

INTERNAL REPORT

Sistema di controllo per la caratterizzazione dell'oscillatore locale ALMA impiegato nel ricevitore 100GHz del Sardinia Radio Telescope

Adelaide Ladu, Pierluigi Ortu, Andrea Saba, Mauro Pili, Tonino Pisanu,
Alessandro Navarrini

REPORT N. 73

RELEASED: 28/08/2018

Revisore: Raimondo Concu



Osservatorio
Astronomico
di Cagliari

TECHNICAL REPORT

Caratterizzazione dell'oscillatore locale ALMA per il ricevitore 100 GHz del Sardinia Radio Telescope

SOMMARIO

1. INTRODUZIONE.....	2
2. SISTEMA DI CONTROLLO	4
3. OSCILLATORE LOCALE	6
4. PROGETTAZIONE E REALIZZAZIONE DELL'HARDWARE.....	8
4.1 ARCHITETTURA E SCHEMA A BLOCCHI.....	8
4.2 SCHEDA AMC	9
4.3 SCHEDA PA.	11
4.4 MODULO BUFFER.	13
5. SOFTWARE E FIRMWARE DELLE SCHEDE DI CONTROLLO.....	15
5.1 INTRODUZIONE.....	15
5.2 INPUT ATTESI DALLE SCHEDE ARDUINO	15
5.3 OUTPUT DELLE SCHEDE ARDUINO	16
5.4 FIRMWARE DELLE SCHEDE AMC-PA	16
5.5 SOFTWARE DI TEST DELLE SCHEDE ARDUINO	17
5.6 SOFTWARE DI CONTROLLO DELLA RASPBERRY	17
5.7 LINGUAGGIO DEI COMANDI	19
6. TEST FINALI	20
7. CONCLUSIONI	22
8. BIBLIOGRAFIA	23

1. INTRODUZIONE

Lo scopo di questo rapporto interno è quello di presentare lo status dello sviluppo del ricevitore criogenico a 3mm, mostrando un'importante modifica fatta al progetto originale di questo ricevitore allo scopo di consentirne l'integrazione nel Sardinia Radio Telescope [1]. Lo strumento, infatti, è un ricevitore di vecchia generazione, progettato e realizzato all'IRAM (Institut de Radioastronomie Millimétrique), impiegato in una delle antenne del Plateau de Bure Interferometer (PdBI) fino al 2006. Dopo la sua dismissione, l'Osservatorio Astronomico di Cagliari l'ha acquistato con lo scopo sia di testare le performance della superficie attiva di SRT alla sua frequenza di lavoro più alta, intorno ai 100 GHz, sia di eseguire le prime osservazioni astronomiche in modalità single-dish e mm-VLBI.

Verranno discusse le caratteristiche dello strumento ed in particolare quelle del nuovo Oscillatore Locale (OL) controllabile elettronicamente, di tipo ALMA Banda 3 (Atacama Large Millimeter Array), che è stato riadattato per l'utilizzo con il ricevitore 3 mm ad SRT. Forniremo i risultati sperimentali preliminari di temperatura di rumore ottenuti con un sistema criogenico non ottimizzato.

Il ricevitore era stato progettato ad IRAM per coprire contemporaneamente due bande di frequenza differenti: la banda 3mm (84-116 GHz) e la banda 1.3mm (210-248 GHz) [2-4]. Una griglia di polarizzazione a banda larga posta all'ingresso del ricevitore divideva due segnali ortogonali e linearmente polarizzati nella banda 84-248 GHz in due percorsi ottici differenti, ciascuno dei quali era ricevuto in una delle due bande. Tuttavia, solo la polarizzazione nella banda a 3mm può essere osservata con SRT, mentre il canale a 1.3mm rimane inutilizzato. Ciascun canale è basato su un mixer a superconduttore a banda semplice basato su backshort meccanico in cavità guida d'onda "*backshort-tuned Single Side Band(SSB) SIS (Superconductor-Insulator-Superconductor) mixer*" che lavora alla temperatura fisica di 4.2K, con performances eccellenti in termini di figura di rumore. La temperatura criogenica del mixer di 4.2K è fornita da un piatto di 10 pollici (25.4 cm) di diametro fatto con un materiale ad alta conducibilità termica originariamente in contatto termico con un serbatoio di elio liquido. Questo serbatoio era situato all'interno di un criostato ibrido (HDV10 prodotto da Infrared Laboratories) con tre stadi criogenici, dove un dito freddo CTI 350CP forniva le temperature dei due stadi aggiuntivi a ~20K e a ~70K.

Tenendo conto delle specifiche relative alla configurazione ottica di questo ricevitore e delle relative limitazioni meccaniche e di tutte le problematiche che comporta la gestione di un ricevitore che usa elio liquido ad SRT [5], si è investigata una soluzione alternativa per raffreddare il criostato. La scelta è ricaduta sull'impiego di un nuovo criogeneratore realizzato ad hoc per questo ricevitore dall'azienda Advanced Research Systems (ARS), modello DE-204SF. Tale criogeneratore fornisce uno stadio a ~20K e uno stadio a 4K con una potenza di raffreddamento rispettivamente di 8W e 0.2W e non richiede l'uso di elio liquido.

Il segnale IF (Frequenza Intermedia) in uscita dal ricevitore ha una banda istantanea di 500MHz (1.3–1.8 GHz) in entrambi i canali riceventi. Il ricevitore è inoltre dotato di due carichi di calibrazione, uno per la temperatura ambiente ed uno per la temperatura criogenica, sia per il canale ricevente a 3mm che per il canale a 1.3mm.

Nel seguente documento vi descriveremo le modifiche apportate al sistema di controllo con un particolare riferimento al sistema di gestione dell'oscillatore locale.

2. SISTEMA DI CONTROLLO

Il ricevitore era stato progettato per le antenne del PdBI e le schede di controllo fornite insieme allo strumento, che servono a monitorare e controllare il ricevitore stesso, sono interamente senza intelligenza *on-board*. Queste risultano inutilizzabili su SRT. Per questa ragione, è stato sviluppato un nuovo sistema di controllo che dotasse tutte le schede di un'intelligenza *on-board*, consentendoci così di sintonizzare e controllare da remoto il ricevitore, insieme a un'interfaccia software che consentisse l'integrazione del sistema di controllo dello strumento con il software di controllo di SRT.

Il ricevitore è dotato di tre schede di controllo progettate e realizzate dall'IRAM: due servono a controllare il *backshort-tuned* SIS mixer e una a controllare il sistema di calibrazione. Ad esse, occorre aggiungere la scheda di controllo per l'oscillatore locale elettronicamente sintonizzabile della banda 3 di ALMA acquistato da NRAO (National Radio Astronomy Observatory).

L'architettura sviluppata per il controllo delle schede sopra elencate è innovativa: un *single-board* computer Raspberry Pi 2 (Model B) [6] controlla tre diversi microcontrollori Arduino UNO REV3 (ATMEL ATmega 328P) [7], uno per ciascuna scheda di controllo del ricevitore (come è mostrato in Figura 2-1), e un programma Python abilita le comunicazioni tra il ricevitore e il software di controllo di SRT [8].

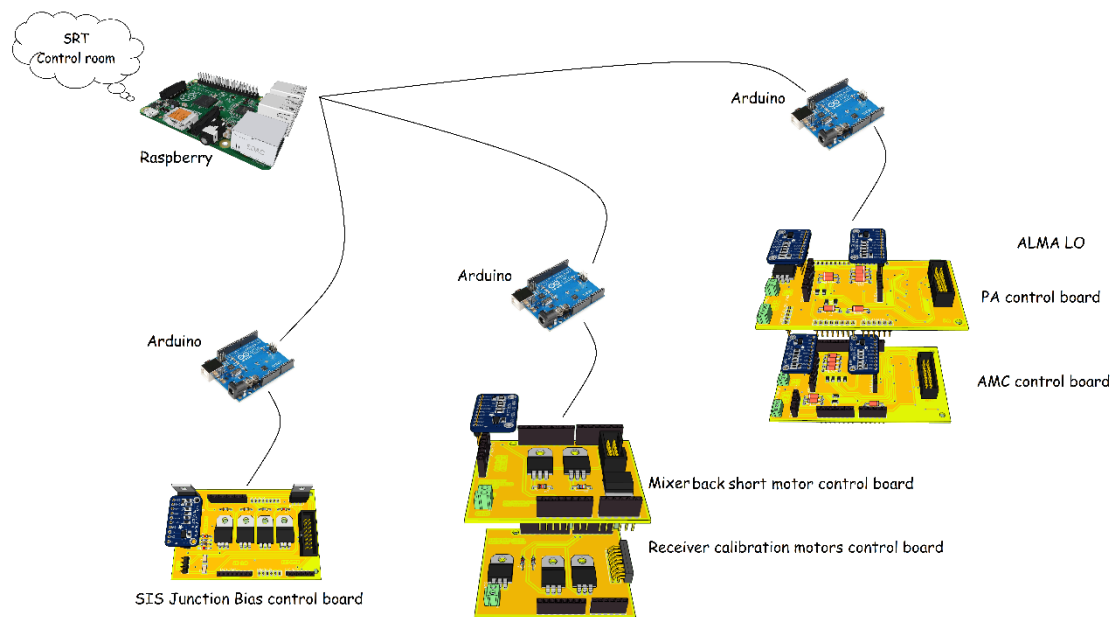


FIGURA 2-1: ARCHITETTURA DEL SISTEMA DI CONTROLLO DEL RICEVITORE 3MM

Particolare attenzione è stata data alle interferenze (Radio Frequency Interferences - RFI) auto generate da Raspberry e Arduino. Per questa ragione tutte le schede di controllo sono state rinchiusi in box di alluminio al fine di isolare questi potenziali segnali indesiderati, come si può vedere anche in Figura 2-2.

