



Celebrazione del centenario del conferimento del Premio Nobel a Guglielmo Marconi



**AMSAT Italia sarà
presente a
SATEXPO 2010**

In questo numero:

Informazioni & novità	p2
Front-end e USRP	p4
DCRTX "verso" la radio digitale #2	p11
Notizie Associative	p17
Notizie aerospaziali	p18

Informazioni & novità

Innanzitutto un augurio di cuore per un

Buon Anno 2010

a voi ed alle vostre famiglie !

L'augurio arriva con l'ultimo numero del 2009 spedito però nel 2010, come ormai era una consuetudine (ma lo fa anche AMSAT NA ...), ed è l'occasione per darvi alcune informazioni e qualche novità che per motivi di spazio e soprattutto di tempo (troppo stretto) non avevamo potuto dare nel precedente numero del Bollettino.

Accordo con ASI

Il 5 Novembre 2009 l'Ingegnere Enrico Saggese Presidente dell'ASI (Agenzia Spaziale Italiana) e il nostro Presidente, Marco Lisi, hanno siglato un accordo di collaborazione tra AMSAT Italia e l'Agenzia suddetta dove entrambe si impegnano a lavorare in modo sinergico per realizzare iniziative congiunte in campo divulgativo.

L'accordo è finalizzato alla valorizzazione ed alla diffusione della cultura scientifica in generale ed in particolare delle Scienze Aerospaziali e anche per stimolare le nuove generazioni verso gli studi scientifici.

L'accordo è fortemente basato sulla promozione e la realizzazione dei collegamenti con gli astronauti a bordo della "International Space Station".

In sintesi, AMSAT Italia sarà responsabile dell'organizzazione operativa dei contatti radioamatoriali, gestione della strumentazione e del coinvolgimento delle scuole; ASI sarà responsabile del coordinamento delle iniziative congiunte, ovvero di quelle, segnalate da AMSAT Italia.

Con questo accordo AMSAT Italia si impegna a partecipare ai meeting di coordinamento del progetto ARISS, proporre ad ASI progetti di particolare contenuto tecnico e valenza divulgativa, segnalare all'ASI realtà accademiche/associative/culturali *no profit*, che operano nel campo della diffusione della cultura scientifica.

Inoltre mantenere ASI costantemente informata circa le proprie iniziative e partecipare alle attività concordate con l'Agenzia con lo scopo comune di promuovere l'immagine dell'ASI, di AMSAT Italia e dell'Italia spaziale nel mondo.

L'ASI si impegna a promuovere iniziative congiunte, fornire materiale divulgativo, sponsorizzare alcune delle iniziative segnalate da AMSAT Italia, favorire lo scambio di informazioni anche con realtà diverse da quelle delle scuole quali imprese, istituzioni e centri di eccellenza che operano per le scienze e lo sviluppo di tecnologie aerospaziali, nazionali ed internazionale.

L'accordo siglato con l'ASI è un risultato eccezionale sotto ogni profilo, soprattutto per il fatto che un'agenzia governativa come l'ASI ha deciso godere della collaborazione di un gruppo di appassionati come AMSAT Italia.

AMSAT-I News, bollettino periodico di **AMSAT Italia**, viene redatto, impaginato e riprodotto in proprio. Esso viene distribuito a tutti i Soci.

La Redazione di **AMSAT-I News**, è costituita da:

Paolo Pitacco, IW3QBN

Segreteria

Francesco de Paolis, IK0WGF

Hanno collaborato a questo numero:

Francesco de Paolis, IK0WGF

Alberto Trentadue, IZ0CEZ

copertina:

Alcune immagini dell'evento ARISS a Mondragone, a sinistra la Principessa Elettra Marconi, nella fila a destra, Dieter Isakeit (ESA-ESRIN) [foto by: IK0WGF]

Tempo di rinnovi

Ricordiamo a tutti i Soci che è tempo di rinnovare l'adesione al Gruppo AMSAT Italia

La quota (minima) rimane invariata

26 euro !!!

Probabilmente il merito di questo è proprio nella nostra natura di Gruppo di volontari dediti alla sperimentazione e alla divulgazione scientifica, ma anche nella capacità e nella diplomazia di alcuni di noi come, primo fra tutti, il nostro Presidente Marco Lisi.

Grazie Presidente!

Il primo seme sparso a favore di questo accordo è stato posto in occasione del meeting AMSAT Italia svolto all'ESA/Esrin di Frascati a Dicembre 2008, dove proprio lo stesso Presidente ASI Ing. Saggese ha avuto modo di conoscerci e dove ha rivolto ai partecipanti parole di stima ed incoraggiamento per l'opera svolta degli appassionati come stimolo ed esempio per la comunità scientifica ed imprenditoriale.

Con questo esprimiamo un caloroso ringraziamento al Presidente Enrico Saggese per la grande opportunità che ci ha voluto concedere.

In sintesi, l'uso dei marchi AMSAT, incluso il nome e il simbolo AMSAT in qualsiasi forma, se non diversamente autorizzato da un contratto di licenza, costituisce una violazione dei diritti di registrazione sul marchio e può essere perseguibile a norma di legge negli Stati Uniti e/o internazionalmente da leggi di registrazione sul marchio equivalente in altri paesi.

Quindi, al fine di regolarizzare la posizione AMSAT Italia con AMSAT NA riguardo all'uso del marchio AMSAT, abbiamo preferito richiedere specifica autorizzazione e procedere alla registrazione del nostro logo secondo la normativa nazionale vigente.

Di fatto il logo AMSAT Italia così come lo conosciamo e come ora risulta registrato era già da tempo inserito nel sito Web AMSAT NA, nella pagina che elenca le organizzazioni AMSAT (partner) nel mondo.

Registrazione del logo AMSAT-I

Dal Verbale del Consiglio Direttivo di AMSAT Italia del 06 Ottobre 2009, avete potuto apprendere che era stato deciso di registrare il logo "AMSAT Italia" presso il Ministero dello Sviluppo Economico - Ufficio Italiano Brevetti e Marchi.

L'incarico di registrare il nostro logo presso l'UIMB è stato affidato al nostro Coordinatore, Emanuele D'Andria, IOELE, che per questa circostanza ha ricevuto una specifica lettera d'incarico dal Presidente, Marco Lisi, IZ0FNO.

Il logo registrato presso l'UIMB è quello ormai a noi familiare, costituito dal tradizionale globo stilizzato AMSAT, che per identità nazionale è colorato di blu e la freccia che lo circonda per i tre quarti è colorata con i nostri colori nazionali. A fianco al globo compaiono in basso a sinistra la lettera "I" (blu), come Italia e a destra la scritta **AMSAT ITALIA**, sempre di colore blu.

La decisione di registrare il nostro logo è stata presa dopo il parere positivo fornitoci da AMSAT NA, nella persona del Presidente, Barry

A. Baines WD4ASW, per l'uso e la modifica (parziale) del marchio AMSAT, che è protetto da leggi nazionali ed internazionali (Trademarks).

<http://www.amsat.org/amsat-new/links/Europe.php>

E' possibile effettuare on-line una semplice visura della registrazione del logo AMSAT Italia nella sezione "ricerche sul database dell'UIBM" del sito Web del Ministero dello Sviluppo Economico - Ufficio Italiani Brevetti e Marchi.

<http://www.uibm.gov.it/it/>

Troverete nel sito Web del UIMB due marchi AMSAT Italia. Il primo marchio è stato depositato il 26 novembre 1997, sei mesi dopo la costituzione ufficiale di AMSAT Italia, da parte di due persone, ma del quale non risulta essere stata rinnovata la registrazione alla scadenza decennale.

Il secondo marchio AMSAT Italia è in "nostro" ed è stato depositato il 25 novembre 2009 da AMSAT Italia dall'incaricato Emanuele D'Andria.

Questo mette fine ad una annosa ed inutile discussione basata esclusivamente su preconcetti e false interpretazioni

<http://www.amsat.org/amsat-new/notices/logos/index.php>

*Consiglio Direttivo e Segreteria
AMSAT Italia*

Analisi di un ricevitore SSB SDR realizzato con GNURadio

Seconda parte: i blocchi SDR

Alberto trentadue - IZ0CEZ

Eravamo rimasti...

Già! Chi si ricorda dove eravamo rimasti?

Tranquilli, potete sempre dare un'occhiata al numero precedente di Amsat-I News...

Nella prima parte abbiamo percorso tutta l'elaborazione del segnale RF SSB dall'antenna, passando per la daughter-board LFRX, i Mixed-Signal front-end (AD9862) che operano la conversione analogico/digitale e infine la FPGA Altera Cyclone, che realizza il digital down-converter (DDC).

L'uscita del DDC è un flusso di numeri a 16 bit che, nel nostro caso di ricezione di un singolo segnale SSB, sono i valori dei campioni del segnale alternati nelle componenti in fase (I) e in quadratura (Q).

Tali campioni, alla fine delle conversioni e decimazioni operate nel DDC, sono prodotti con una frequenza di campionamento pari a 256KHz^1 , per cui, complessivamente abbiamo un flusso di bit ad un rate di $256\text{K} * 2 * 16 = 8\text{Mbit/s}$.

Questo flusso di dati viene convogliato dal controllore USB Cypress FX2 sulla porta USB2 e reso disponibile al software sul PC.

Quella che descriverò è una implementazione di ricevitore SSB disponibile nella utility "**hf-explorer**"[4] disponibile tra gli esempi pronti all'uso di GNURadio.

Essa è solo una delle possibili realizzazioni di ricevitore SSB ed anche particolarmente semplificata. Realizzazioni più complesse e sofisticate sono ovviamente possibili.

La frontiera dell'SDR

Prima di proseguire con l'analisi, mi soffermerei un attimo, perchè è proprio a questo punto che avviene il salto concettuale dall'hardware al software.

Sì, è come se stessi attraversando un ideale posto di frontiera che richiede, come minimo, la consapevolezza che stiamo lasciando un terreno e stiamo entrando in un altro.

Fino a qui abbiamo analizzato il comportamento statico di componenti, integrati e logiche programmabili, che agiscono sulle grandezze elettriche in maniera anche molto complessa, e con un certo grado di configurabilità, ma non modificabile nell'architettura fondamentale.

In questo includerei anche la FPGA, pur essendo questa un componente programmabile e nonostante essa realizzi al suo interno filtri digitali DSP.

In effetti, al momento dell'avvio di qualsiasi programma che faccia uso della USRP, la FPGA viene programmata in maniera statica con un firmware.

Il comportamento di quel firmware è sempre lo stesso, qualsiasi sia la SDR che si voglia realizzare.

Invece, i dati trasmessi sulla porta USB2 e letti dal PC con opportuno driver, sono numeri puri e come tali vengono considerati dalle procedure DSP/SDR.

Da questo punto in poi non esiste alcuna architettura realizzata in hardware specificamente per l'elaborazione del segnale: l'architettura è definita da programmi scritti in un linguaggio opportuno e dai drivers sottostanti che astraggono l'interazione tra PC e USRP.

Questo aspetto è comune a tutti i sistemi SDR.

Nelle soluzioni più semplici (come ad es. i kit SoftRock) la parte hardware serve solo a filtrare il segnale e a convertirlo in banda base; l'effettiva conversione A/D viene realizzata dal codec della scheda multimediale del PC: opportune librerie del sistema operativo permettono di leggere i campioni in uscita dal codec e quindi di elaborarli con tecniche DSP/SDR.

