella directory *bin*.
- **tmp** – E' solo una directory temporanea dove viene creata/distrutta la *pipe* di IS1.

IS2

Il modulo IS2 e' stato progettato e realizzato specificatamente per interfacciarsi con i motori criogenici di Giano. Nella directory principale di IS2 sono presenti glii script eseguibili, e questa directory dovrebbe essere aggiunta alla variabile d'ambiente PATH:

setenv PATH \$PATH":"\$GIANO_ICS/IS2

Sul PC del laboratorio di Arcetri questo PATH e' stato correttamente definito per cui e' possibile lanciare, da linea di comando, un qualsiasi script di IS2 partendo da una qualunque directory. Per muovere un motore o un gruppo di motori si utilizza il comando:

gianoMotors &

Sul desktop del PC di controllo apparira' una pannello (e' un'interfaccia di "medio livello") che contiene (nella parte bassa della GUI) un pulsante di "STARTUP". Prima di lanciare lo startup puo' essere utile configurare la modalita' di lavoro tramite il bottone "Options". Quest'ultimo consente di inizializzare o meno tutti i motori al momento dello startup e/o di mantenerli tutti accesi (power ON) o meno durante tutta la sessione di lavoro.

Il pulsante di "STARTUP", una volta attivato, lancia automaticamente in background la procedura ***IS2encoder*** incaricata della lettura delle posizioni (in *ohm*) di tutti i potenziometri collegati/solidali con i motori (il risultato verra' poi riportato nella colonna "Pos Ω ").

Il programma ***gianoMotors*** consente di muovere ognuno dei 9 motori di Giano in due modalita': o selezionando dal menu "a tendina" una delle posizioni gia' codificate oppure scrivendo direttamente nella colonna "New Ω " il valore di destinazione. Una volta scelta una nuova posizione sara' sufficiente cliccare il pulsante "START" (del relativo motore) per far partire la movimentazione (il pulsante "STOP" interrompera' il movimento).

Nella directory IS2 sono presenti altri due tool: ***gld*** e ***gld_ohm.tcl***. Entrambi devono essere avviati indicando sempre il numero del motore da utilizzare (ad es.: "***gld_ohm.tcl 5***" per utilizzare il motore n.5). I due programmi non sono altro che interfacce di "basso livello", progettate e scritte in una fase del progetto in cui necessitavano sia test che "messe a punto" di basso livello su singoli motori. Questi due tool consentono anche di inviare comandi "manuali" direttamente ad un *GLD stepper motor controller* (per la corretta sintassi dei comandi riferirsi sempre al manuale ***\$GIANO_ICS/IS2/doc/ipcomm.pdf***).

La differenza tra i due tool e' che il primo, ***gld***, visualizza e conteggia solo le posizioni in unita' di step (veniva utilizzato quando non erano stati ancora implementati i potenziometri), mentre il secondo e' "spazialmente" piu' affidabile in quando gestisce anche le "true position in *ohm*."

Inoltre ***gld*** (a diffenza di ***gld_ohm.tcl***) consente anche di inviare sequenze di comandi di basso livello ad un determinato controller. Il meccanismo di caricamento di queste sequenze avviene inserendo direttamente il nome del file ".seq" nella casella "Input file" (oppure selezionando il file tramite il pulsante "select"). Un esempio di tipiche sequenze di comandi di basso livello si trova in ***\$GIANO_ICS/IS2/testExample.seq***.

Infine ***gld*** accetta anche file con estensioni ".enc" per eseguire sequenze di test in *batch mode*

(un esempio di file *batch* si trova in ***encoderPositionTest.enc***).

In IS2 e' presente anche l'importantissimo file di configurazione ***GLD.conf*** , necessario per il corretto funzionamento di tutti i motori criogenici. Esso e' suddiviso in 3 sezioni contenenti:

- ***ADDRESS directives***. Qui vengono assegnati i numeri IP delle Lantronix (previste piu' di una) collegate ai controller GLD dei motori, alle LakeShore dei potenziometri ed ai *power switch* dei motori. Inoltre vengono definite le porte, sulle Lantronix, per il *power switch* dei motori, per il *powering* dei motori per il gruppo n.1 e per il gruppo n.2 .
- ***POWER SUPPLY directives***. Servono per assegnare/definire i comandi da inviare al *PowerControl* per l'accensione o lo spegnimento delle n.2 LakeShore (potenziometri), delle n.4 LakeShore (sensori di temperatura), del PC104, dell'elettronica (detector), del *powering* dei motori gruppo 1 e gruppo 2.
- ***MOTORS directives***. In questa sezione vengono assegnati, per ogni motore, i numeri progressivi, i numeri delle porte (GLD e LakeShore), i canali di input (LakeShore), i nomi dei motori (keyword), i comandi per l'accensione/spegnimento.