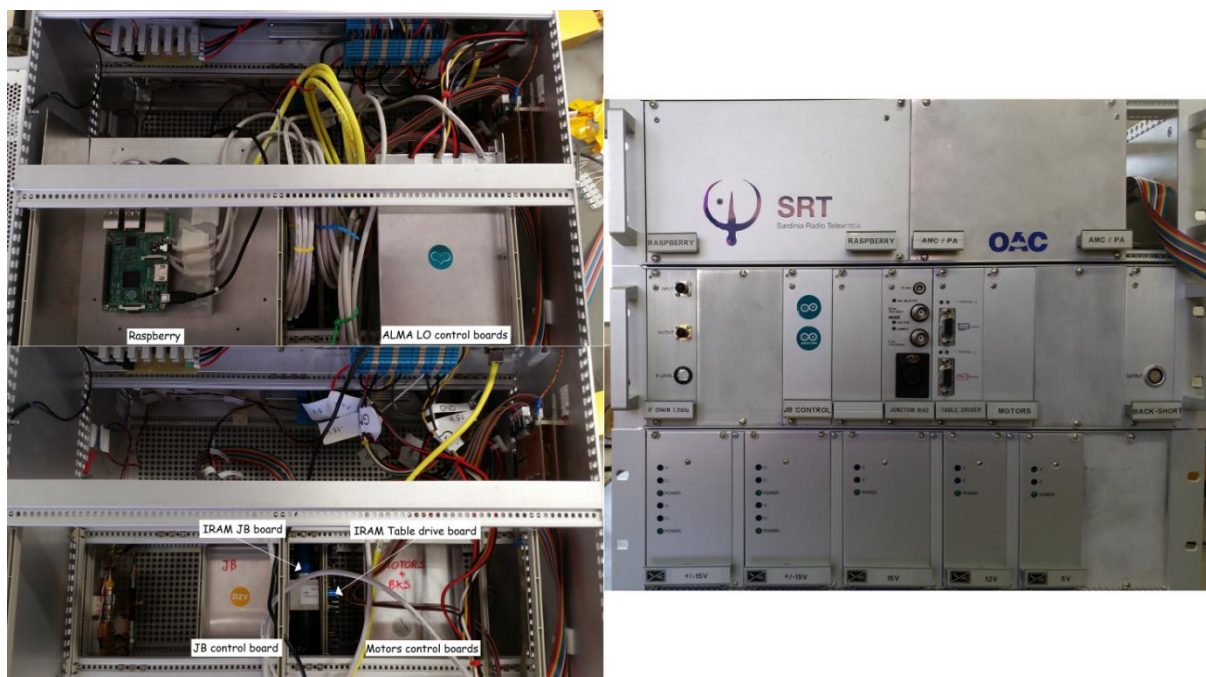


FIGURA 2-2: RACK ASSEMBLATO DEL SISTEMA DI CONTROLLO DEL RICEVITORE 3MM.

Il rack integra il single-board computer Raspberry, la scheda di controllo dell'Oscillatore Locale ALMA, la scheda di controllo del mixer SIS, la scheda di controllo dei motori del bachshort del mixer e dei motori di calibrazione e le schede prodotte dall'IRAM.

Nei capitoli successivi verranno descritte nel dettaglio le schede di controllo dell'Oscillatore Locale ALMA, il firmware sviluppato per controllare da remoto lo strumento e i risultati dei test effettuati.

3. OSCILLATORE LOCALE

L'oscillatore locale sviluppato dal NRAO per la banda 3 mm di ALMA [9] - [10] (ALMA Banda 3) è stato utilizzato per pompare il mixer SIS del nostro ricevitore 3mm. Tale oscillatore sostituisce quello di vecchia generazione impiegato dall'IRAM nei telescopi del PdBI, che era basato sul tradizionale diodo Gunn.

Possiamo suddividere l'oscillatore ALMA in due blocchi: 1) un'Active Multiplier Chain (AMC); 2) un Power Amplifier (PA), posto in cascata con l'AMC. In ingresso al modulo AMC, un generatore di segnale esterno (R&S SMF 100A) manda un singolo tono in CW in una banda tra 15.333-18.000GHz con una potenza tra +10dBm e +15dBm. Tale segnale RF viene moltiplicato in frequenza per 6, filtrato e amplificato dal modulo AMC per generare in uscita un segnale tra 92GHz e 108GHz, con una potenza tra 1 e 10 mW. Il modulo PA divide il segnale proveniente dal modulo AMC in due parti uguali, ciascuno dei quali è ulteriormente amplificato di una quantità controllabile dall'utente. La potenza del segnale in uscita varia tra 0.4mW e 1.6mW (-4 dBm to +2 dBm) e l'utente può regolarne la potenza con step di circa 0.2dB in tutta la banda di interesse 92-108 GHz. Tuttavia, solo una delle due uscite del modulo PA sarà utilizzata per le nostre applicazioni, mentre l'altra dovrà essere chiusa su carico adattato.

Nonostante la banda RF del nostro ricevitore 3 mm per SRT sia la stessa della Banda 3 di ALMA (84-116 GHz), la banda di sintonizzabilità dell'OL richiesta per utilizzo in SRT risulta essere molto più larga di quella necessaria ad ALMA Banda 3. Il motivo è che la banda IF del segnale del ricevitore ex-IRAM per SRT è di 500 MHz e situata fra 1.25-1.75 GHz, mentre per ALMA è di 4 GHz e situata fra 4.0-8.0 GHz.

Pertanto, la banda dell'oscillatore locale richiesta per coprire la stessa banda RF 84-116 GHz risulta pari a 85.75-114.25 GHz (28.5 % di banda relativa) ad SRT e pari a 92-108 GHz (16 % di banda relativa) ad ALMA Banda 3. Test preliminari in laboratorio ci hanno permesso di ricavare sperimentalmente la potenza dell'oscillatore locale ALMA al di fuori della sua banda nominale "ALMA Banda 3" e di determinare che questa ha dei valori accettabili per le nostre applicazioni ad eccezione nella parte a più alta frequenza, intorno ai 114 GHz.

Dal momento che l'oscillatore locale è uno strumento indispensabile per il ricevitore 3 mm è necessario che tale strumento sia completamente caratterizzato per le nostre applicazioni. Per questo motivo è di fondamentale importanza fare in modo che lo strumento sia controllato efficientemente da remoto anche in vista di una futura installazione del ricevitore in antenna. Dell'oscillatore locale ALMA ci è stato fornito solamente l'hardware poiché l'elettronica di controllo non era adatta alle nostre esigenze. Per questo motivo, al fine di controllare efficientemente lo strumento abbiamo sviluppato due schede di controllo: una per controllare l'AMC e una per controllare il PA. Entrambe le schede usano un bus di

comunicazione seriale per controllare i due moduli. Sono dotate di uno stadio di alimentazione accuratamente studiato per assicurare un'ottima stabilità delle correnti elevate che sono assorbite da AMC e PA (AMC/PA: +6V (1.5A), $\pm 15V$ (500mA)) e di un ulteriore livello di protezione contro sovratensioni dovuto a possibili malfunzionamenti di due moduli.

Nei due capitoli successivi descriveremo nel dettaglio le schede di controllo sia dal punto di vista hardware che dal punto di vista software. Concluderemo questo rapporto tecnico presentando i risultati relativi alla caratterizzazione in laboratorio dello strumento. Mostreremo i grafici relativi alla potenza che l'oscillatore locale ALMA dovrebbe fornire al mixer SIS del ricevitore. I risultati sono ottenuti tramite test in laboratorio delle due schede. Inoltre, mostreremo i risultati dei test di laboratorio del ricevitore ex-IRAM modificato e completamente assemblato con il nuovo oscillatore locale che dimostrano il corretto funzionamento dell'oscillatore, che riesce a pompare correttamente il mixer SIS.