Il nuovo terreno concettuale si basa quindi su questi due caratteristiche:

- 1 il "segnale" è composto da numeri puri (campioni);
- 2 la rappresentazione può utilizzare numeri complessi, cioè impiegare per ogni campione una coppia di numeri, la componente in fase e quella in quadratura.

Una cosa in comune con l'approccio analogico c'è: è possibile descrivere il segnale elaborato con una rappresentazione spettrale che mostra come è distribuita la potenza nel dominio delle frequenze.

E così come per la radio analogica, il ruolo dei diversi componenti (o procedure) SDR è definito in base all'effetto che essi hanno sullo spettro del segnale elaborato.

Tuttavia, nel caso della SDR, sono possibili manipolazioni dello spettro che sarebbero solo approssimate o addirittura impossibili nel mondo analogico, che è costretto ad utilizzare solo componenti reali e leggi fisiche.

E proprio questo è il punto di forza del DSP/SDR.

Nel seguito, per utilizzare la terminologia di GNURadio, indicherò con il termine **blocco** ciascun "componente" software dell'architettura SDR che analizzeremo.

Aspetti generali del software sull'host

Un'altra premessa importante è necessaria per inquadrare nel complesso gli aspetti operativi di GNURadio, prima di entrare nello specifico dei blocchi che compongono il ricevitore in esame.

Ricordo brevemente che la piattaforma software GNURadio è organizzata come una libreria di procedure modulari, chiamate appunto blocchi.

La modularità di queste procedure deriva dalle seguenti caratteristiche:

- 1 Sono descritte dal numero di ingressi, numero di uscite, dal tipo di dato utilizzato dai dati (campioni) in ingresso e in uscita, e dalla funzione che svolgono;
- 2 Un sistema DSP/SDR realizzato con GNURadio è la

1 I più attenti ricorderanno che, nella prima parte, la frequenza di campionamento di uscita dalla USRP era di 250 kHz. Tuttavia nell'analisi di questo ricevitore, l'utilizzo di una frequenza di 256 kHz risulterà più comodo, come vedremo. D'altronde, programmando la USRP con una decimazione di 250, questo è facilmente ottenuto.

connessione di uscite di blocchi con ingressi di altri blocchi, senza alcuna altra interazione tra essi.

Per maggiori dettagli, vi rimando all'articolo su GNUradio apparso sul bollettino di Marzo/Aprile [2].

Una SDR di GNURadio è rappresentata da un programma in linguaggio Python² indicato come **top block**.

Qualunque sia il sistema DSP/SDR realizzato dal top block, esso compie le seguenti operazioni:

- Definisce il sistema "connettendo" i diversi blocchi l'uno a l'altro, proprio come in un diagramma di flusso. I blocchi di GNUradio sono gerarchici, nel senso che possono essere ottenuti dalla connessione di blocchi più elementari. Perciò un blocco ha definito il metodo `connect()` che prende come argomenti i blocchi subordinati da connettere.
- Connette i blocchi "interfaccia" che rappresentano le risorse hardware dell'host e delle periferiche ad esso connesse. Tali periferiche includono ovviamente la USRP e la scheda multimediale che permetterà l'ascolto attraverso gli altoparlanti del computer o in cuffia.
- All'avvio, inizializza e configura le periferiche. Il blocco relativo alla sorgente USRP invocherà le librerie native sottostanti che configureranno la connessione USB2 verso la Cypress FX2 e configureranno la USRP, settando tra l'altro la frequenza del DDC.
- Legge i campioni da quelle che li generano (sorgenti), e sincronizzando il trasferimento dei campioni da un blocco all'altro, realizza gli algoritmi di elaborazione, fino ai blocchi terminali che consumano i campioni (scarichi). La scheda audio è uno scarico.

GNUradio fornisce un livello di astrazione estremamente semplificato che permette di ignorare dettagli a basso

livello ed di interagire con l'hardware utilizzando pochi parametri. La forte modularità dei blocchi permette di utilizzarli senza dover necessariamente conoscere come siano realizzati.

Le conoscenze richieste all'utilizzatore sono i fondamenti del DSP/SDR, le tecniche di modulazione e demodulazione e il linguaggio Python. Se poi viene usata l'interfaccia grafica **GNUradio Companion**, anche l'uso di Python diventa completamente trasparente.

Nella fig.1 diamo uno sguardo di insieme al diagramma di flusso che definisce il software SDR che demodulerà il nostro segnale SSB.

Il segnale in ingresso

Lo spettro del segnale campionato a valle del DDC e trasmesso sulla USB è descritto nella fig.2.

Notiamo subito che è uno spettro composto di repliche, come avviene sempre per i segnali campionati, spaziate della frequenza di campionamento F_{s0} , ovvero la frequenza di campionamento presentata dalla USRP all'ingresso del primo blocco software SDR.

In questo caso abbiamo scelto il minimo possibile, ovvero **256 kHz**. Un'altra osservazione interessante è la dissimetria rispetto alla banda base, e questo è reso possibile dal fatto che stiamo ricevendo non già campioni di un segnale reale ma coppie di valori di un segnale complesso, ovvero le componenti in fase I_i e in quadratura Q_i .

Ecco un primo esempio di una manipolazione che non è immediatamente possibile nel mondo analogico.

E' sì vero che anche nel mondo analogico si possono generare segnali in quadratura, ma ovviamente è necessario considerare due distinti segnali reali ed una maggiore complessità componentistica, cosa che non avviene nel DSP/SDR.

Nella prima parte si era detto che in generale la frequenza di conversione del DDC poteva differire dalla frequenza base del segnale SSB, un po' come avviene per il ricevitore a doppia conversione.

E infatti, è proprio ciò che accade nel ricevitore che stiamo analizzando: la conversione operata dal DDC della USRP possiamo vederla come una sintonia "grossa" che serve più che altro a selezionare la banda di interesse. Il primo blocco software opera la sintonia "fine".

Quindi nella fig.2, consideriamo il segnale SSB di interesse posizionato su di una frequenza base F_b . In figura sono anche presenti segnali indesiderati, per esempio modulazioni adiacenti (tratteggiate) e rumore di fondo.

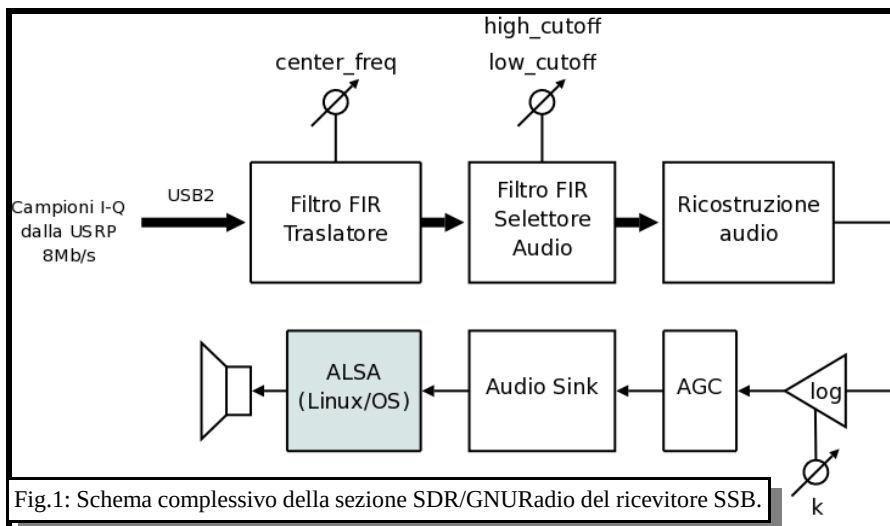


Fig.1: Schema complessivo della sezione SDR/GNUradio del ricevitore SSB.

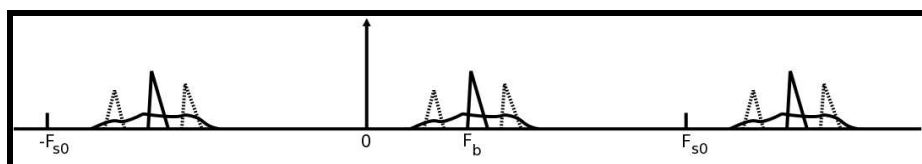


Fig.2: Spettro del segnale campionato in ingresso al filtro traslatore SDR.

2 Python è un linguaggio Object Oriented, perciò è possibile definire classi di oggetti dotati di un insieme di procedure che ne definiscono il comportamento e le interazioni con altri oggetti. Tali procedure prendono il nome di **metodi**.

I filtri FIR e GNURadio

Già nella prima parte di questo articolo avevo introdotto i filtri FIR. Riporto sinteticamente la struttura di tali filtri per interpretare al meglio la struttura del blocco software in esame. Un filtro FIR è formato da una serie di celle (taps) che ritardano il segnale di un'intervallo di campionamento T_s e lo moltiplicano per un fattore h_i (con $i=1..L$), essendo L un valore che va dalle decine fino al migliaio e più, detto ordine del filtro FIR.

I risultati di queste celle sono poi sommati insieme e presentati in uscita.

Perciò il comportamento del FIR è completamente definito dal set di coefficienti h_i usati nelle celle.

Come è facile intuire la scelta dei coefficienti h_i rappresenta un importante passo di progettazione.

L'operazione di sintesi dei coefficienti ha dei fondamenti teorici che si basano sulle proprietà matematiche delle sequenze numeriche e dei segnali campionati.

Possiamo però fare un'osservazione.

Supponiamo che il FIR abbia ordine L , e supponiamo che il segnale di ingresso $x(i)$ sia una sequenza di zeri, eccetto il primo valore $x(0) = 1$.

Un tale segnale lo chiamiamo **impulso unitario**. E' facile vedere che, in queste condizioni, l'uscita $y(i) = h_i$, per $i=0..L-1$. Perciò i coefficienti h_i sono i campioni della **risposta impulsiva**. Questa semplice osservazione ci fa intuire come la forma della risposta impulsiva e le caratteristiche dinamiche del suo spettro in frequenza siano determinanti per i valori h_i e viceversa.

Esistono procedimenti abbastanza complessi che permettono, a partire da caratteristiche dinamiche della risposta impulsiva, quali la **frequenza di taglio**, la **banda di transizione**, l'**ampiezza massima del ripple** nella banda passante ed altri ancora, di determinare l'ordine minimo L e i coefficienti h_i .

Nella pratica ci si muove proprio come nell'analogico, ovvero si ricorre ad opportune tabelle standard o si usano software di sintesi. La buona notizia è che GNURadio ha una utility di sintesi integrata nella libreria che ci libera da qualsiasi complessità di carattere matematico.

Ecco come è possibile in uno script python ottenere i coefficienti della risposta impulsiva di un passabasso:

```
xl_taps = gr.firdes.low_pass (xl_gain,
                              samp_rate, cutoff,
                              transition, windowing)
```

La variabile `xl_taps` è una lista di numeri reali restituita dalla chiamata alla sintesi di un passabasso `gr.firdes.low_pass` passando come parametri:

- un fattore moltiplicativo costante (**gain**)
- la frequenza di campionamento in gioco (**samp_rate**)
- la frequenza di taglio (**cutoff**)
- l'ampiezza della banda di transizione (**transition**)
- la funzione di windowing da applicare (**windowing**)

La **funzione di windowing** è da interpretarsi come un filtro in cascata al FIR teorico, che ne riduce gli effetti di non idealità.

