

Software di Controllo per  
ARcetri CORrelation Spectrometer

Programma di Prova per  
la lettura dei total power dei campionatori

F. Palagi<sup>1</sup> e G. Comoretto <sup>2</sup>

<sup>1</sup> IRa - CNR, Sezione di Firenze, l.go E. Fermi 5, 50125 Firenze (Italy)

<sup>2</sup> Osservatorio Astrofisico di Arcetri, l.go E. Fermi 5, 50125 Firenze (Italy)

**Arcetri Technical Report N° 2/2002**  
**Florence January 2002**

## Sommario

Il programma di controllo dello spettrometro *ARcetri COrrelation Spectrometer*, *ARCOS*, `adlb_serv`, è stato modificato per utilizzare i misuratori di *total power* interni ai compionatori. le modifiche sono descritte in un precedente rapporto tecnico dell'Osservatorio di Arcetri.

In questo rapporto si descrive un banco di prova e il relativo programma di misura, sviluppato per verificare il corretto funzionamento delle modifiche introdotte in `adlb_serv` e delle routines di misura da i inserire nel programma esterno `adlb_arc`, che costituisce l'interfaccia con l'osservatore.

Come nel caso delle modifiche ad `adlb_serv`, anche in questo casi la progettazione segue una metodologia orientata agli oggetti, mentre la realizzazione e' fatta in linguaggio C.

# 1 Descrizione del programma

Il programma di prova `testtpw` è stato sviluppato inizialmente solo per verificare il corretto funzionamento della funzione `TpArcosRead`. Successivamente si è manifestata l'esigenza di creare un programma di uso più generale, in grado di misurare le caratteristiche di linearità e stabilità dei rivelatori di total power di ARCOS.

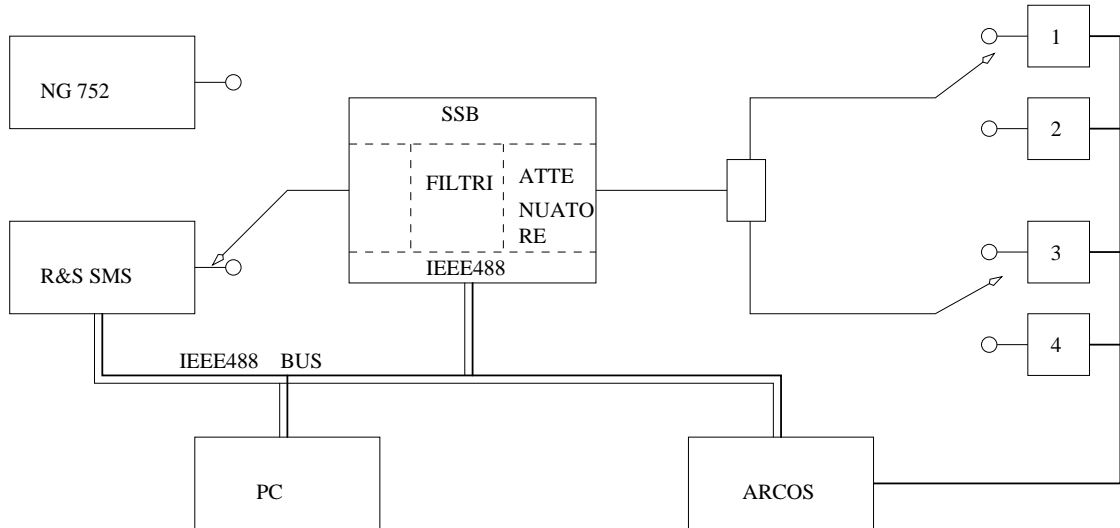


Figura 1: Schema del banco di misura per i lettori di total power di ARCOS

Il banco di misura (Fig. 1) è composto da:

1. un generatore di rumore
2. un generatore di segnale Rhode & Schwarz modello SMS.
3. una convertitore SSB sviluppato ad Arcetri, e caratterizzato da:
  - un selettore del filtro di uscita 1/10 con bande comprese fra 40 Mhz e 0.16 MHz.
  - di un attenuatore variabile che puo' assumere valori compresi fra 0 dB e 31.5 db
  - frequenzadi uscita compresa fra 500 e 900 MHz
4. un partitore del segnale prodotto, per alimentare contemporaneamente due campionatori.
5. un PC di controllo.
6. Il correlatore ARCOS.