4. PROGETTAZIONE E REALIZZAZIONE DELL'HARDWARE

4.1 Architettura e schema a blocchi

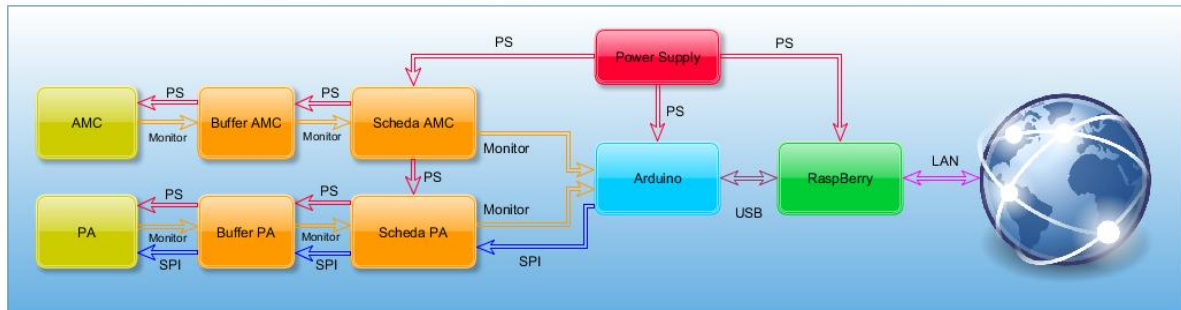


FIGURA 4-1: SCHEMA A BLOCCHI DELL'ARCHITETTURA

Per pilotare al meglio l'oscillatore locale è stata scelta un'architettura come quella mostrata in Figura 4-1 dove Arduino UNO Rev.3 è interfacciato tramite delle schede (schermate per mitigare gli effetti RFI) a un buffer che legge i parametri di funzionamento dei due moduli PA e AMC e invia i comandi necessari al controllo del modulo PA tramite bus SPI. Le schede AMC e PA devono gestire tutte le alimentazioni necessarie al funzionamento dei due moduli dell'oscillatore garantendo la corrente richiesta dall'utilizzatore. In aggiunta, la scheda PA deve fornire anche le connessioni a un bus di comunicazione di tipo SPI necessario per pilotare le tensioni delle giunzioni presenti nell'amplificatore. I due buffer, invece, garantiscono una lettura pulita dei segnali di ritorno (monitor) senza caricare, dal punto di vista elettrico, l'elettronica già presente dentro l'oscillatore locale. Inoltre, Arduino comunica tramite interfaccia USB ad una scheda Raspberry PI 2 Model B sulla quale viene eseguito il software di comunicazione che si interfaccia tramite LAN alla GUI di controllo. Tale architettura ci consente di avere un pieno controllo sia dei singoli moduli che dell'intero sistema.

Nei paragrafi successivi descriveremo nel dettaglio l'hardware che compone le due schede AMC e PA e i rispettivi buffer.

4.2 Scheda AMC

La scheda dedicata all'AMC ospita l'elettronica per la corretta polarizzazione del modulo in questione.

La scheda è stata progettata per ospitare 2 moduli ADC a 16 Bit (ADS1115 della Texas Instruments su modulo Adafruit [11]) che garantiscono una risoluzione di $187.5 \mu V$ e ci permettono la corretta lettura di tutti i monitor di tensione e corrente provenienti dal modulo AMC dell'oscillatore locale. Gli ADC dialogano con il microcontrollore Arduino tramite bus seriale I²C ed utilizzano 4 pin specifici per essere indirizzati univocamente.

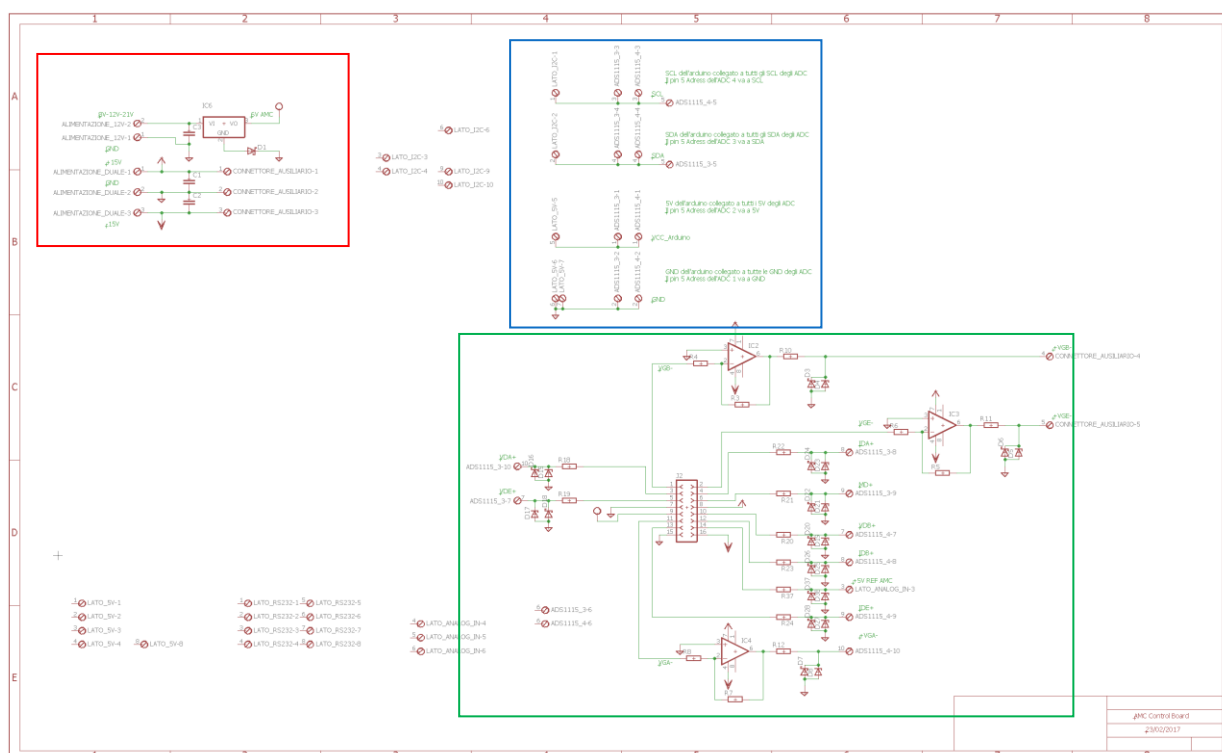


FIGURA 4-2: SCHEMATICO SCHEDA AMC

Nella Figura 4-2 è mostrato lo schematico che è stato progettato per pilotare l'AMC. I principali blocchi che lo compongono sono:

1. **Blocco di alimentazione (in rosso):** questo blocco, formato da uno stabilizzatore lineare e dai condensatori dimensionati a garantire il corretto filtraggio, fornisce tutte le tensioni di alimentazione di cui l'AMC necessita (6V, -15V, +15V). Particolare cura è stata posta per quanto riguarda la 6V in quanto i valori di corrente necessari risultano essere di 1.5A misurati contro i 900mA dati da specifica. Lo stabilizzatore, infatti, è stato dotato di opportuno dissipatore dimensionato per la potenza in gioco che nello specifico risulta essere poco superiore a 5W. La scelta di uno stabilizzatore lineare è vincolata al fatto di minimizzare il più possibile le eventuali RFI emesse dal circuito.
2. **Blocco di indirizzamento (in blu):** questo blocco, costituito da semplici collegamenti su PCB, fornisce gli indirizzi agli ADC utilizzati per misurare tutte le tensioni di ritorno provenienti

dall'utilizzatore (monitor). I 2 ADC utilizzati su ogni scheda sono stati interfacciati al microcontrollore tramite un bus I²C. Ogni singolo ADC è identificato in modo univoco da 4 indirizzamenti tramite i pin di configurazione utilizzati sia in fase di programmazione che di sviluppo del firmware.

3. **Blocco di condizionamento e protezione (in verde):** con questo blocco, vengono condizionati i segnali monitor provenienti dall'utilizzatore prima che questi vengano letti dagli ADC. Ogni ramo è stato customizzato in base ai parametri del segnale da campionare. È stato quindi rispettato il range del segnale da leggere amplificandolo o attenuandolo, in base alla necessità, per poter rispettare il range dettato dall'ADC. Sono stati inoltre aggiunti dei diodi *zener* e *schottky* in protezione a tutti gli ingressi ADC che interverranno al verificarsi di eventuali *overshoot* e *undershoot* di tensione.

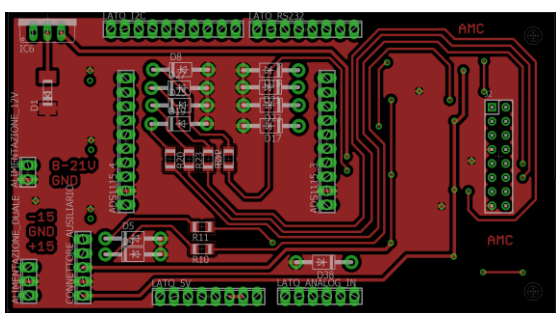


FIGURA 4-3: SCHEDA AMC TOP

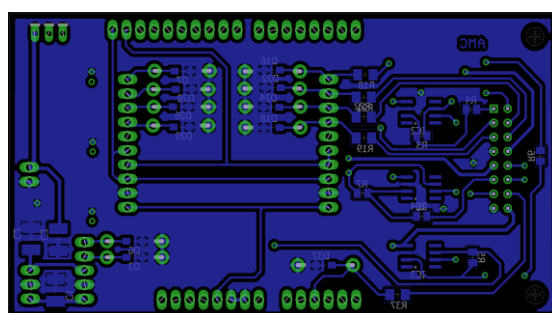


FIGURA 4-4: SCHEDA AMC BOTTOM

In Figura 4-3 e Figura 4-4 sono mostrati rispettivamente il lato TOP (in rosso) e BOTTOM (in blu) della scheda sbrogliata. Il layout è stato progettato in modo che la scheda possa connettersi come shield di Arduino. Questa soluzione, impiegata anche per altri prototipi [12-13], ottimizza gli spazi e gli ingombri garantendo un'ottima connessione tra tutti i pin della scheda.