Pensando alla forma di un passabasso, ad esempio, si potrebbe pensare di definire un filtro con risposta in frequenza solo fino ad una certa banda passante, poi nulla (un *sogno*, direi...).

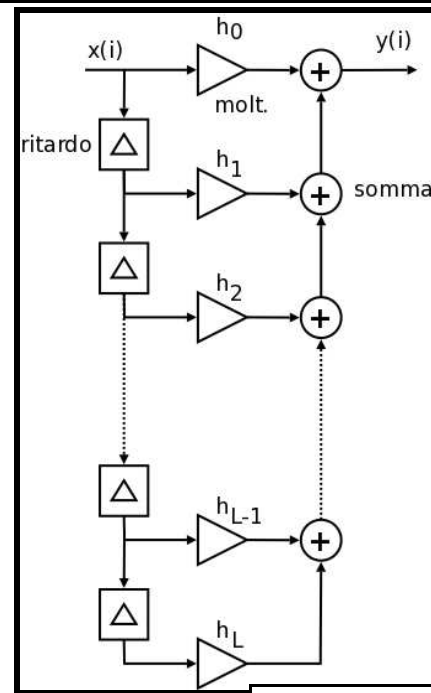


Fig. 3: Struttura di un filtro FIR.

Per una proprietà delle trasformazioni di Fourier, una tale risposta in frequenza si potrebbe ottenere solo con una risposta impulsiva illimitata nel tempo passato e futuro, cosa ovviamente impossibile nella realtà. Per contro, una risposta impulsiva reale, finita nel tempo e tutta compresa negli istanti $i \geq 0$, non ha le proprietà desiderate e, anzi, in prossimità del limite della banda passante scelta, presenta delle oscillazioni (*ripple*) molto pronunciate che degradano la qualità della risposta. Per questo sono state studiate una serie di funzioni di correzione, da sovrapporre alla risposta in frequenza, che riducono al minimo gli effetti di bordo e massimizzano l'idealità nella banda di interesse. Alcune note finestre sono quella di Hamming, di Blackman e di Hann, ciascuna con pregi e compromessi. Rimando ad una letteratura più qualificata per il loro approfondimento.

Il filtro FIR traslatore

Guardando la fig.2, è evidente che la prima cosa da fare sia riportare il segnale di interesse in banda base e filtrare i segnali indesiderati. La prima di queste operazioni è il compito del primo blocco software SDR, il **filtro FIR traslatore** (in GNURadio è il "frequency translating FIR filter").

In realtà il filtro FIR traslatore è veramente simile al DDC che abbiamo visto nella FPGA. Esso applica un filtro FIR passabasso, riduce ulteriormente la frequenza di campionamento e trasla in frequenza il segnale. Quando si riduce la frequenza di campionamento (decimazione), si riduce della stessa misura la banda massima utilizzabile dal segnale campionato (banda di Nyquist = $\frac{1}{2} F_{camp}$), e perciò è necessario pre-filtrare il segnale per eliminare tutte le componenti spettrali che possono generare aliasing.

Ecco come in GNURadio si utilizza un blocco di questo tipo:

```
self.xlate = gr.freq_xlating_fir_filter_ccf
(decim, xl_taps, freq_x, samp_rate)
```

Notare il suffisso "**ccf**": in GNURadio, esistono varianti dei

blocchi in base al tipo di dato che viene utilizzato.

“ccf” significa: “ingresso complesso, uscita complessa, coefficienti FIR reali (float)”.

La chiamata alla procedura ritorna il puntatore ad un blocco (oggetto) di GNURadio, passando i seguenti parametri:

- Il fattore di decimazione desiderato (**decim = 8**)
- i coefficienti del FIR (**xl_taps** sintetizzati dalla utility come al paragrafo precedente)
- la traslazione da applicare (**freq_xl**)
- la frequenza di campionamento d'ingresso (**samp_rate = 256000**).

Nel nostro esempio, i coefficienti del FIR (**xl_taps**) sono stati ottenuti passando a `gr.firdes.low_pass` le seguenti specifiche:

- **gain = 1**
- **cutoff = 16000 (16 kHz)**
- **transition = 4000 (4kHz)**
- **windowing: Hamming**

La decimazione pari a 8 fa sì che i campioni in uscita al filtro siano campionati a **F_{s1} = 256 / 8 = 32 kHz**, che è una frequenza gestibile da una qualsiasi scheda multimediale per PC.

La frequenza di cut-off viene quindi scelta pari alla banda di Nyquist (16kHz).

Il parametro **freq_xl** diventa proprio quel “comando di sintonia fine” che traslerà la modulazione di interesse, e quindi sarà valorizzato con **F_{xl} = -F_b** (notare che se la frequenza centrale usata nel DDC della FPGA è minore della frequenza di sintonia, **F_{xl}** deve essere un valore negativo).

Viene quindi naturale pensare di utilizzare **freq_xl** come parametro modificabile dinamicamente durante l'esecuzione del programma. Non mostrerò come questo avvenga, ma vi assicuro che con GNURadio si tratta di una cosa molto facile.

La struttura interna del FIR traslatore è mostrato dallo schema in fig.4

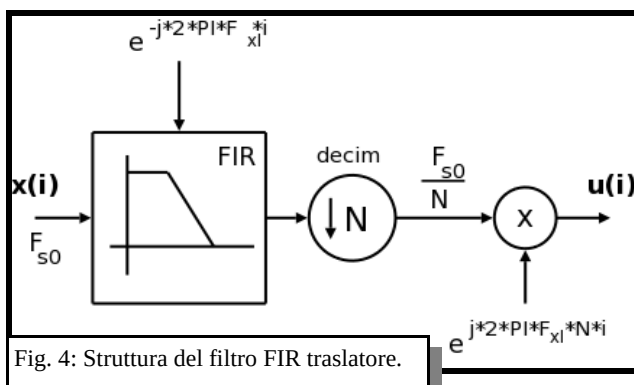


Fig. 4: Struttura del filtro FIR traslatore.

Il segnale viene subito filtrato dal passabasso FIR.

La risposta in frequenza di questo passabasso viene però traslata del valore opposto alla frequenza di traslazione **F_{xl}**, moltiplicando i suoi coefficienti per una opportuna fase rotante **exp(-2j*i*δ*F_{xl})**.

Questo è necessario perchè la banda di Nyquist dovrà essere garantita al segnale dopo la traslazione, perciò essa deve essere calcolata a partire dalla frequenza che, una volta traslata, si ritroverà in banda base.

Essa è proprio **F_b = -F_{xl}**.

Dopo la decimazione per 8, il segnale viene traslato in banda base dall'eterodina complessa **exp(2j*i*δ*F_{xl})**.

Da notare un altro esempio di manipolazione possibile solo nel mondo campionato. La banda passante del FIR è stata traslata su **F_b** passando in eterodina i coefficienti della risposta impulsiva. Una sorta di passabasso controllato in frequenza. Un simile approccio non ha equivalenti nel mondo analogico.

Un'altra nota implementativa del filtro è che in realtà la decimazione è integrata nel FIR stesso.

Questo si ottiene facilmente facendo sì che la somma dei prodotti (vedi fig.3) venga effettuata solo una volta ogni N scorrimenti degli ingressi attraverso i ritardi.

In effetti, l'operazione di decimazione serve a ridurre al minimo il carico computazionale del sistema, e perciò è utile sfruttarlo sin dall'inizio.

I passi di elaborazione sono mostrati nella fig.5 : al n.1, è mostrata risposta in frequenza del FIR traslata sul segnale di interesse per filtrare la banda di Nyquist; al n.2, il segnale traslato in banda base.

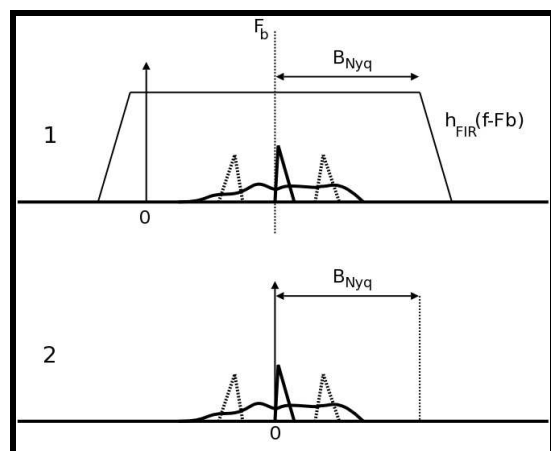


Fig. 5: L'elaborazione del filtro FIR traslatore.

Notare come un passabasso reale traslato da un esponenziale complesso si trasformi in un passabanda centrato su **F_b**.

Il FIR selettore audio

A seguire il FIR traslatore viene inserito un **filtro audio**. Anche questo filtro è un FIR e anche per esso viene sfruttata l'utility **firdes** per ottenere i coefficienti della risposta impulsiva a partire dalle specifiche del filtro:

```
audio_coeffs =
gr.firdes.complex_band_pass(gain,
samp_rate, low_cutoff, high_cutoff,
transition, windowing)

self.audio_filter =
gr.fir_filter_ccc(decim, audio_coeffs)
```

Si può notare che si tratta di un filtro passabanda dotato di funzione di decimazione. Non viene utilizzata ulteriore decimazione, per cui **decim = 1**.

Per i rimanenti parametri vale:

- **gain = 1**
- **samp_rate = 32000 (sarebbe F_{s1} = 32kHz)**
- **transition = 100 (Hz)**
- **windowing = Hamming**

Il filtro audio ha una duplice funzione.

La prima è quella di selezionare la modulazione di interesse ed eliminare tutte le modulazioni adiacenti ed altri disturbi che sono passati attraverso gli stadi precedenti.

Da questo punto di vista, questo filtro corrisponde ad un selettore audio, come si può trovare, sotto forma di manopola, nelle apparecchiature di ricezione dotate di filtri audio ad alte prestazioni.

Anche in questo caso si utilizzano i parametri **low_cutoff** e **high_cutoff** come valori dinamici variabili durante l'esecuzione del programma.

L'ampiezza del filtro dovrebbe in genere essere intorno ai 3 kHz.

La seconda funzione del filtro la scopriremo al paragrafo successivo.

La la ricostruzione audio

Probabilmente questa parte del sistema è quella che impressiona di più per quanto sia drasticamente più semplice rispetto all'omologa analogica.

Questo blocco ha il compito di ricostruire il segnale modulante audio dal segnale SSB riportato in banda base dai blocchi precedenti.

Notare come la ricostruzione dell'audio avvenga in banda base, a differenza dei ricevitori analogici, che generalmente operano la ricostruzione nella media frequenza.

Il blocco in questione è il **complex_to_float()**.

```
c2f = gr.complex_to_float()
```

Tutto quello che fa questo blocco è leggere il flusso di dati complessi in uscita al filtro audio e considera solo la parte reale del segnale. Vale a dire:

$$u(i) = \text{Re}(x(i)) = \text{Re}(I(i) + jQ(i)) = I(i)$$

Sì, TUTTO QUI! Viene da pensare a quanto era laborioso ricostruire la SSB in uscita dal mixer nei cari ricevitori analogici...

Da un punto di vista spettrale la spiegazione è semplice ed è descritta simbolicamente dalla fig.6 .

La componente in fase è simmetrica rispetto all'origine $F=0$, mentre quella in quadratura è antisimmetrica.