Il controllo delle periferiche avviene attraverso la linea parallela IEEE488.

La tabella seguente riassume i parametri di ciascun dispositivo, i cui valori devono essere specificati prima di iniziare una misura.

Un ciclo di misura consiste di uno o più comandi necessari per configurare il banco di misura, seguiti dal comando MISURA. Le misure possono essere eseguite sia in modo

Tabella 1: Parametri da specificare per ciascun dispositivo del banco di prova

Dispositivo	Identificatore	Indirizzo IEE488	Parametri
R&S	RS_SMS	14	Frequenza
Generatore di rumore	—	Livello di uscita —	—
SSB	SSB	5	Banda passante Attenuazione Frequenza di uscita
ARCOS	ARCOS	13	Sezioni Tempo di integrazione

*interattivo* che in modo *batch*. Il modo *interattivo* è quella di default a cui si predispongono il programma alla partenza. Esso prevede che i comandi siano dati da tastiera e può essere riattivato dal comando KEYBOARD. Nel modo *batch* i comandi sono letti da un file. Esso è attivato dal comando BATCH che ha come parametro il nome del file da leggere. Il comando KEYBOARD inserito in un file di comandi provoca il ritorno del controllo in modo interattivo.

Per assegnare un valore ai parametri di configurazione dei vari dispositivi del banco di prova si usano comandi formati da:

- Identificatore del dispositivo.
- Stringa dei parametri che rispetta un formato predefinito.

Il comando MISURA esegue il numero di misure specificato.

## 2 Descrizione dei comandi

Segue una descrizione dettagliata dei comandi con sintassi e esempi.

### 2.1 SSB

#### Sintassi

`SSB < freq > < banda > < attenuazione >`

Si utilizza per assegnare i valori ai parametri del filtro SSB

#### Esempio

`SSB 600.0 10.0 31.5`

Questo esempio produce un segnale di uscita con una frequenza di 600 MHz, una banda passante di 10 MHz e un'attenuazione di 31.5 dB.

## 2.2 ARCOS

### Sintassi

ARCOS < *tempo\_integ* >< *Campionatori* >

Si utilizza per specificare i campionatori usati. La posizione del numero indica la sezione a cui viene associato il campionatore corrispondente. I campionatori sono numerati da 1 a 8. Il tempo di integrazione è espresso in secondi.

### Esempio

ARCOS 10.0 13

Questo esempio assegna il campionatore 1 alla prima sezione e il campionatore 3 alla seconda, con un tempo di integrazione di 10 secondi.

## 2.3 RS\_SMS

### Sintassi

RS\_SMS < *frequenza* >< *ampiezza* >

Specifica il valori del segnale prodotto dal generatore Rhode & Schwartz. La frequenza è spressa im MHz e l'ampiezza in dBm.

### Esempio

RS\_SMS 8.5 -23.0

Questo esempio genera un segnale a 8.5 MHz con una potenza di -23.0 dBm.

## 2.4 MISURA

### Sintassi

MISURA < *numero\_di\_misure* >

Esegue il numero di misure specificato.

### Esempio

MISURA 10

Questo esempio esegue 10 misure nella configurazione corrente.

## 2.5 BATCH

### Sintassi

BATCH < *nome\_file* >

Passa il controllo della misura al file di comandi indicato. Il controllo ritorna alla tastiera se si incontra il comando KEYBOARD oppure alla fine del file.