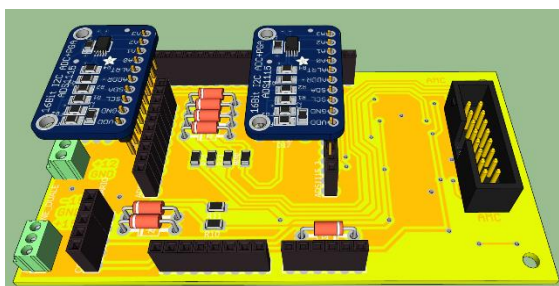


FIGURA 4-5: RENDERING 3D SCHEDA AMC

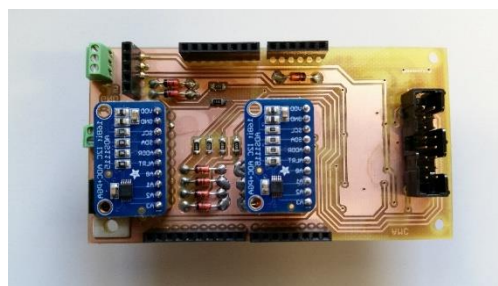


FIGURA 4-6: SCHEDA AMC REALIZZATA

Per valutare al meglio gli ingombri e quindi prevenire eventuali interferenze meccaniche legate al posizionamento dei componenti è stato realizzato un rendering 3D della scheda mostrato in Figura 4-5. Per realizzare il rendering 3D è stato utilizzato il software Google Sketchup[14].

In Figura 4-6, invece, è mostrato il prodotto finale realizzato homemade nei laboratori dell'Osservatorio Astronomico di Cagliari.

4.3 Scheda PA.

La scheda dedicata al PA ospita l'elettronica per la corretta polarizzazione del modulo in questione.

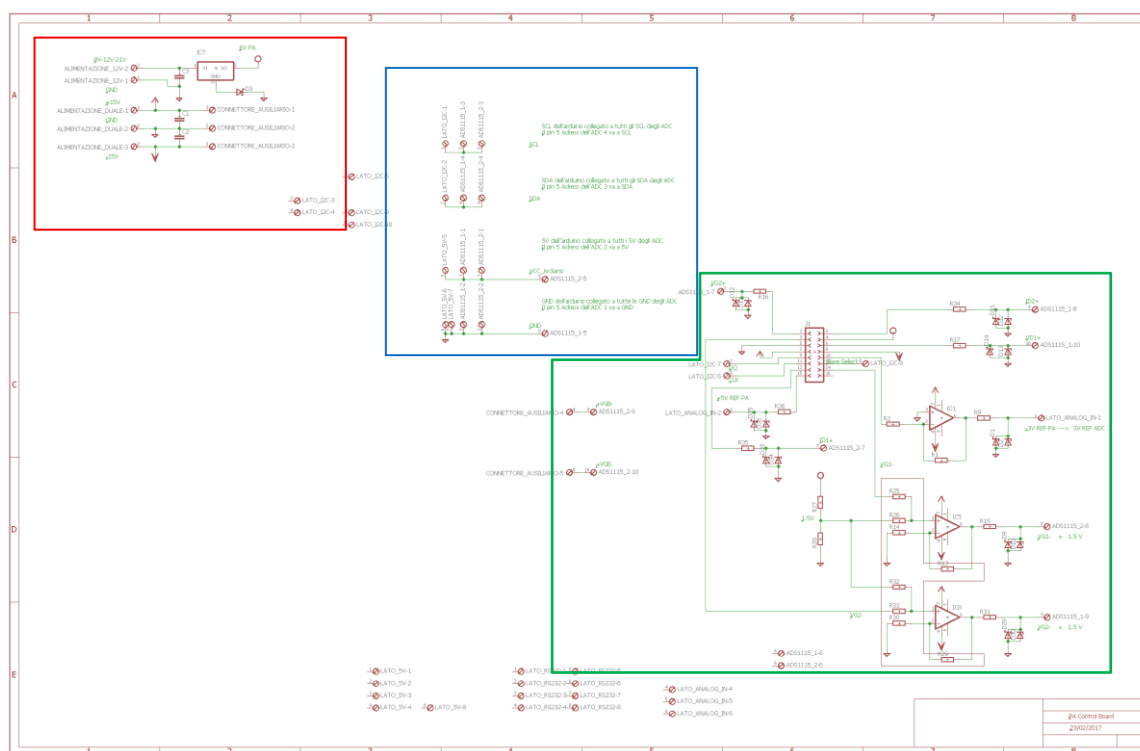


FIGURA 4-7: SCHEMATICO SCHEDA PA

Nella Figura 4-7 è mostrato lo schematico che è stato progettato per pilotare il PA. I principali blocchi che lo compongono sono molto simili a quelli che compongono la scheda per il controllo AMC. Le principali differenze sono essenzialmente presenti nel blocco di condizionamento e monitoraggio, anch'esso totalmente calibrato e customizzato per tutte le tensioni presenti sui monitor del PA, dove è presente un bus di comunicazione SPI che si interfaccia e trasferisce i comandi al PA tramite gli appositi pin connessi al bus di Arduino. Pertanto abbiamo:

1. **Blocco di alimentazione (in rosso):** questo blocco formato da uno stabilizzatore lineare e dai condensatori dimensionati per garantire il corretto filtraggio, fornisce tutte le tensioni di alimentazione di cui il PA necessita (6V, -15V, +15V). Anche in questo caso, particolare cura è stata posta per quanto riguarda la 6V in quanto i valori di corrente reali risultano essere di poco superiori a quelli nominali (600mA) e pari a circa 1A. Anche in questo caso, il dissipatore necessario è stato collegato allo stabilizzatore lineare per dissipare una potenza stimata per eccesso nell'ordine di 5W.
2. **Blocco di indirizzamento (in blu):** questo blocco, costituito da semplici collegamenti su pc, fornisce gli indirizzi agli ADC utilizzati per misurare tutte le tensioni di ritorno provenienti dall'utilizzatore. Dal momento che le schede PA e AMC risiedono sul medesimo Arduino, gli indirizzi sono stati settati tramite appositi pin stando attenti a non sovrascrivere gli indirizzi già utilizzati per gli ADC della scheda AMC.

3. Blocco di condizionamento e protezione (in verde): con questo blocco, sono condizionati i segnali monitor provenienti dall'utilizzatore prima che questi vengano letti dagli ADC. Anche in questo caso sono stati predisposti dei diodi con giunzione tipo *zener* e *schottky* per prevenire eventuali *overshoot* e *undershoot* in fase di misurazione.

Il modulo PA, a differenza del modulo AMC, contiene al suo interno due potenziometri **DS1867** collegati in *daisy chain* tramite bus SPI. Pertanto, come accennato in precedenza, è stato predisposto un collegamento tra il microcontrollore e il PA per poter settare le giunzioni di *drain* e di *gate* degli stadi di amplificazione interni all'amplificatore. Questi due potenziometri servono per regolare il punto di riposo (*bias point*) nei fet dello stadio di amplificazione interno al PA. Nella scheda del PA, quindi, è stato necessario predisporre le seguenti connessioni:

- **DQ:** dove transitano i bit di setup;
- **CLK:** che fornisce il clock per la sincronizzazione e il timing del bus;
- **GND:** potenziale di riferimento del bus di comunicazione.

Per verificare l'effettivo numero di bit inviati e testare il bus di comunicazione prima di collegare tutto il sistema, è stato effettuato un test dove la stringa di bit veniva inviata direttamente ad un oscilloscopio (Rohde&Schwarz RTO 1044) che, avendo a disposizione l'analisi dei bus digitali ci ha consentito di verificare tutti i bit inviati e i fronti di salita corrispondenti. Inoltre, grazie a questo test è stato possibile verificare anche le temporizzazioni del segnale e dimostrare che queste fossero le medesime dettate dal datasheet del componente (DS1867).

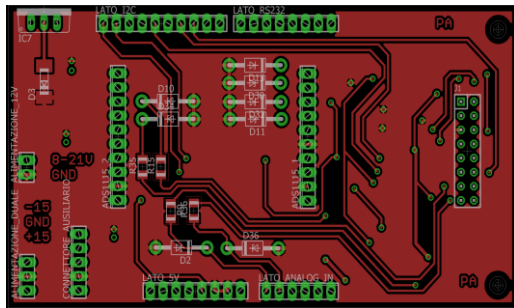


FIGURA 4-8: SCHEDA PA TOP

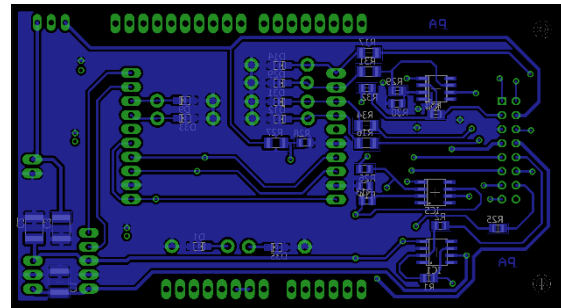


FIGURA 4-9: SCHEDA PA BOTTOM

In Figura 4-8 e Figura 4-9 sono mostrati rispettivamente il lato TOP (in rosso) e BOTTOM (in blu) della scheda sbrogliata. Anche in questo caso il layout è stato progettato in modo che la scheda possa connettersi come shield di Arduino.