La traccia sotto l'asse delle frequenze per $Q(f)$ indica che la fase delle componenti spettrali differisce di 180° dalle corrispondenti della componente $I(f)$, a parità di frequenza.

Il modulo è corrispondente per entrambe le componenti.

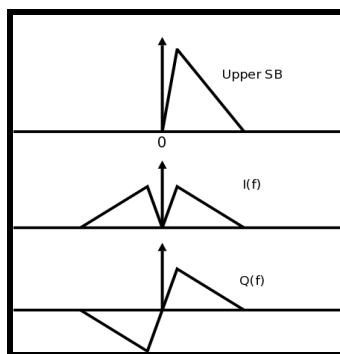


Fig. 6: Decomposizione del segnale SSB.

La singola banda laterale perciò contiene intrinsecamente tutte le informazioni per ricostruire il segnale modulante reale.

E' interessante anche notare che questo stadio ricostruisce indifferentemente sia la banda laterale superiore che quella inferiore.

La selezione della banda laterale di interesse è proprio la seconda funzione svolta dal selettore audio descritto in precedenza.

Perciò, se si vuole la banda laterale superiore, **low_cutoff** e **high_cutoff** saranno positivi (ad es. **low_cutoff=0** e **high_cutoff=3000**); per la banda laterale inferiore, **low_cutoff** e **high_cutoff** saranno negativi (ad es. **low_cutoff=-3000** e **high_cutoff=0**).

Quindi abbiamo un'implementazione della selezione della banda laterale che è forse la più semplice ed intuitiva ma che, nel mondo analogico, avremmo scartato perché avrebbe richiesto filtri audio dalle prestazioni dinamiche irraggiungibili.

Nel mondo DSP, invece, quasi tutto diventa possibile!

Volume e AGC

L'ultimo blocco del nostro ricevitore gestisce il livello audio. All'ingresso (vedi fig.1), il controllo del volume è un attenuatore logaritmico variabile, che passa da un'attenuazione di -100dB ad un'attenuazione nulla. Un simile attenuatore si ottiene banalmente con un blocco moltiplicatore con un fattore 10^{-vol} , essendo **vol** una variabile controllata dinamicamente:

```
self.scale =
gr.multiply_const_ff(math.pow(10., -vol))
```

Lo stadio AGC, come noto, adatta l'ampiezza del segnale in modo da evitare saturazione agli stadi di amplificazione. In questo modo viene aumentata la dinamica complessiva del ricevitore, rendendo il segnale audio più gradevole e comprensibile.

Nei ricevitori analogici, questo stadio viene in genere realizzato nella media frequenza ed è controllato dal livello medio del segnale audio nello stadio finale.

Nel ricevitore SDR in esame, invece, esso lavora tutto a livello audio.

Ad essere precisi, non è un vero e proprio AGC, essendo privo di reazione e quindi non controllato della reale dinamica dell'hardware a valle.

Tuttavia applica un semplice principio *feed-forward* che di fatto mantiene il livello al volume medio percepito dall'orecchio al momento della regolazione del livello fatta con il controllo del volume visto sopra.

La struttura interna del blocco AGC è descritta dallo schema in fig.7.

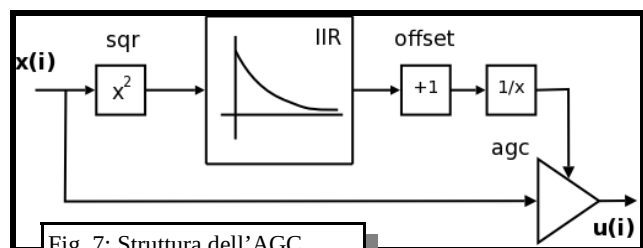


Fig. 7: Struttura dell'AGC.

Sostanzialmente viene letta la potenza istantanea (**sqr**) del segnale audio in uscita dalla ricostruzione; tale valore viene filtrato da un filtro IIR e i valori numerici in uscita vengono sommati all'unità (**offset**).

Il risultato della somma viene utilizzato come fattore di attenuazione del segnale audio medesimo (**agc**).

La chiave del funzionamento è tutta nel filtro IIR, di cui do un breve cenno al solo scopo di capirne il ruolo in questa sezione.

Cenno sugli IIR

I filtri IIR si differenziano dai FIR per il fatto che l'uscita è data non solo dalla somma dei prodotti dei campioni ritardati dell'ingresso (vedi fig.3), ma anche dalla somma dei prodotti dei campioni ritardati dell'uscita stessa. In altre parole, l'uscita è messa in feedback su se stessa come se fosse un'altro ingresso. La struttura è mostrata in fig. 8.

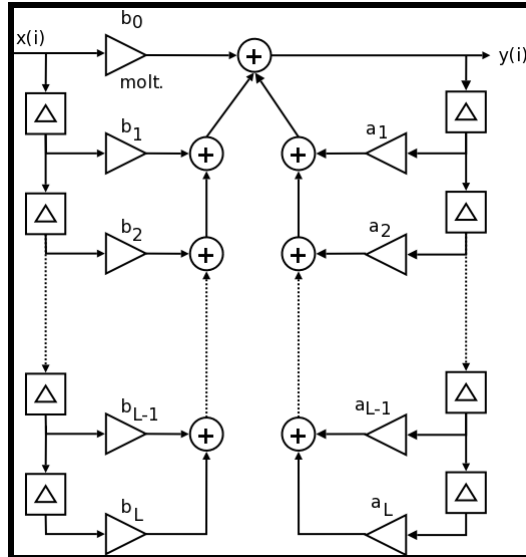


Fig. 8: Struttura canonica dei filtri IIR.

A dispetto dell'apparente maggiore complessità, i filtri IIR sono quelli più vicini al nostro caro mondo analogico, in quanto la risposta all'impulso unitario è illimitata nel tempo, esattamente come avviene nelle funzioni di trasferimento dei filtri analogici che, ricordiamo, sono combinazioni di esponenziali e oscillazioni.

E in realtà i filtri IIR sono affini ai filtri analogici, al punto che, dato un filtro analogico, è possibile trovare il corrispondente filtro IIR con il medesimo comportamento nel mondo campionato.

Nella sezione AGC che stiamo esaminando, viene usato un IIR del primo ordine, cioè con $L=1$, e avente i seguenti coefficienti: $b_0 = 0.04$, $b_1=0$ e $a_1=0.999$.

```
iir = gr.iir_filter_ffd ( [.004, 0], [0, .999] )
```

Osservando la fig.8, si vede subito che in queste condizioni, la risposta impulsiva diventa:

$$h(n) = b_0 * a_1^n = b_0 * e^{-n \log(1/a)}$$

ovvero equivale ad un filtro analogico del primo ordine con la tipica risposta impulsiva ad esponenziale negativo (quando $a_1 < 1$).

In buona sostanza, è una specie di condensatore "caricato" dal valore della potenza segnale audio: in presenza di variazioni improvvise del livello audio, il filtro reagisce modificando in senso opposto il guadagno del blocco moltiplicatore finale.

L'euristica dietro è che il livello di volume più frequente (settato manualmente con il controllo del volume) è quello che l'orecchio percepisce come buono.

Non certo rigoroso, ma funzionale.

La terminazione audio

Finalmente, i campioni del segnale in uscita dall'AGC vengono passati all'interfaccia verso la scheda multimediale.

Come detto, le interfacce che consumano i campioni al loro ingresso vengono chiamate scarichi (**sinks**).

GNURadio definisce il blocco **audio.sink** che rappresenta l'interfaccia verso il sottosistema multimediale del sistema operativo.

```
dst =
audio.sink(long(self.af_sample_rate))
```

Nel caso di Linux, questo corrisponde generalmente (ma non esclusivamente) con l'architettura **ALSA**, che realizza un servizio di astrazione verso l'hardware multimediale.

In particolare, esso accetta campioni audio e li riproduce attraverso i codec della scheda e in questo modo è possibile ascoltare la modulazione sia in cuffia che in altoparlante.

La cosa interessante è che i sink di GNURadio sono tutti uguali dal punto di vista dell'interfacciamento con il sistema SDR. Per cui nulla vieterebbe di indirizzare i campioni audio non già alla scheda multimediale ma, per esempio, ad un file, per realizzare una registrazione in formato PCM.

Oppure si potrebbero incapsulare i campioni in pacchetti dati da inviare su Internet per realizzare un ascolto remoto...

Insomma la SDR offre possibilità limitate solo dalla fantasia dello sperimentatore.

La costruzione della SDR

Nei paragrafi precedenti abbiamo analizzato i sei blocchi che compongono la parte software del ricevitore SSB. E' interessante dare uno sguardo a come in GNURadio si connettano i vari blocchi una volta definiti, ovvero come si "costruisce" la SDR.

Avevamo visto come il programma Python che definisce la SDR in GNURadio sia un oggetto `top_block`. Ebbene, il `top_block`, definisce il metodo `connect()` che opera proprio la connessione uscita/ingresso dei blocchi definiti. Le chiamate che seguono sono proprio quelle che realizzano il ricevitore SSB analizzato:

```
self.tb.connect(src, self.xlate,
self.audio_filter, c2f, self.scale)
self.tb.connect(self.scale, (sqr, 0))
self.tb.connect(self.scale, (sqr, 1))
self.tb.connect(sqr, iir, offset, (agc, 1))
self.tb.connect(self.scale, (agc, 0))
self.tb.connect(agc, dst)
```

Il metodo `connect()` accetta un numero variabile di blocchi, fintanto che l'uscita di ognuno si collega all'ingresso del successivo.

Quando il cammino del segnale si biforca o si riunisce, allora è necessaria un'ulteriore chiamata alla `connect` per specificare a quanti e quali ingressi deve essere inviata l'uscita di un blocco, o quali uscite devono essere collegate agli ingressi multipli di un blocco.

La connessione uscita/ingresso dei blocchi non richiede alcuno sforzo di sincronizzazione del flusso dei valori dei campioni.

E' la piattaforma GNURadio che si prende carico di tutto ciò, facendo sì che la struttura realizzata mediante le varie chiamate connect() sia una vera e propria "catena di montaggio" su cui circolano i campioni elaborati.

La versione integrale del programma python che realizza questo ricevitore lo trovate in [4].

Credo che questo articolo sia la prova di come GNURadio avvicini il mondo della SDR e del DSP alla sperimentazione, rimuovendo gran parte di quegli ostacoli che generalmente tengono lontano i non "addetti ai lavori". A parer mio, GNURadio è un'opportunità per fare esperienza di SDR e di scoprire nuovi terreni per estendere le nostre conoscenze con l'approccio amatoriale che ci contraddistingue.