Nel ramo IS2 sono presenti le seguenti sotto-cartelle:

- **config** – Contiene 9 file di configurazione, specifici per ogni singolo motore:
ArrayZ1.pos, ArrayZ2.pos, ArrayZ3.pos, GratingZ1.pos, GratingZ2.pos, Image.pos, FilterWheel.pos, SlitFork.pos e SlitWheel.pos .
Questi file ASCII sono suddivisi in due sezioni: la prima codifica l'elenco delle posizioni (nome/keyword e corrispondente valore in ohm), la seconda i comandi da inviare al controller GLD al momento dell'accensione del singolo motore (sequenza di inizializzazione).
Nella prima sezione sono presenti due colonne per definire i valori delle posizioni in ohm: rispettivamente quando lo strumento e' *warm* oppure *cold*. Al momento pero' solo i valori scritti nella prima colonna vengono letti ed accettati dal programma ***gianoMotors***, dato che l'implementazione del meccanismo di riconoscimento dello stato reale dello strumento (cioe': *cold / middle / warm / unknown*) non e' ancora stata attivata.
- **doc** – Contiene il manuale ***ipcomm.pdf*** (della *phytron*) relativo alle specifiche del protocollo di comunicazione *IPCOMM*. Inoltre e' presente un breve ***ChangeLog*** in cui sono stati riportati i principali cambiamenti/aggiornamenti avvenuti nel corso del tempo sul modulo IS2.
- **log** – E' una directory utile nelle fasi di debugging per risalire alle cause di eventuali errori e/o malfunzionamenti. I file prodotti da ***gianoMotors*** sono di tre tipi: ***gianoMotors.log, IS2encoder.log, IS2error.log***. I primi due sono file "a rotazione", e vengono mantenute sempre le ultime 4+1 versioni, con aggiornamenti ogni 48 ore. Il terzo file viene utilizzato solo per l'eventuale raccolta di informazioni provenienti dallo *stderr*.
In questa directory vengono anche riportati i file ***Motor_n.log*** (con n da 1 a 9) relativi

all'utilizzo dei programmi (di basso livello) *gld* e *gld_ohm.tcl*.

- **src** – Contiene alcuni file/libreria (metodo “source”) utilizzati dagli script eseguibili presenti nella directory **\$GIANO_IC/IS2**.
Per motivi storici e' presente anche un vecchio eseguibile (di basso livello) *gld_v07_Serial.tcl* utilizzato in passato per la comunicazione seriale diretta con un singolo motore (l'utilizzo dei potenziometri non era previsto).

Note sulla GUI di OS (Observation Software).

Il pannello principale di acquisizione dati (OS) di Giano prevede, tra le molte altre cose, la gestione dei motori, per il settaggio delle slit, dei filtri e del grating. La comunicazione con questi motori puo' essere fatta utilizzando (in background ed in modalita' nascosta) il programma *gianoMotors*. Inviando a questo programma opportune stringhe ASCII (da una shell *tclsh* o *wish*) e' possibile effettuare la comunicazione con uno o piu' motori. Ad esempio per inizializzare il motore n.5 sara' sufficiente inviare a *gianoMotors* la stringa “*initMotor M5*”, per muovere il motore n.1 alla posizione 4600 inviare “*startMotion M1 .f1 4600*” ecc. Guardando all'interno dello script *gianoMotors* (sufficientemente commentato in inglese) e' possibile recuperare quei (pochi) comandi necessari per poter accendere, spegnere, inizializzare e muovere un motore da un'applicazione esterna. In ogni caso nella directory *log* sara' sempre possibile (eventualmente) leggere e recuperare utili informazioni.