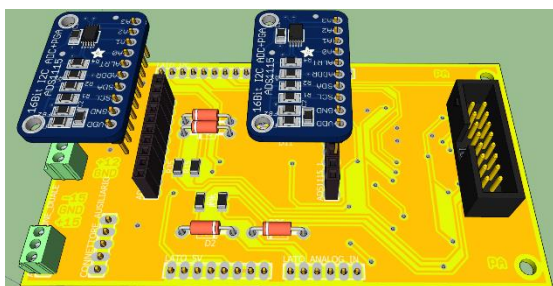


FIGURA 4-10: RENDERING 3D SCHEDA PA

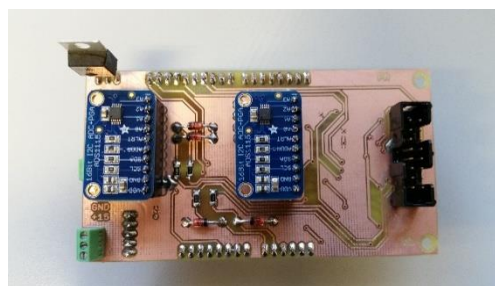


FIGURA 4-11: SCHEDA PA REALIZZATA

Per eventuali ingombri meccanici inoltre, è stato realizzato un rendering 3D della scheda mostrato Figura 4-10. In questo caso il 3D ci ha permesso di individuare alcuni diodi che potevano interferire meccanicamente in fase di montaggio. Tramite il 3D, infine è stato possibile verificare che i connettori che portano l'alimentazione duale fossero perfettamente in asse.

In Figura 4-11, invece, è mostrato il prodotto finale realizzato homemade nei laboratori dell'Osservatorio Astronomico di Cagliari.

4.4 Modulo buffer.

Per garantire una lettura corretta dei valori senza caricare dal punto di vista elettrico l'elettronica già presente dentro l'oscillatore locale è stato realizzato un modulo buffer. Questo è caratterizzato da un'elevata impedenza di ingresso (tipicamente nell'ordine del megaohm) e da una bassa impedenza di uscita (tipicamente di 70Ω). È stato realizzato con degli operazionali della Texas Instruments (opa37) in configurazione a guadagno unitario. Il modulo buffer si compone di 2 schede sovrapposte e connesse tramite 4 torrette di tipo m3 con connettori a 16 poli che garantiscono un rapido collegamento in caso di manutenzione.

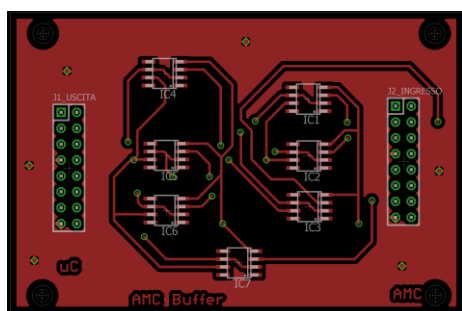


FIGURA4-12: SCHEDA BUFFER AMC TOP

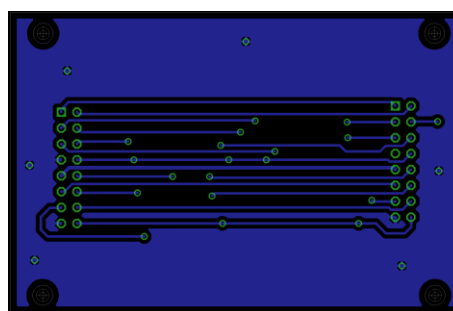


FIGURA4-13: SCHEDA BUFFER AMC BOTTOM

In Figura4-12 e Figura4-13 sono mostrati il lato TOP (in rosso) e il lato BOTTOM (in blu) del buffer dedicato per il modulo AMC.

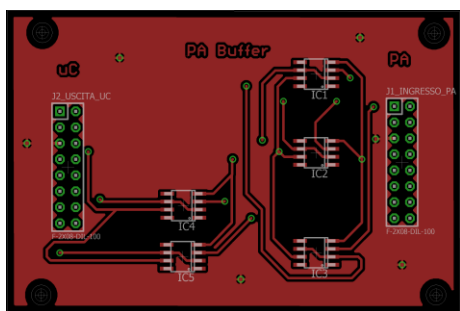


FIGURA 4-14: SCHEDA BUFFER PA TOM

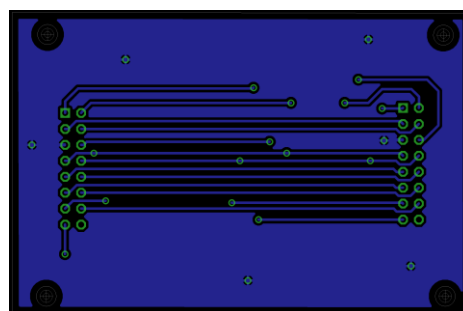


FIGURA 4-15: SCHEDA BUFFER PA BOTTOM

In Figura 4-14 e Figura 4-15 sono mostrati il lato TOP (in rosso) e il lato BOTTOM (in blu) del buffer dedicato per il modulo PA.

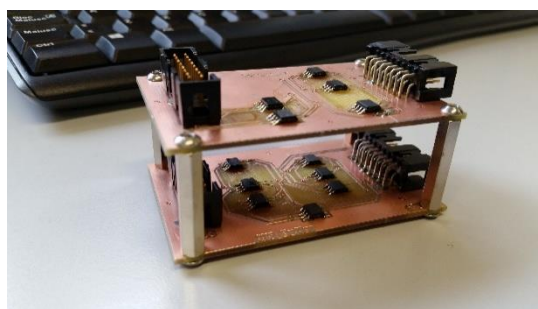


FIGURA 4-16: SCHEDE BUFFER AMC E BUFFER PA

Poiché gli ingombri meccanici risultano palesemente irrilevanti in questo modulo, non è stato realizzato alcun rendering 3D. In Figura 4-16 è mostrato il prodotto finale realizzato nel laboratorio del Osservatorio Astronomico di Cagliari.

5. SOFTWARE E FIRMWARE DELLE SCHEDE DI CONTROLLO

5.1 Introduzione

Come già descritto nel capitolo 4.1, Arduino comunica tramite interfaccia USB con una scheda Raspberry PI 2 Model B, con sistema operativo Linux, nel quale viene eseguito il software di comunicazione che si interfaccia tramite LAN alla GUI di controllo.

La Raspberry svolge sostanzialmente due compiti fondamentali:

- ricevere i comandi provenienti dall'interfaccia ethernet per inviarli ad un Arduino o ad un altro a seconda del comando ricevuto;
- ricevere i valori e gli stati da Arduino per inviarli ai client che li hanno richiesti tramite interfaccia ethernet.

Per questo motivo, sia il firmware di Arduino che il software di controllo della Raspberry constano di un loop principale che rimane costantemente in attesa di ricevere dei dati sulla porta seriale il primo e delle stringhe di comando sulla porta ethernet il secondo.

Nei paragrafi successivi descriveremo sia il firmware scritto per le schede Arduino che il software di controllo della Raspberry.

5.2 Input attesi dalle schede Arduino

Il formato dei pacchetti attesi sulla porta seriale ammessi da Arduino devono avere la lunghezza di 7 byte. Il primo byte è sempre il codice del comando da eseguire ricevuto, i restanti byte assumono un significato diverso a seconda del comando. I comandi possono avere il seguente formato:

BYTE 0	BYTE 1	BYTE 2	BYTE 3	BYTE 4	BYTE 5	BYTE 6
COMMAND_01	0	0	0	0	0	0
COMMAND_02	0	H_BYTE	L_BYTE	0	0	0
COMMAND_03	POT_01_STATE	POT_01_H	POT_01_L	POT_02_STATE	POT_02_H	POT_02_L

dove 0 indica che viene inviato un byte con valore 0 e il suo valore viene ignorato.

Il formato di comando COMMAND_01 è utilizzato per i comandi di lettura per i quali non è necessario inviare ulteriori dati. Questo tipo di comando è utilizzato nei casi in cui non sia necessario inviare parametri, per esempio per tutti i comandi di lettura di valori degli ADC.

Il comando di tipo COMMAND_02 è utilizzato per inviare un valore a 16 bit sulla seriale. Il valore è diviso nel byte MSB (H_BYTE) e in quello LSB (L_BYTE). Questo tipo di comando è quello utilizzato quando si deve inviare un comando di impostazione con un solo valore che può essere interpretato sia come booleano che come valore intero a 16 bit. Per esempio questo formato di comando è utilizzato nel

caso di posizionamento di un motore passo passo, attivazione/disattivazione di una modalità o di invio di un valore a un DAC.

Infine, il comando di tipo `COMMAND_03` è utilizzato esclusivamente per impostare i valori dei due potenziometri del PA. Per fare questo, a ciascuno dei due potenziometri (`POT_01`, `POT_02`) deve essere inviata una stringa da 17 bits suddivisa in un valore a 16 bit, rispettivamente chiamato `POT_01_H` e `POT_01_L` come byte alto e basso al potenziometro 1 e `POT_02_H` e `POT_02_L` come byte alto e basso al potenziometro 2, più un byte `POT_01_STATE` al potenziometro 1 e `POT_02_STATE` al potenziometro 2 utilizzato per inviare il 17esimo bit.