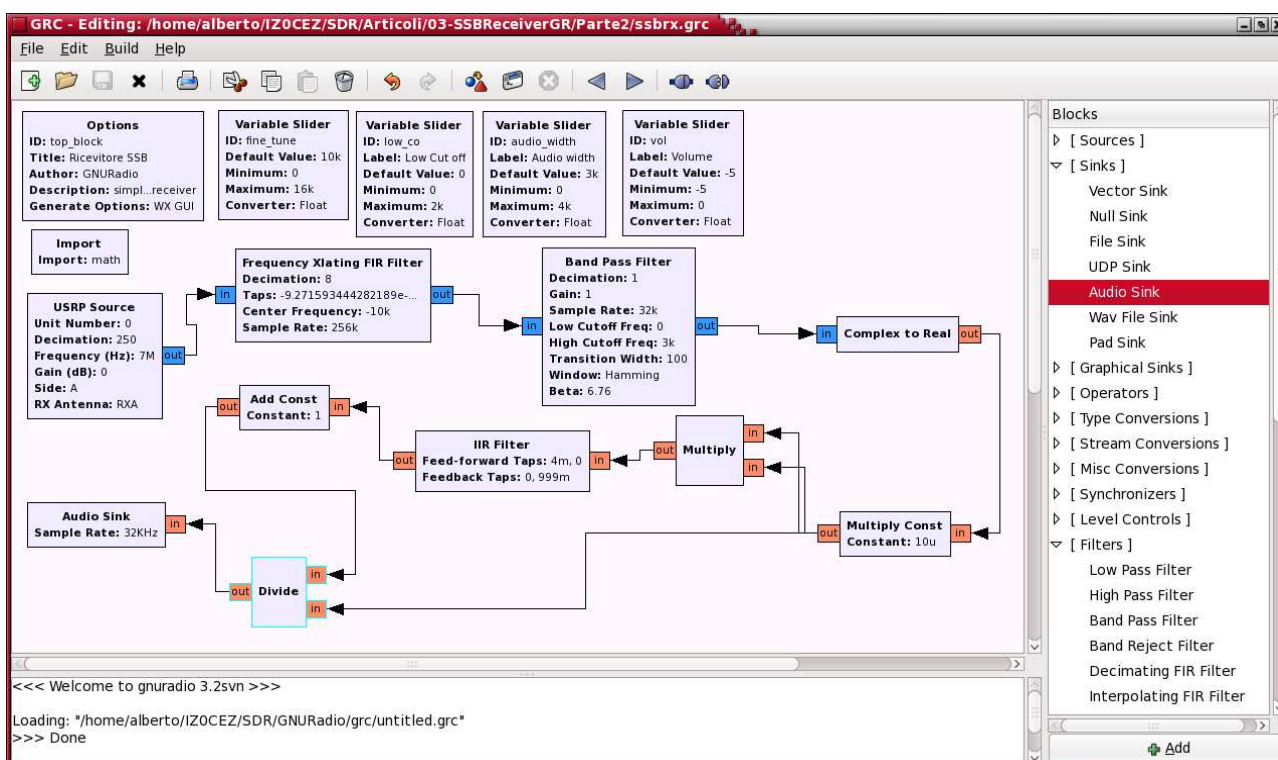
Conclusione

L'analisi di questo ricevitore SSB ha avuto il duplice scopo di mostrare alcune tipiche soluzioni di elaborazione del segnale digitale e quello di presentare alcuni strumenti che GNURadio mette a disposizione per realizzarle. Ovviamente questa realizzazione non ha l'ambizione di essere un ricevitore ad alte prestazioni, ma ovviamente è un buon punto di partenza perfettamente funzionante.

Vale la pena ricordare che GNURadio mette a disposizione uno strumento grafico (**GNURadio Companion**) per poter costruire SDR senza dover avere particolari conoscenze di programmazione e di Python. La figura che segue è infatti il ricevitore SSB descritto nell'articolo ricostruito in maniera visuale con GNURadio Companion.

Riferimenti:

- [1]: Sito di GNURadio <http://gnuradio.org/trac>
- [2]: "Introduzione a GNURadio" su AMSAT-Italia News di Marzo/Aprile 2009
- [3]: Haddad-Parsons – Digital Signal Processing – Computer Science Press – 1991
- [4]: Applicazione "hf-explorer" di GNURadio: http://gnuradio.org/trac/browser/gnuradio/trunk/gnuradio-examples/python/apps/hf_explorer/hfx2.py



Avviso importante

Come annunciato precedentemente, questo è l'ultimo numero "cartaceo" del Bollettino AMSAT-Italia. Come deciso dall'Assemblea dei Soci (marzo 2009) e dal CD AMSAT-Italia, a partire dal numero 1 gennaio-febbraio 2010 (Volume 18) la pubblicazione sarà

esclusivamente elettronica

Il Bollettino non chiude, ma passa ad una veste moderna e potrà essere scaricato direttamente da ogni Socio, in formato PDF, usando una password fornita dalla Segreteria.

Continua la descrizione realizzativa della radio "diversa" dal solito.

DCRTX "verso" la radio digitale - #2

Paolo Pitacco - IW3QBN

Premessa

Non so a quanti sia risultato interessante il mio primo articolo su questo nuovo tipo di radio realizzabile in casa, con pochissimi (o addirittura niente) strumenti, ma siccome quello era l'inizio, ecco la continuazione ...

Dopo aver visto la parte di generazione del segnale RF e come controllare la radio stessa, passo a descrivere la parte analogica vera e propria (o almeno quello che "sembra" essere ancora molto analogico).

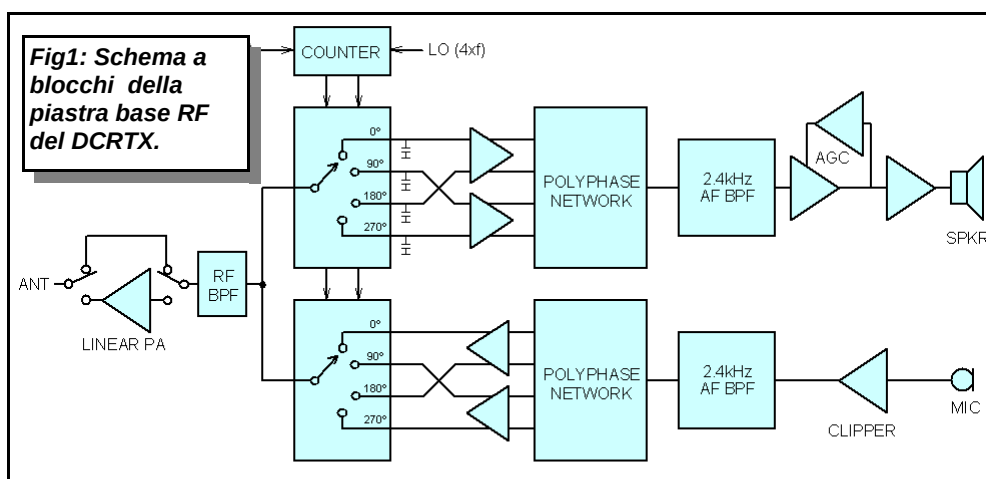
Scheda base RF

La scheda CPU/DDS vista nella prima parte era essenziale per gestire la parte radio vera e propria che sta invece su questa seconda scheda.

Lo schema a blocchi era quello riportato nella figura #2 dello scorso articolo (l'avete guardata o ci siene passati sopra al volo?) e per quanto semplice, di cose a bordo ne ha diverse, e non tutte "note" come vedrete!

Per evitare di andare a cercare, la ripropongo nuovamente nella figura 1, un pò più ingrandita della volta scorsa.

Potete distinguere immediatamente che la parte centrale è praticamente simmetrica ed uguale: in alto la catena del ricevitore, in basso quella del trasmettitore.



In entrambe le catene ci sono dei filtri e dei commutatori veloci, e poi altri circuiti specializzati per la ricezione (amplificatore b.f. e altoparlante) oppure per la trasmissione (preampl. microfonico e modulo amplificatore lineare di potenza, P.A.). Comune alla prima coppia di circuiti commutatori c'è un contatore; la sua caratteristica è di poter contare "su e giù" ovvero UP/DOWN.

Vedrete poi che questa caratteristica servirà a far decidere se si opera in USB (banda superiore) o LSB (banda inferiore), semplicemente mettenso a 0 oppure a 1 (logici) un pin.

Lo schema elettrico della scheda base RF è riportato nelle pagine che seguono.

Descrizione del Ricevitore

Il segnale che arriva all'antenna viene portato attraverso un filtro passa-banda all'ingresso di un commutatore veloce, che è poco conosciuto come "mixer di Tayloe" (dal nome del suo inventore, che era anche radioamatore, nominativo N7VE) ma molto di più come "H mixer" (con questo nome trovate molti articoli) citato ad esempio, nei cataloghi di chi vende materiale per radioamatori.

La caratteristica di questo mixer è di essere un "campionatore analogico" del segnale che gli viene applicato all'ingresso; operando ad una frequenza pari a 4 volte quella del segnale d'ingresso, fornisce esattamente i valori della tensione (vettorialmente parlando) relativamente ai punti di fase 0°, 90°, 180° e 270°.

Le quattro tensioni sono "memorizzate" su alcuni condensatori (uno per ogni punto di fase), ma risulta evidente che se vogliamo disporre dei campioni (4) di un segnale a 5MHz, il commutatore dovrà operare a $5 \times 4 = 20\text{MHz}$. Questo è il primo "punto debole" del sistema di Tayloe, ovvero che volendo ricevere un segnale a 30MHz (estremo alto della banda HF) dovremo avere un commutatore in grado di operare a $30 \times 4 = 120\text{MHz}$!!!

Ad oggi non esistono in commercio (o disponibili agli OM) commutatori così veloci, per cui nel mio caso, e come già

indicato nella descrizione del circuito CPU/DDS, ho deciso di limitare il funzionamento del ricetrasmittente ai 10MHz.

La parte più divertente e complessa viene dopo: le tensioni relative ai quattro punti di fase, devono essere combinate per ricostruire il segnale originale (modulante).

Molto probabilmente finora avrete notato la presenza di molte soluzioni (e soprattutto in quelle SDR) di elaborazioni su due sole

fasi, "In fase" e "in Quadratura" (dove la frequenza del mixer è "soltanto" doppia a quella del segnale da demodulare), ma qui non è che si è voluto strafare, piuttosto si è cercato di risolvere in altro modo il problema dell'elaborazione successiva.

Nelle SDR infatti, usando i soli segnali I/Q, non è possibile fare a meno di una certa "elaborazione" che di solito viene affidata ad un PC (scheda audio), ma questo significa che non abbiamo una sola scatola (Rx) ma anche il PC e relativi cavi, alimentazione e software; in questo sistema non serve il PC e nessun software, resta solo una scatola! Disponendo di 4 fasi, si è potuto usare la tecnica dei filtri analogici chiamati "polyphase" che si comportano proprio come degli sfasatori analogici che permettono alla fine,

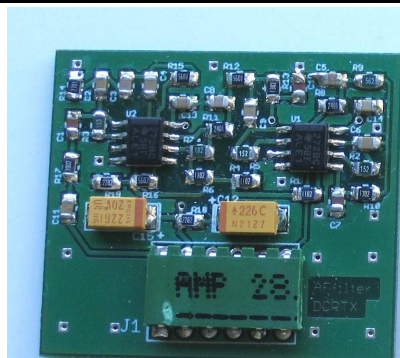


Fig.2: Circuito del filtro audio con OP-AMP del DCRTX.

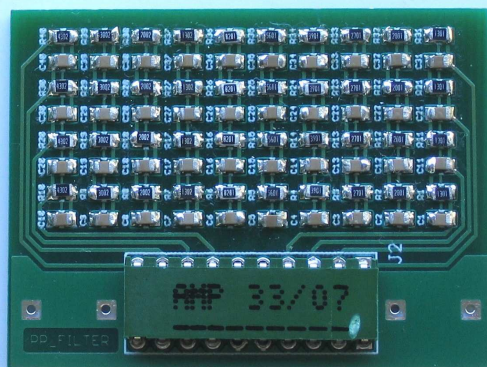


Fig.3: Circuito del filtro polifase a 10 sezioni (4 fasi) del DCRTX.

combinando le loro uscite senza l'uso di circuiti complessi o costosi (leggi filtri a quarzo), di recuperare la forma originale del segnale modulante.