### Esempio

BATCH 13 10.0

## 2.6 KEYBOARD

### Sintassi

KEYBOARD

Trasferisce l'input dei comandi alla tastiera del PC.

### Esempio

KEYBOARD

## 2.7 HEADER

### Sintassi

HEADER

Stampa i valori attuali dei parametri di misura nel file di uscita.

### Esempio

HEADER

## 3 Il progetto

Il diagramma delle classi di questo programma è rappresentato in Fig. 2.

### 3.1 L'interprete dei comandi

Il nucleo centrale della struttura è rappresentato dall'interprete dei comandi. Questo oggetto ha il compito di verificare la correttezza dei comandi letti e di eseguire la funzione corrispondente. A questo scopo esso contiene una tabella che associa ad ogni identificatore di dispositivo una funzione, come nell'oggetto `CommanInterpreter` di `adlb_serv`. La funzione da eseguire è fornita da ciascun oggetto di classe *dispositivo*.

L'interprete tiene traccia del modo di funzionamento (interattivo o batch) e delega la lettura dei comandi ad un oggetto di classe `Usercmd`. Al termine della misura, l'interprete ha anche il compito di mostrare i risultati sullo schermo e scriverli nel file di uscita.

L'interprete dei comandi ha il compito di tenere traccia di quale è la sorgente attiva dei comandi, attraverso l'esecuzione dei comandi BATCH e KEYBOARD.

#### 3.1.1 Lettura dei comandi

Questa parte costituisce il principale loop di controllo del programma. Essa si svolge secondo lo schema seguente:

1. Se il modo corrente è interattivo, il comando è letto da tastiera.
2. Se il modo corrente è batch, il comando è letto dal file di ingresso.
3. Se nessuno dei modi è definito emette un messaggio di errore.
4. Se il comando esiste, esegue la funzione associata.

### 3.2 La classe dispositivo

Gli oggetti di questa classe hanno essenzialmente il compito di fornire la funzione associata all'identificatore presente nella tabella dei comandi. Nella maggior parte dei casi essa assegna il valore passato come argomento, ai propri attributi. Questa operazione prevede anche dei controlli di validità dei valori da assegnare.

Quando il modo è interattivo, questi oggetti forniscono la possibilità di aggiustare i parametri attraverso un semplice (e per il momento grezzo) menu' di comandi aggiuntivi, gestito direttamente dal dispositivo interessato. Per il momento, questa possibilità è limitata al filtro SSB.

Le proprietà della classe disposito sono ereditate dagli oggetti SSB, R&S e ARCOS che ha anche il compito di eseguire la misura.

### 3.2.1 Assegnazione dei parametri

Gli attributi dei dispositivi sono definiti in una struttura simile a quella usata in ADLB4ARC, per ricreare per quanto possibile lo stesso ambiente e, quindi, poter riutilizzare le funzioni sviluppate per il programma di prova, nel programma ADLB4ARC.

In particolare:

- Si usa la struttura `param`, di tipo `aa`, contiene un array di strutture `sect` (tipo `SECT`) e una struttura `scan` (tipo `SCAN`) con l'aggiunta di due elementi: il tempo di integrazione (`total power`) e il numero di misure da eseguire.
- La struttura di tipo `SCAN` contiene solo il numero di sezioni del correlatore.
- La struttura di tipo `SECT` contiene tre elementi: il tipo di contatore, il suo indirizzo e il risultato della misura.

L'assegnazione dei parametri segue lo schema seguente:

1. Si decodificano i parametri.
2. Si verifica la validità del loro valore. Se il valore è accettabile esso viene assegnato all'attributo corrispondente al parametro in questione.

### 3.3 Lettura dei comandi

La lettura dei comandi è eseguita dall'oggetto `input` di classe `Usercmd` attraverso le operazioni `get_a_command`, `read_a_command` per il modo interattivo e batch rispettivamente.