5.3 Output delle schede Arduino

Una volta che Arduino riceve dei comandi che richiedono una risposta o un feedback, questo deve restituire uno stream di 3 byte nella seguente forma:

BYTE 0	BYTE 1	BYTE 2
OUTPUT	H_BYTE	L_BYTE

dove il primo byte identifica il codice della risposta, e gli altri due l'eventuale valore a 16 bit suddiviso, come visto in precedenza, nel byte MSB e nel byte LSB.

5.4 Firmware delle schede AMC-PA

Il firmware di queste schede si occupa di monitorare tutti i valori di tensione provenienti dai due moduli AMC-PA dell'oscillatore locale e di impostare il valore delle tensioni di *gate* e *drain* (`VD1`, `VD2`, `VG1`, `VG2`) dei due potenziometri presenti nel modulo PA.

I diversi valori di tensione vengono letti da dei convertitori analogico digitali con un comando specifico per ogni valore richiesto, mentre l'impostazione del potenziometro prevede l'invio al PA di 34 bit (17 bit per ogni potenziometro). Il comando per eseguire questa impostazione è così composto:

- 1 bit per indicare come deve impostato il potenziometro. Questo bit nella configurazione utilizzata all'interno del nostro PA, è ignorato;
- 1 byte per impostare il "valore" da associare al *drain* (`VD`) del potenziometro;
- 1 byte per impostare il "valore" da associare al *gate* (`VG`) del potenziometro.

Devono essere inviati nell'ordine prima i valori per il potenziometro 1 e poi quelli per il potenziometro 2 e, inoltre, è necessario inviare sempre tutti e 34 i bit.

Non essendo i bit da inviare separabili in byte è stato necessario inviarli singolarmente utilizzando a basso livello il protocollo SPI.

5.5 Software di test delle schede Arduino

Per testare il funzionamento delle schede Arduino ed effettuare le prime prove di caratterizzazione è stato scritto un software in Microsoft Visual Basic .NET che permette di inviare tutti i possibili comandi accettati dalle schede e leggere gli output delle stesse. È stato scelto questo linguaggio per la velocità di sviluppo e la possibilità di avere immediatamente un'interfaccia grafica semplice da utilizzare.

Il software è stato progettato in schede come si può vedere in figura 5.1:

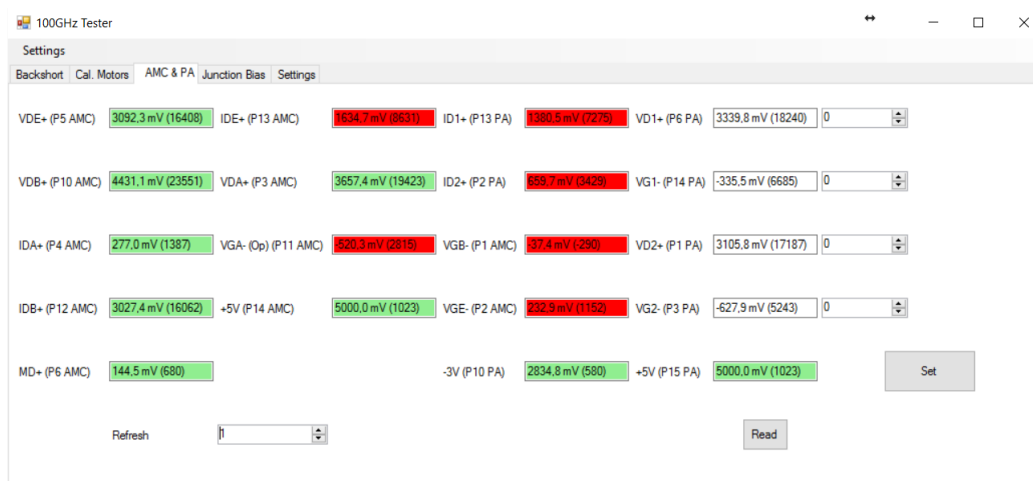


FIGURA 5.1: GUI DEL SOFTWARE DI TEST.

La scheda AMC & PA permette di visualizzare tutti i valori monitorabili dell'AMP e del PA. Impostando un valore di refresh è possibile effettuare un monitoraggio continuo. I singoli valori vengono evidenziati in verde quando sono all'interno del normale range di funzionamento, in rosso in caso contrario. Sulla destra sono presenti le quattro caselle in cui è possibile inserire i valori dei potenziometri del PA (nel range [0-255]) che vengono inviati alla scheda tramite il pulsante "Set". Questa scheda del programma è stata utilizzata per caratterizzare con precisione gli ADC che restituiscono i valori visualizzati, rendendo possibile la definizione della retta di corrispondenza tra i valori digitali dell'ADC e quelli di tensione dell'oscillatore locale. Inoltre è stata utilizzata per valutare inizialmente il comportamento della giunzione a temperatura ambiente.

La scheda SETTING del programma imposta invece i parametri di comunicazione con le schede Arduino permettendo di impostare le porte seriali utilizzate da ogni scheda. Un pulsante "SearchBoards" abilita la ricerca automatica delle schede. Quest'ultima funzionalità viene avviata ogni volta che è mandato in esecuzione il programma e restituisce un errore nel caso in cui una delle schede non dia risposta.

5.6 Software di controllo della Raspberry

Il software di controllo installato nella Raspberry scritto in Python 2.7 effettua le seguenti operazioni:

- All'avvio ricerca sulle sue porte USB le schede Arduino ad essa collegate;
- Attiva una porta per la comunicazione tramite SOCKET alla Raspberry;
- È avviato un loop infinito di ascolto sulla porta per rispondere ai comandi ricevuti e restituire i valori richiesti.

La procedura di avvio della ricerca delle schede è identica a quella descritta nella scheda SETTING del software di test scritto in Visual Basic .NET. Dal momento che sulla scheda Raspberry è installato un sistema operativo Linux Ubuntu 15.4, tale sistema operativo associa ad ogni periferica connessa una porta seriale virtuale. Per individuare quale porta seriale sia stata associata ad una specifica scheda è stato predisposto il comando speciale IDENTIFY al quale risponde ciascuna delle schede collegate. Il comando IDENTIFY è identificato dal valore 100. Ogni scheda risponde diversamente a questo comando restituendo lo stesso codice di output (100) seguito da un valore identificativo della scheda stessa secondo la seguente tabella:

Scheda	Valore decimale	Valore Esadecimale
AMC-PA	200	0XC8
Calibration	201	0XC9
JB	202	0XCA

All'avvio del programma di controllo delle schede Arduino, sulla Raspberry è eseguito un controllo di tutte le porte seriali utilizzate. Ad ognuna di queste è inviato lo stesso comando IDENTIFY (0X100 0X00 0X000X000X000X00 0X00); se la Raspberry riceve una risposta con il codice di output 100 e uno dei valori della tabella precedente, a quella porta seriale associa la scheda corrispondente. Questo permette di comunicare con quella scheda utilizzando i suoi comandi specifici; nel caso in cui non sia stata ricevuta alcuna risposta, la scheda ricercata sarà considerata non presente e quindi verrà sollevata un'eccezione.

Il socket per la ricezione dei comandi viene aperto sulla porta 5025 e non presenta filtri o limitazioni sulla comunicazione.

La figura 5.2 mostra la GUI del software di controllo del ricevitore 100 GHz dove in azzurro è evidenziata la parte dei comandi e monitor relativa al controllo dell'oscillatore locale ALMA.

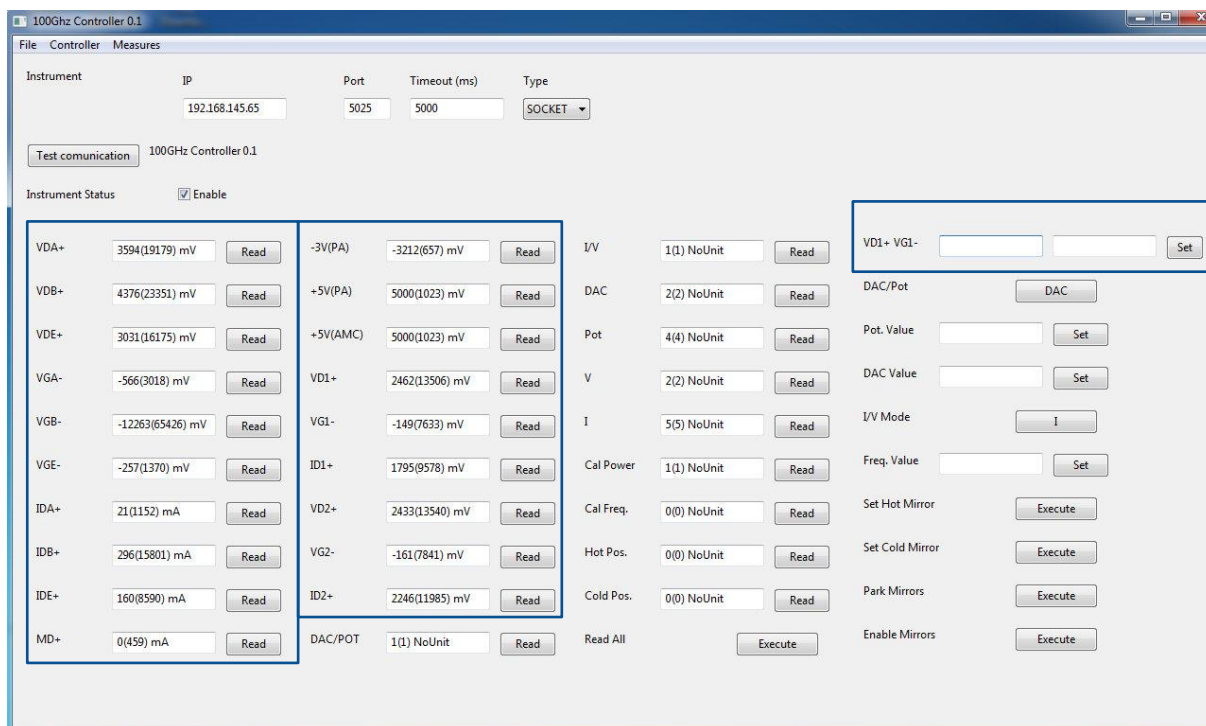


FIGURA 5.2: GUI DEL SOFTWARE DI CONTROLLO.