Questi filtri sono semplicemente dei gruppi di resistori e condensatori, opportunamente collegati tra loro; una volta calcolati i valori dei componenti (con tolleranze "umane") basta avere la pazienza di saldarli (e sono tanti!) e non c'è bisogno di taratura alcuna.

Le uscite di questo filtro possono essere inviate a dei normali amplificatori operazionali (connessi come amplificatori) ottenendo la ricostruzione del segnale originale. Per migliorare la resa (e la selettività, ovviamente) l'amplificazione viene fatta su una banda di frequenze (da 300 a 2400 Hz, ovvero la banda audio che ci interessa), mediante dei classici filtri attivi, anch'essi facilmente calcolabili, realizzabili e di sicuro funzionamento.

Per questi due circuiti, che sono uguali sia per la ricezione che per la trasmissione, ho realizzato dei piccoli circuiti stampati che vanno "inseriti" in quello più grande, principale (vedi figure 2 e 3) mediante dei connettori verticali (come visibile in figura 5, a sinistra la parte rx e a destra quella Tx).

Nella catena di amplificazione si trova un rivelatore che svolge la funzione di controllo di guadagno (AGC), e

questo consente di avere anche un segnale da fornire al microcontrollore per la misura dell'intensità relativa del segnale (RSSI o Smeter) e quindi visualizzarlo.

La cosa sorprendente è la sensibilità del tutto, considerando che non c'è alcun preamplificatore o circuito accordato (limitatamente alla banda passante scelta per il filtro d'ingresso a componenti fissi e quindi non sintonizzabile), paragonabile ad un apparato commerciale di normali prestazioni (ma sicuramente più costoso).

Descrizione del Trasmettitore

In senso inverso a quello appena descritto, la parte trasmettitore; il segnale del microfono viene preamplificato, filtrato, sfasato dal filtro polifase e poi applicato al commutatore veloce (mixer di Tayloe) per diventare il segnale RF modulato. Anche in questo caso, il filtro audio e quello polifase, sono gli stessi del ramo ricevitore.

Come già anticipato, il ricetrasmettitore opera in SSB, e la scelta della banda da utilizzare (U o L) viene fatta semplicemente cambiando il senso di conteggio del segnale dal DDS, ovvero facendo cambiare il senso di rotazione del commutatore! Questa operazione è "indolore" in quanto il contatore è del tipo UP/DOWN e basta quindi commutare il livello logico di un pin; nessun filtro, diodo o altro complicato o costoso meccanismo!

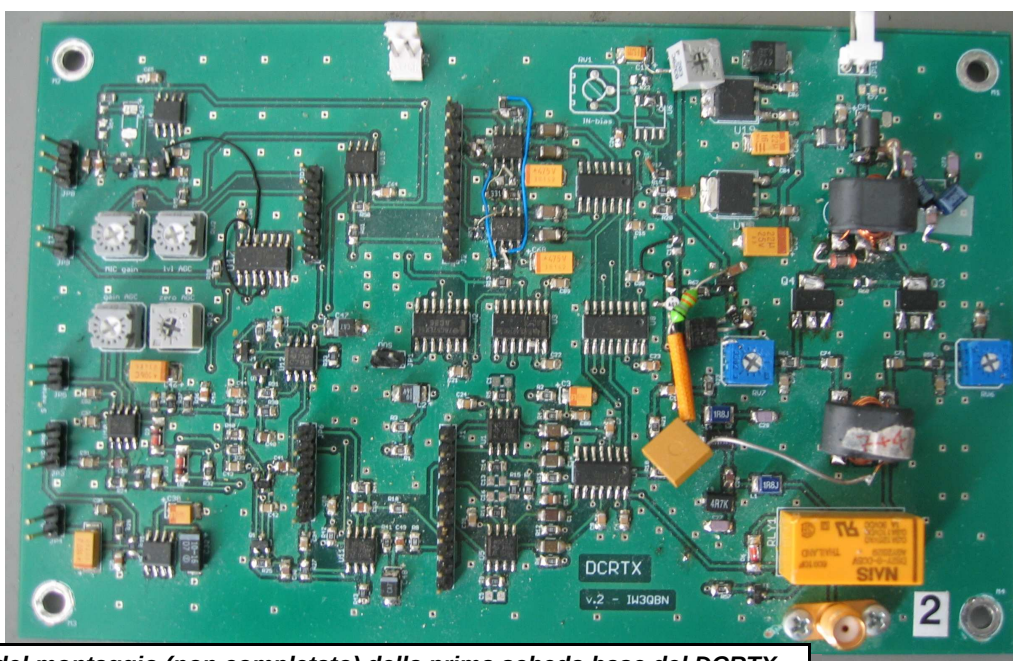
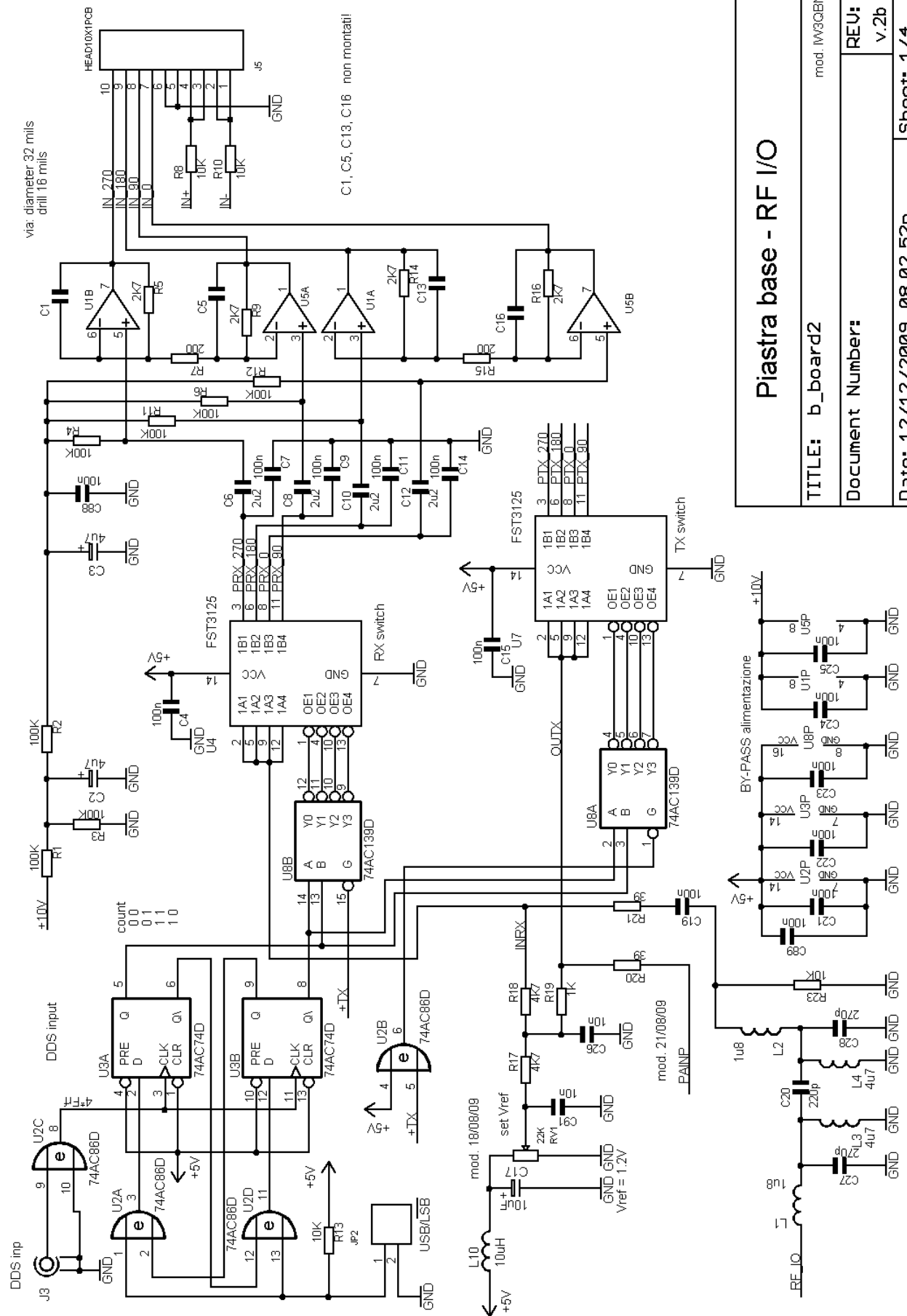


Fig.4: Vista del montaggio (non completato) della prima scheda base del DCRTX.



Piastra base - RF I/O

TITLE: b_board2

mod. IW3QBN

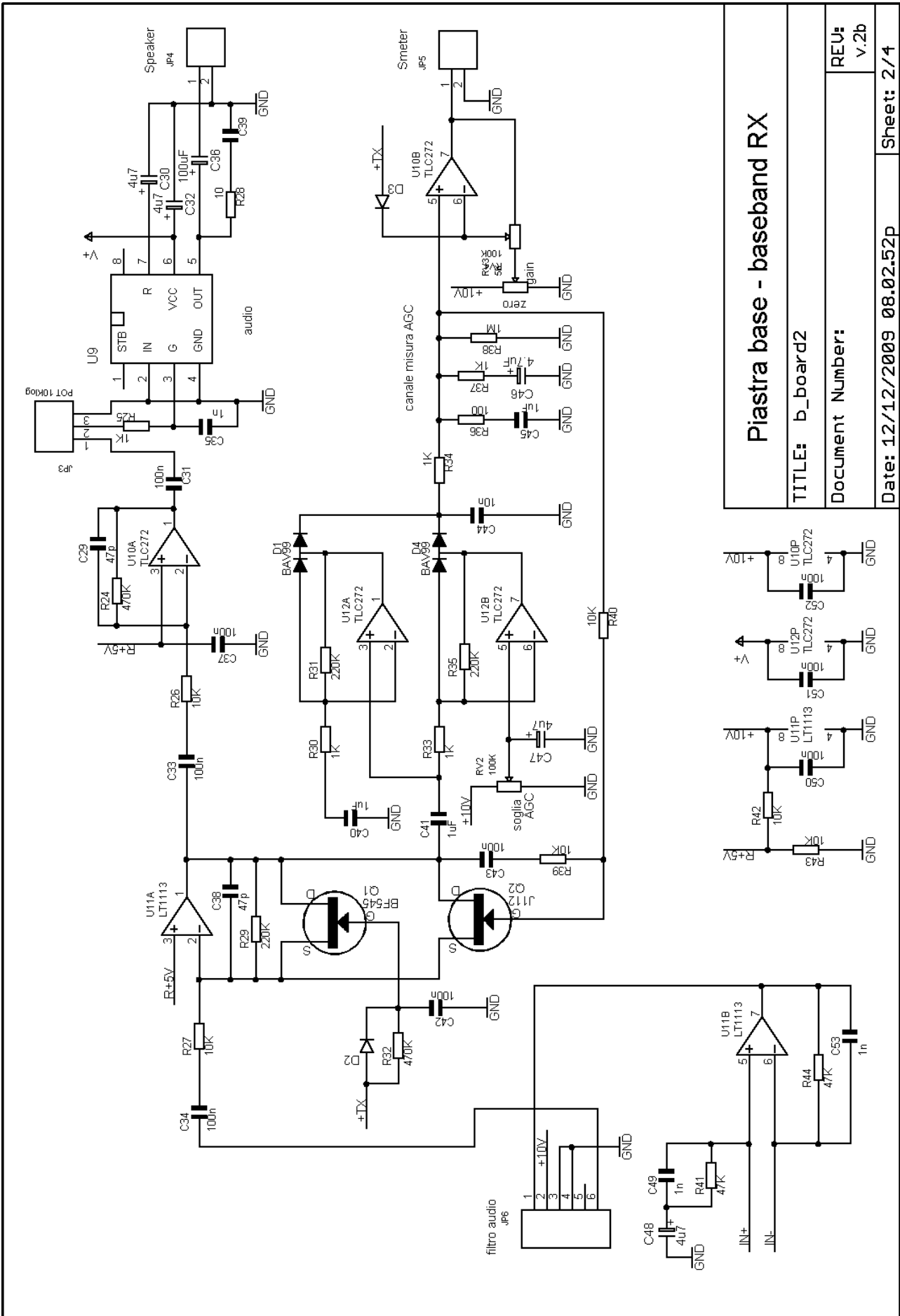
Document Number:

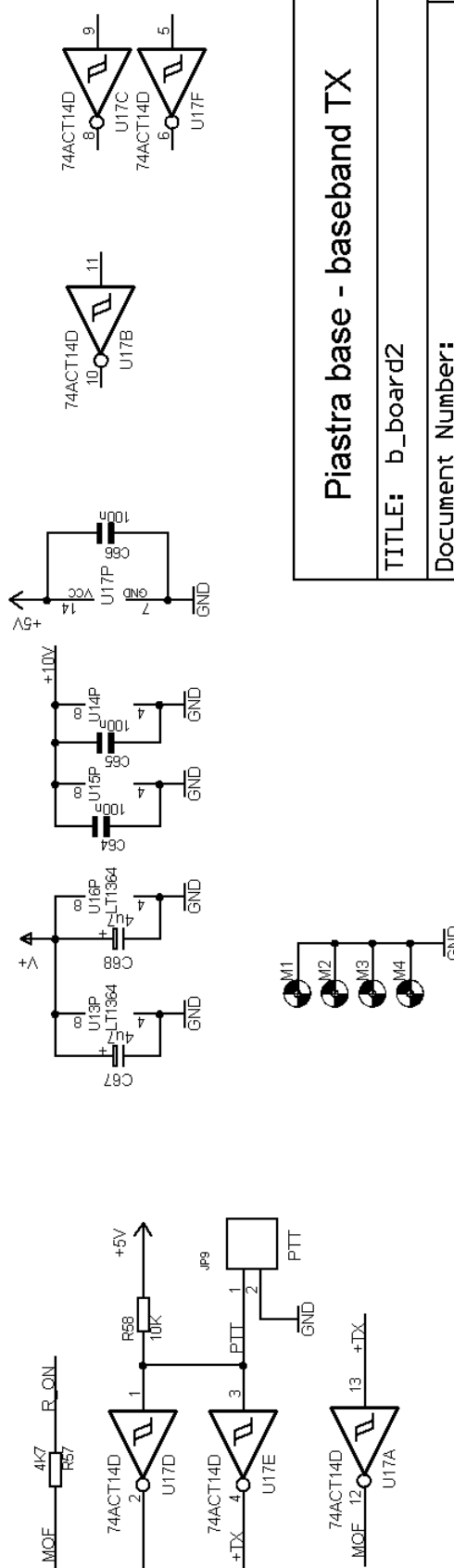
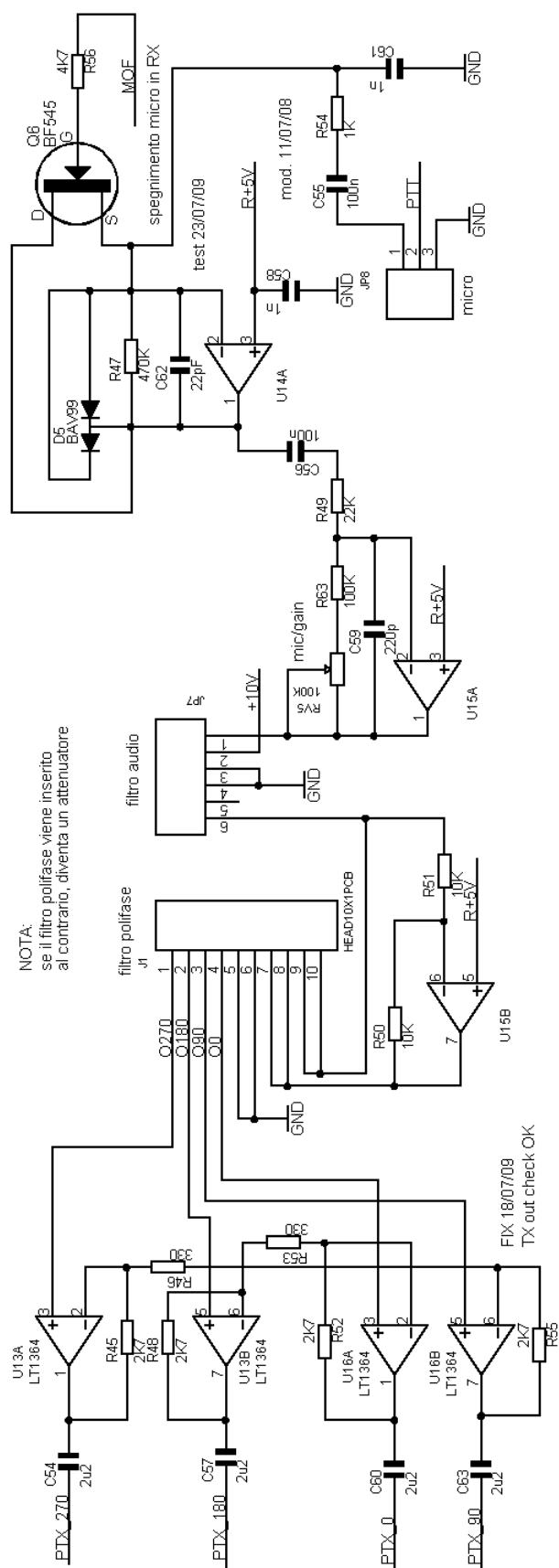
REV:

v.2b

Sheet: 1/4

Date: 12/12/2009 08:02.52p





Piastra base - baseband TX

TITLE: b_board2

Document Number:

Date: 12/12/2009 08:02.52p

Sheet: 3/4

Non è richiesta alcuna taratura se non una verifica del livello del microfono (ma se esagerate, saranno i vostri corrispondenti a farvelo diminuire!).

In un primo momento avevo pensato di portare il controllo di guadagno del microfono sul pannello frontale della mia realizzazione, poi ho scelto la strada del trimmer su circuito stampato, regolato una volta per tutte in base al microfono. Per evitare un fastidioso "autoascolto", ho inserito lo spegnimento del preamplificatore quando il circuito è in ricezione, usando un FET come visibile nel foglio #3.

Il segnale RF in uscita dal mixer è intorno a +0~1 dBm, quindi necessita di uno stadio preamplificatore per arrivare ai +10dBm (10mW) per pilotare un circuito finale.

La parte PA (amplificatore di potenza) ha richiesto un pò di prove ed esperimenti, ma è realizzata direttamente sulla scheda base e la descriverò nel prossimo articolo.

Realizzazione

Credo non sia difficile capire come il tutto sia un tantino "complicato" per essere fatto dal nulla; per questo motivo, anche in questo caso, ho disegnato un circuito stampato realizzato poi da una ditta specializzata, con tanto di "solder resist" che facilita la saldatura dei componenti *smd* evitando di fare cortocircuiti (basta un pò d'attenzione) e serigrafia con i nomi dei componenti (vedi figura qui sotto). Si tratta anche in questo caso (come per la scheda CPU/DDS) di un lavoro da fare con pazienza, in quanto ho usato quasi esclusivamente la tecnologia *smd*, sia per modernità che per riduzione delle dimensioni (che NON sono esasperatamente ridotte, anche se sarebbe stato possibile); sicuramente un vero "esperimento radio".

Basteranno delle normali attenzioni per saldare i chip, mano ferma ed un pò di pazienza; questa volta le dimensioni dei pin dei componenti "critici" sono molto più larghe di quelle del chip DDS!

Ho previsto l'uso di connettori e di header per poter rendere la scheda "riutilizzabile" per aggiornamenti e modifiche, senza dover ricorrere al saldatore ad ogni smontaggio!

Quello che mi ha sorpreso dopo il montaggio del primo esemplare, è stata la facilità con cui si può ascoltare di tutto! considerando che non ci sono componenti speciali da tarare dopo aver avuto paura per saldarli (tipo i GaAsFET), complice il fatto che in HF le stazioni abbondano, sia sui 40 che sugli 80 metri, basta un pezzo di filo per sentire il baccano che c'è

Molto più interessante il fatto che sia possibile ascoltare anche "fuori gamma", ad esempio segnali RTTY commerciali o stazioni WEFAX (specie quelle per i servizi marittimi); ovviamente sulla frequenza che si ascolta si può anche trasmettere ... e questo rende interessante il circuito in quanto qualunque banda amatoriale può essere coperta senza modifica alcuna.

Il problema è quello di mantenere un livello di potenza decente su tutte le gamme, o meglio, sull'intera copertura del ricevitore. Io ho optato per un compromesso, visto che commutare filtri (anche se fissi sul circuito stampato) non mi piaceva: poca potenza (2W max) ma consumo ridotto per essere "portatile".

Sul tema DCRTX ho aperto una discussione sul forum AMSAT Italia, Area tecnica -> DCRTX, a cui invito gli interessati a partecipare.