Il comando acquisito è di tipo `Usercmd` e viene memorizzato in una struttura formata da due stringhe: la prima che contiene l'identificatore del dispositivo e la seconda che contiene i valori da assegnare agli attributi.

Le due funzioni restituiscono un oggetto di classe `Usercmd` che viene passato all'interprete dei comandi per la sua verifica ed esecuzione.

La classe `Usercmd` fornisce alcune operazioni di servizio:

**`clear_cmd`** Azzera gli attributi della classe.

**`assign_cmd`** decodifica la stringa letta e separa l'identificatore dall'argomento.

### 3.4 La verifica ed esecuzione dei comandi

Questa operazione compete all'interprete dei comandi. Essa è eseguita dalla funzione `search_command`. Essa scorre la tabella dei comandi cercando l'identificatore ricevuto da `input`; se la ricerca ha successo, la funzione restituisce l'indice della tabella e l'interprete è in grado di eseguire la funzione corrispondente assegnando la stringa dei parametri all'argomento della funzione.

La tabella dei comandi è terminata da un identificatore costituito da una stringa vuota e una funzione il cui indirizzo è `NULL`. In questo modo è possibile modificare la tabella senza dover tener conto esplicitamente delle sue dimensioni.

## 4 Implementazione

Il programma `testtpw` risiede nella directory `\adlb4arc\source\`).

Le strutture dei dati sono definite nel file `totalpow.h` del quale si riporta il frammento corrispondente.

```
typedef struct {
    float lev;          /* Output Noise Level          */
    float att;          /* Attenuator value            */
    float att_min, att_max;
    int ssb;            /* Sideband allowed:0=none,1=up,2=low,3=both */
    int ifchan;         /* IF channel number (1 or 2)  */
    int type;           /* hardware syntetizer model   */
    int addr;           /* hardware address (IEEE488 addr, VC num. ... */
    int tp_type;        /* Total power type            */
    int tp_addr;        /* total power address         */
/* private variables */
    double sfreq;       /* Set frequency                */
    double fmin, fmax;
    int sssb;          /* Set sideband: +1=upper, -1=lower */
} L02;

typedef struct {
    int tp_addr;
    int tp_type;
    float counton;
        int bandcode;
} SECT;

typedef struct {
    int nsections;
} SCAN;

struct aa {
    int nmeas;
    float tp_integ_time;
    SECT sect[MAXMODULE];
    SCAN scan;
        L02 arcssb;
        L02 rs_sms;
} ;
```

Il `CommandInterpreter` contiene una tabella di comandi, che è costituita da un array di strutture di tipo `cmdtab` formate da una stringa, che contiene l'identificatore del dispositivo, e un puntatore alla funzione associata all'identificatore.



## 5 Manuale d'uso

Le parole chiave indicano:

MISURA parametri usati per l'esecuzione della misura:

- tempo di integrazione [sec].
- campionatori usati [1-8].
- numero di misure richieste

SSB parametri usati per la configurazione dei filtri SSB:

- frequenza L.O. [Mhz].
- Banda passante [MHz].
- Attenuazione [dB].

RS\_SMS parametri usati per la configurazione dei filtri SSB:

- frequenza generatore [Mhz].
- Livello del segnale [dBm].

Dal terminale di ARCOS deve essere attivato il programma `adlb_serv`.

La riga di comando deve contenere il nome del file di inizializzazione come parametro:

Per esempio, se il file di inizializzazione si chiama `testtpw.ini`, si dà il comando:

```
testtpw testtpw.ini
```

La prima operazione del programma consiste nel configurare la strumentazione secondo i parametri indicati nel file di inizializzazione. Per quanto riguarda il livello di attenuazione del segnale in uscita dagli SSB e la sua banda passante, e' data la possibilita' di variare i valori impostati inizialmente, prima di effettuare la misura. L'interazione avviene attraverso un semplice menu di comandi.