5.7 Linguaggio dei comandi

I comandi che vengono ricevuti tramite la rete ethernet sono delle stringhe di testo nella forma:

COMMAND_TYPE PARAMETER [VALUE VALUE VALUE]

dove COMMAND_TYPE può assumere i valori “GET” o “SET” e indica se il comando è un comando di lettura di un valore e quello di impostazione di un parametro. VALUE sono i byte che vengono inviati come valori per il parametro nel solo comando “SET”.

La notazione BNF dei comandi è la seguente:

<COMMAND> ::= <COMAND_TYPE><PARAMETER>[<VALUE>]

<COMMAND_TYPE> ::= SET | GET

<PARAMETER> ::=

VDE_PLUS | VDB_PLUS | IDA_PLUS | IDB_PLUS | MD_PLUS | IDE_PLUS | VDA_PLUS | VGA_MINUS | ID1_PLUS | VG1_MINUS | VGB_MINUS | VGE_MINUS | VD2_PLUS | ID2_PLUS | VG2_MINUS | VD1_PLUS | ARD_A0 | ARD_A1 | FIVE_V_PLUS_REF | GET_DAC_STATUS | GET_I_V_MODE | GET_DAC_VALUE | GET_POT_VALUE | GET_V_VALUE | GET_I_VALUE | SET_DAC_STATUS | SET_POT_VALUE | SET_DAC_VALUE | SET_I_V_MODE

VALUE ::= Byte

6. TEST FINALI

Per validare il sistema di controllo dell'oscillatore locale, in laboratorio sono state fatte una serie di misure al fine di valutare l'effettiva potenza in uscita dallo strumento e verificare che la potenza di tale segnale sia sufficiente a pompare il mixer SIS del ricevitore.

Per fare questo abbiamo usato i seguenti strumenti di misura presenti nel laboratorio microonde dell'Osservatorio Astronomico di Cagliari:

- Generatore di segnale Rohde&Schwarz SMF100A (100kHz – 22GHz);
- Oscilloscopio Rohde&Schwarz RTO 1044;
- Power meter millimetrico Erickson PM5.

Un software di monitoraggio remoto "RFLab" [15] permette di poterli controllare remotamente. Sfruttando il principio di funzionamento di "RFLab", al software di controllo della Raspberry è stata aggiunta una routine che permetta il controllo del generatore di segnale e del power meter millimetrico. Tale routine fa variare a steps il valore della frequenza e della potenza del segnale inviati dal generatore di segnale all'ingresso dell'oscillatore locale e, per ogni impostazione, effettua una o più letture del powermeter per poi salvare i risultati in un file csv. I valori di potenza inviati all'ingresso dell'oscillatore sono stati ulteriormente corretti inserendo un file di calibrazione del cavo utilizzato per l'invio del segnale, creato separatamente dal programma di monitoraggio degli strumenti.

Dopo aver ricavato i range dei valori delle tensioni di *drain* e *gate* da impostare nel PA al di fuori dei quali si rischia di danneggiare lo strumento e impostati come limiti nel software di controllo, abbiamo verificato che la potenza in uscita dai canali dell'oscillatore fosse nominalmente sufficiente a pompare il mixer SIS. Si stima che il valore minimo della potenza di oscillatore locale che è necessario fornire sul piano della flangia in guida d'onda WR10 ($2.54 \times 1.27 \text{ mm}^2$) dell'ingresso OL del ricevitore SIS 3 mm ex-IRAM sia dell'ordine di 100 μW . In effetti, la potenza di oscillatore locale P_{OL} richiesta per pompare un dato ricevitore SIS è proporzionale al quadrato della frequenza ν_{OL} , al quadrato del numero N di giunzioni SIS (se poste in serie), e dipende dalla resistenza normale R_N della caratteristica IV della giunzione secondo la formula:

$$P_{OL} = \frac{(N h \nu_{OL} \alpha_n)^2}{2 R_N e^2}$$

dove $h = 6.626 \times 10^{-34} \text{ J s}$ è la costante di Planck, $e = 1.602 \times 10^{-19} \text{ coulomb}$ è la carica dell'elettrone ed α_n è una costante. Si stima pertanto che la potenza necessaria per pompare il mixer, valutata sulla giunzione stessa, sia dell'ordine di 10 nW a 100 GHz per un mixer a singola giunzione, $N=1$. Considerando le perdite di circa -6 dB (≈ 4 in lineare) dovute al disadattamento dell'OL nel mixer SSB, le perdite di accoppiamento di circa -28 dB (≈ 630 in lineare) dovute all'accoppiatore direzionale criogenico posto di fronte al mixer SIS, e le perdite di circa 6 dB (≈ 4 in lineare) dovute alla guida d'onda WR10 posta dentro al criostato che collega l'accoppiatore direzionale alla porta WR10 dell'ingresso OL al ricevitore, si ottiene un valore di circa 100 μW come valore di potenza OL che è necessario fornire alla flangia di ingresso del dewar. La potenza in uscita dal PA dovrà essere almeno di un ordine di grandezza superiore a tale valore, in modo da compensare le perdite della guida d'onda WR10 posta esternamente al dewar che collega l'OL al ricevitore. Ad esempio la specifica per la minima potenza richiesta dal sistema di oscillatore locale di ALMA Banda 3 è di 2 mW nella banda 92-108 GHz. La figura 6.1 mostra l'andamento della potenza in

uscita misurata dal canale 1 al variare della frequenza per un dato valore di tensione di *drain* e *gate* ($V_{D1}=1.67V$ and $V_{G1}=-0.1V$). In particolare, la curva mostra un picco di potenza intorno ai 7 mW alla frequenza di 98 GHz. Si nota inoltre che la potenza è superiore a 1 mW su tutta la banda di frequenza OL di nostro interesse, 85.75-114.25 GHz. Ottimizzando i valori delle tensioni di *drain* e di *gate* per ogni specifica frequenza OL in banda 85.75-114.25 GHz si riesce a raggiungere la specifica di $P_{OL}>2$ mW su tutta la banda di interesse.

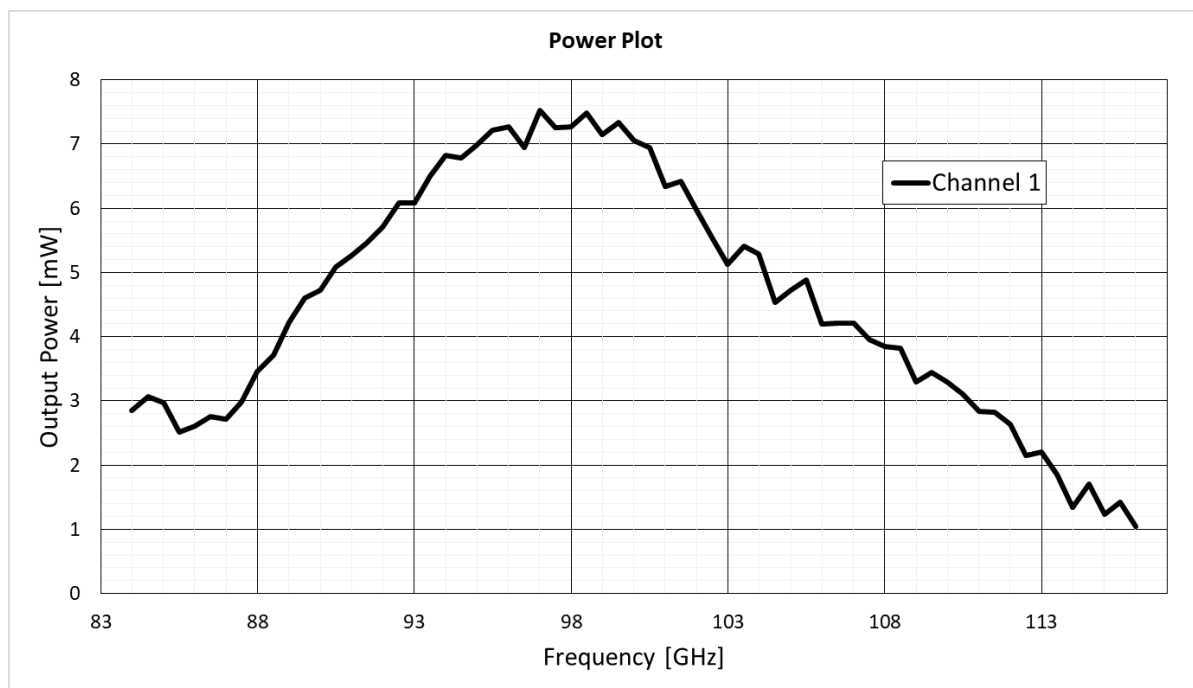


FIGURA 6.1: POTENZA MISURATA ALL'USCITA DEL CANALE 1 DEL LO ALMA.

La figura 6.2 mostra, invece, la caratteristica I-V della giunzione del mixer SIS pompata e non pompata dall'oscillatore locale e misurata con l'oscilloscopio. In questo caso, la frequenza dell'oscillatore è stata impostata a 87.5GHz con un tuning del mixer LSB a 86 GHz.

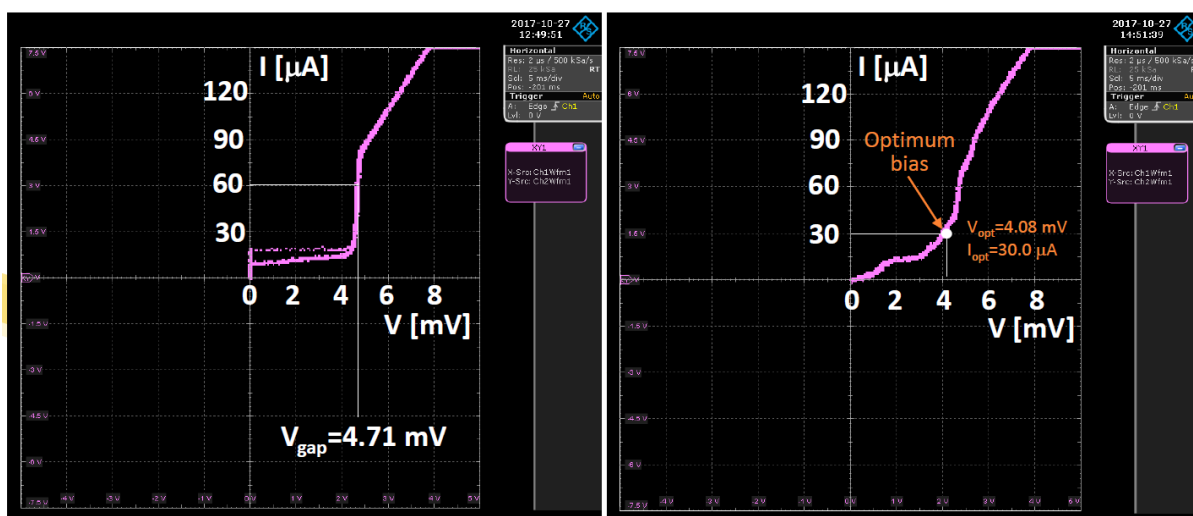


FIGURA 6.2: MISURA DELLA CARATTERISTICA I-V DELLA GIUNZIONE SIS: (SINISTRA) I-V NON POMPATA, (DESTRA) I-V POMPATA.

Infine, mostriamo i risultati dei test preliminari in laboratorio del ricevitore 3mm. La tabella 6.1 mostra le temperature di rumore del ricevitore misurate nella banda 84-115 GHz ottenute in condizioni non ottimali di raffreddamento del mixer SIS che ha operato alla temperatura fisica di circa 6.6 K, molto più alta dell'ottimale, tramite raffreddamento con criogeneratore ARS. In particolare, il valore ottimale per la temperatura fisica di tale mixer SIS è inferiore o uguale a circa 4 K (deve essere inferiore alla metà della temperatura critica T_c del superconduttore, dove per il Niobio $T_c=9.2$ K)

Banda RF osservata [GHz]	Tuning mixer SIS	Frequenza OL [GHz]	Valori di bias	Temperatura rumore [K]
83.7-84.3	LSB	85.5	$V_b=4.07$ mV; $I_b=28.0$ μ A	73
85.7-86.3	LSB	87.5	$V_b=4.08$ mV; $I_b=30.0$ μ A	65.8
89.7-90.3	LSB	91.5	$V_b=4.06$ mV; $I_b=34.5$ μ A	78
94.7-95.3	LSB	96.5	$V_b=4.05$ mV; $I_b=31.7$ μ A	143
98.2-98.8	LSB	100	$V_b=4.04$ mV; $I_b=28.7$ μ A	158
104.7-105.3	LSB	106.5	$V_b=3.93$ mV; $I_b=28.5$ μ A	287

TABELLA 6.1: RISULTATI TEST PRELIMINARI OTTENUTI CON RAFFREDDAMENTO NON OTTIMALE DEL MIXER SIS (OPERANTE ALLA TEMPERATURA DI 6.6 K ANZICHÉ 4 K).

Nonostante l'alto valore della temperatura fisica del mixer i test mostrano risultati promettenti nella parte a bassa frequenza 84-95 GHz del ricevitore. In particolare, abbiamo misurato un rumore del ricevitore inferiore a 70 K a 86 GHz, sufficientemente basso per consentire di testare SRT ed effettuare osservazioni preliminari nell'ambito del VLBI millimetrico [16].

7. CONCLUSIONI

Con il presente internal report abbiamo mostrato la caratterizzazione dell'oscillatore locale ALMA per il ricevitore 100 GHz del Sardinia Radio Telescope. Lo strumento è stato testato dal punto di vista delle prestazioni sia come dispositivo singolo sia integrato nel sistema ricevitore confermando il comportamento e i risultati attesi. I risultati dei test, inoltre, confermano che il sistema di controllo da noi progettato per controllare remotamente il dispositivo funziona correttamente anche in vista della futura installazione in antenna del ricevitore.

8. BIBLIOGRAFIA

- [1] The Sardinia Radio Telescope. Web site: <https://www.srt.inaf.it/>
- [2] J. Blondel, M. Carter, A. Karpov, B. Lazareff, F. Mattiocco and J. Lamb, “Dual-Channel SIS Receivers for the IRAM Plateau de Bure Interferometer”, International Journal of Infrared and Millimeter Waves, Vol. 17(12), 2133-2144 (1996).
- [3] M. Carter, “Optics for the HDV10 Dual Frequency Band Receiver for the Plateau de Bure Interferometer”, IRAM Receiver Technical Memo, n°3, (1994).
- [4] B. Lazareff, “IRAM Newsletter n°38” (January 1999).
- [5] A. Ladu, T. Pisanu, A. Navarrini, P. Marongiu, G. Valente, “A 3 mm band SIS Receiver for the Sardinia Radio Telescope”, Proc. SPIE 9153 Millimeter, Submillimeter and Far-Infrared Detectors and Instrumentation for Astronomy VII, (July 2014).
- [6] <https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>
- [7] <https://www.arduino.cc/en/Main/ArduinoBoardUnoSMD>
- [8] A. Ladu, P. Ortu, A. Saba, M. Pili, F. Gaudiomonte, A. Navarrini, E. Urru, T. Pisanu, G. Valente, P. Marongiu, G. Mazzarella, “The control system of the 3 mm band SIS receiver for the Sardinia Radio Telescope”, Proc. SPIE 9914 Millimeter, Submillimeter and Far-Infrared Detectors and Instrumentation for Astronomy VIII, (July 2016).
- [9] E. Bryerton, K. Saini, M. Morgan, M. Stogoski, T. Boyd and D. Thacker, “Development of Electronically Tuned Local Oscillators for ALMA”, 30th Conf. on Infrared and Millimeter Waves & 13th Intl. Conf. on Terahertz Electron, (2005).
- [10] K. Saini and J. Muehlberg, “Local Oscillator Driver Users Guide for Bands 3 through 10”, ALMA Internal Report, (August 2013).
- [11] <https://www.adafruit.com>
- [12] A. Cabras, P. Ortu, A. Saba, E. Urru, T. Pisanu, “Sistema di monitoraggio in RT della frequenza di rotazione ” OAC Internal Report n°63, (March 09, 2017).
- [13] P. Ortu, M. Pili, A. Saba, E. Urru, T. Pisanu, “Progettazione e realizzazione del Position Sensing Device” OAC Internal Report n°53, (January 12, 2016).
- [14] <https://www.sketchup.com/it>
- [15] A. Saba, G. Valente, “RFLab” OAC Internal Report n°60, (December 07, 2016).
- [16] A. Ladu, P. Ortu, A. Saba, F. Gaudiomonte, E. Urru, T. Pisanu, G. Valente, P. Marongiu, A. Navarrini, “Sviluppi di ricevitori e componentistica in banda 3 mm ad INAF-OA Cagliari,” IV workshop sull’Astronomia Millimetrica in Italia, Bologna, 9 Nov. 2017, <https://zenodo.org/record/1116242#.W3wQxrh9iUk>

ERRATA CORRIGE

La copertina dell'internal report è la seguente:

INTERNAL REPORT

Sistema di controllo per la caratterizzazione dell'oscillatore locale ALMA impiegato nel ricevitore 100GHz del Sardinia Radio Telescope

Adelaide Ladu, Pierluigi Ortu, Andrea Saba, Mauro Pili, Alessandro Navarrini, Francesco Gaudiomonte, Pasqualino Marongiu, Enrico Urru, Tonino Pisanu, Giampaolo Serra

REPORT N. 73

RELEASED: 28/08/2018

Revisore: Raimondo Concu



Osservatorio
Astronomico
di Cagliari