I risultati sono stampati sullo schermo in forma di tabella che riporta:

1. il numero della sezione
2. il numero del campionatore
3. il valore del total power
4. l'errore (rms) della misura

e sono anche memorizzati nel file `testtpw.dat` in un formato compatibile con `smongo` (v. appendice 5).

Al termine di ciascuna misura viene riportato il numero delle misure che devono essere ancora eseguite.

## A Esempio di file di uscita

```
# Counter Tot.Pow. Error (rep.ed 2 sect.s)
1 1.7240e+004 1.2020e+002 3 1.6973e+004 1.6355e+002
1 1.7240e+004 1.8923e+002 3 1.6979e+004 2.3456e+002
1 1.7246e+004 2.5695e+002 3 1.6998e+004 2.9324e+002
1 1.7238e+004 3.2619e+002 3 1.6973e+004 3.8538e+002
1 1.7225e+004 3.9466e+002 3 1.6948e+004 4.5273e+002
1 1.7222e+004 4.6416e+002 3 1.6950e+004 5.5042e+002
1 1.7218e+004 5.3323e+002 3 1.6941e+004 6.4478e+002
1 1.7213e+004 6.0174e+002 3 1.6945e+004 7.3530e+002
1 1.7262e+004 2.8946e+001 3 1.7007e+004 6.1956e+001
1 1.7244e+004 9.7685e+001 3 1.6973e+004 1.2824e+002
```

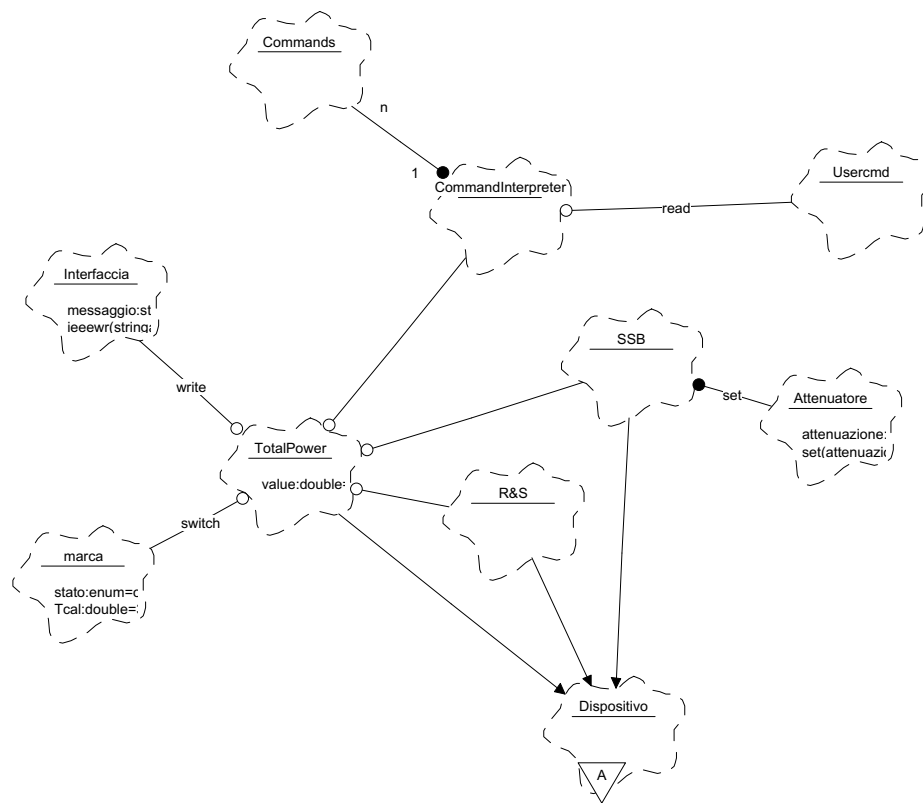


Figura 2: Diagramma delle classi del programma di prova dei misuratori di *total power*.