Per quanti fossero interessati, non dispongo di "kit" per la realizzazione (a parte i circuiti stampati), ma tutto il materiale utilizzato è reperibile senza particolari difficoltà e anche via posta (non serve avere in negozio sotto casa). Per i circuiti stampati, contattatemi qui:

iw3qbn@amsat.org

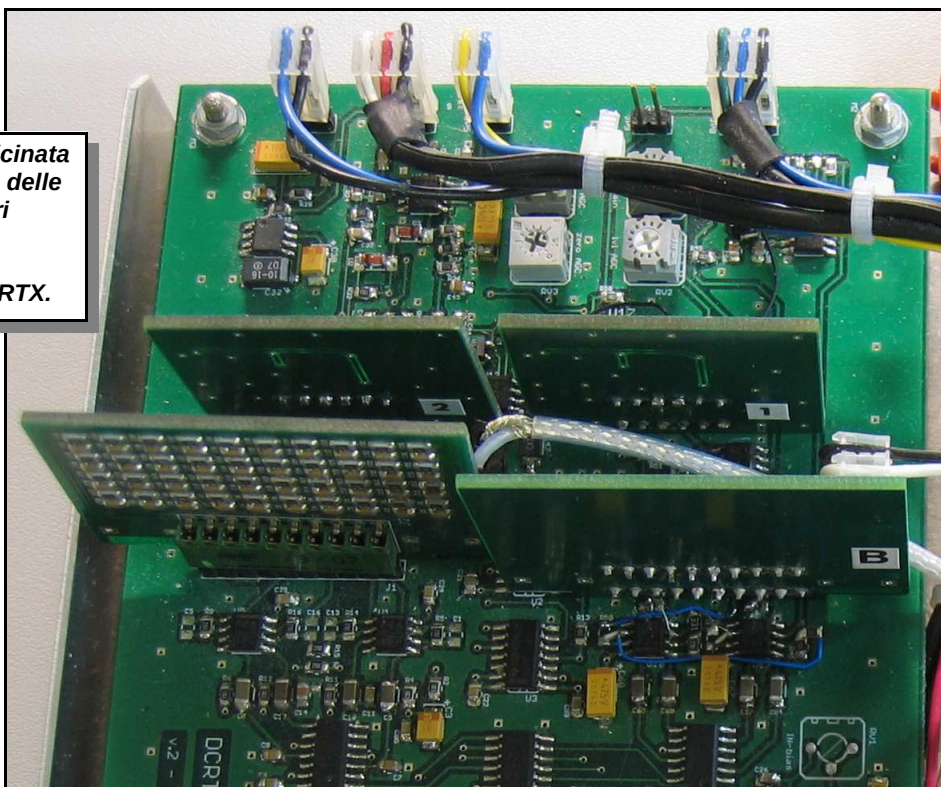
Riferimenti:

Direct Conversion HF transceiver - www.nikkemedia.fi/hohtola/dc-trcvr/DCradio.htm

Dan Tayloe N7VE - Product Detector

Understanding and Designing Sequence Asymmetric Polyphase Networks - W.J. Niessen, PA2PIM vers. 4.0

fig.5: Vista ravvicinata dell'inserimento delle piastrine dei filtri audio e dei filtri polyphase nella piastra base DCRTX.



Notizie Associative

Commemorazione di G. Marconi (centenario conferimento del premio Nobel)

In Giugno, una nostra delegazione ha partecipato al Meeting Internazionale di ARISS presso l'ESTEC in Olanda ed aveva proposto di commemorare il centenario del premio Nobel a Guglielmo Marconi attraverso la ISS.

L'idea originale era quella di usare l'equipaggiamento ARISS per un broadcast dalla ISS che avrebbe distribuito la voce e le immagini (SSTV) di Guglielmo Marconi.

Causa alcuni inconvenienti tecnici degli equipaggiamenti di bordo l'idea iniziale di broadcast è stata abbandonata e convertita in ARISS school contact.

Nell'impossibilità di pianificare l'evento/celebrazione entro il 2009 la NASA ha riservato per questa circostanza uno "slot" nella seconda settimana di Gennaio 2010.

Candidata a partecipare con i suoi studenti a questo evento è stata l'Università di Roma "Tor Vergata".

Qui di seguito è riportato il rapporto redatto per questo evento ARISS:

Giovedì, 7 Gennaio 2010 alle ore 09:00 UTC, 10:00 ora locale, è stato stabilito un contatto ARISS tra le Università di Roma "Tor Vergata" e l'astronauta della NASA Jeffrey Williams KD5TVQ a bordo della Stazione Spaziale Internazionale.

Il contatto radio con la ISS è stato eseguito via telebridge tramite la stazione ARISS di terra W6SRJ presso il Junior College di Santa Rosa, California, USA.

Il collegamento è stato organizzato per commemorare il Premio Nobel per la Fisica a Guglielmo Marconi ha offerto nel 1909.

Alle ore 09:00 UTC, stabilito il contatto con NA1SS, l'Astronauta Jeffrey Williams KD5TVQ ha risposto a 11

domande degli studenti di "Tor Vergata" e della Scuola Media "Enrico Fermi" di Monteporzio Catone.

L'Astronauta Jeffrey Williams ha affermato che l'equipaggio della ISS apprezza i contatti ARISS con le scuole, così come il contributo della comunità radioamatoriale per la promozione della divulgazione tecnica e scientifica tra i giovani.

Dopo la sequenza di domande degli studenti, la principessa Elettra Marconi, figlia di Guglielmo Marconi e il signor Dieter Isakeit, di ESA ESRIN Corporate Communication Office & Communication and Knowledge Department, hanno inviato i saluti per l'equipaggio della ISS.

L'ARISS Mentor Francesco De Paolis, IK0WGF ha presentato le attività ARISS al pubblico.

A seguire l'evento è stato commentato dal Sig. Augusto Cramarossa dell'ASI (Agenzia Spaziale Italiana), responsabile della strategie nazionali ed internazionali, e dal Sig. Massimo Puccini, Vice Sindaco di Monteporzio Catone, e dalla Dottoressa Marina Ruggieri, Direttore Dipartimento di Ingegneria Elettronica - Università "Tor Vergata".

A Villa Mondragone, Centro Congressi dell'Università di "Tor Vergata", in Monteporzio Catone vicino Roma, ha seguito il collegamento un pubblico di oltre 200 persone composto da studenti, insegnanti, visitatori e giornalisti.

L'audio del collegamento è stato distribuito via EchoLink.

Congratulazioni allo staff dell'Università "Tor Vergata" e al team AMSAT Italia!

ARISS è un programma internazionale educativo svolto in collaborazione con le agenzie spaziali Nasa, RKA, ESA, CNES, JAXA, CSA, con le organizzazioni AMSAT e le organizzazioni IARU.

ARISS offre l'opportunità agli studenti di vivere l'emozione di un collegamento radioamatoriale diretto con i membri dell'equipaggio a bordo della Stazione Spaziale Internazionale. Insegnanti, genitori e pubblico di vedere, di prima mano, come i radioamatori e i membri dell'equipaggio a bordo della ISS possono stimolare l'interesse dei giovani nel campo della scienza, della tecnologia e dell'apprendimento.



AMSAT Italia sarà presente alla prossima edizione di
SAT EXPO 2010
 che si terrà presso la Fiera di Roma
 nei giorni
4 - 5 - 6 febbraio 2010

NOTIZIARIO AEROSPAZIALE

aggiornato al
23 gennaio

La nostra principale fonte di informazioni è l'autorevole rivista settimanale *Flight International*. Fonti aggiuntive di informazioni sono la rivista mensile *Spaceflight*, edita dalla *British Interplanetary Society*, ed alcuni notiziari elettronici, tra cui il *Jonathan Space Report*. Con questi siamo in grado di presentare una selezione di notizie sempre aggiornate con l'uscita del Bollettino.

Shuttle e ISS

L'attuale equipaggio (EXP22) della ISS è composto da Jeff Williams, Maksim Suraev, Oleg Kotov, T.J. Creamer e Soichi Noguchi; gli astronauti Suraev e Williams, il 21 gennaio, hanno effettuato lo spostamento della Soyuz TMA-16 dal boccaporto di coda del modulo Zvezda a quello in posizione zenitale del nuovo modulo russo Poisk (sopra quello Zvezda).

La Soyuz TMA-17 è agganciata al boccaporto sul nadir del modulo Zarya, mentre la Progress M-03M si trova sul modulo Pirs lasciando libero il boccaporto del modulo PMA-2.

Lo scorso 11 gennaio è stato usato il braccio robotizzato SSRMS per spostare il pallet che contiene parti di ricambio, ESP-3, dalla struttura P3 a quella S3, sul lato opposto della stazione.

Il 14 gennaio, alle 10:05 UTC gli astronauti Suraev e Kotov, indossando le tute Orlan-MK, sono usciti dal modulo Pirs per effettuare le operazioni di preparazione all'uso del modulo Poisk. Durante l'EVA sono stati abbandonati nello spazio due pannelli per l'isolamento termico. I due sono rientrati alle 15:49 UTC, completando la 24.a EVA russa della ISS.

2010 AL 30

Un oggetto catalogato come asteroide, sigla 2010 AL 30, è transistato a soli 130000 km dalla Terra lo scorso 13 gennaio. Esso ha un periodo orbitale esattamente di un anno solare, ma la sua

traiettoria è ellittica e lo porta a ripetuti passaggi ravvicinati del nostro pianeta e di Venere. Un esperto dell'ESA, Michael Khan, lo identificava come lo stadio finale Fregat della sonda Venus Express, che è transistato vicino a Venere all'inizio del 2006, ma l'orbita calcolata dal JPL Horizons, indica un passaggio a 1.6M km da Venere, quindi non vi sarebbe riscontro.

Le osservazioni successive del Minor Planet Center hanno dimostrato che si trattava proprio di un asteroide.

Stato delle sonde

Il primo gennaio 2010, Voyager 1 era a 112.1 AU e Voyager 2 a 91.0 AU dal Sole. La New Horizons a 15.5 AU dal Sole, tra le orbite di Saturno e Urano.

La sonda Cassini si trova in un'orbita di 155000 x 2300000 km intorno a Saturno, ed ha passato 5.5 anni nel suo sistema.

La sonda Dawn è in un'orbita solare di 1.50 x 1.94 AU a 6.5° d'inclinazione; all'inizio 2010 era a 1.75 AU dal Sole, e continua ad allargare la sua orbita usando i motori a ioni per raggiungere l'asteroide Vesta nel 2011.

Rosetta è in un'orbita solare a 0.98 x 5.09 AU e 4.4° d'inclinazione e si sta allontanando dalla Terra per avvicinarsi a Lutetia.

Mars Odyssey, Mars Express e Mars Reconnaissance Orbiter sono operativi in orbita attorno a Marte, mentre dalla superficie i due rover Spirit e Opportunity, continuano ad inviare dati.

La sonda giapponese Hayabusa è in un'orbita solare tra Marte e terra, ed è in attesa di iniziare la traiettoria di rientro verso la Terra, pianificata verso la metà di quest'anno.

La sonda Deep Impact Flyby è in un'orbita di 0.97 x 1.02 AU a 6.7° d'inclinazione per incontrare la cometa 103P/Hartley-2 il prossimo 11 ottobre.

Stardust si trova in un'orbita a 0.97 x 1.73 AU e 8.5° d'inclinazione per raggiungere la cometa P/Tempel 1 nel febbraio 2011.

Le sonde di ricerca Kepler, STEREO A, STEREO B, e Spitzer sono anch'esse in orbita solare ma vicine alla terra, mentre WMAP, Herschel e Planck si trovano nel punto di Lagrange L2, tra Terra e Sole, mentre SOHO, Wind e ACE su quello L1. Entro la misura di 1 AU, la sonda Venus Express continua ad inviare informazioni su Venere, mentre la Messenger si sta avvicinando a Mercurio per poter iniziare ad orbitarvi intorno nel marzo del 2011.

La sonda Lunar Reconnaissance Orbiter (LRO) sta girando attorno alla Luna, in un'orbita di 40 x 60 km e 90° d'inclinazione.

ARTEMIS P1 (ora THEMIS P1) era in orbita di trasferimento lunare, a 47058 x 356845 km e 59.5° d'inclinazione.

ARTEMIS P2 (ora THEMIS P2) era in un'orbita a 5518 x 258042 km e 6.5° d'inclinazione, e raggiungerà P1 nel percorso verso la Luna.

La sonda IBEX è in un'orbita a 8038 x 307710 km, e continua ad essere operativa. Attorno alla Luna orbita anche una piccola sonda giapponese, Ouna.

AVVISO IMPORTANTE:

Se non altrimenti indicato, tutti gli articoli pubblicati in questo bollettino rimangono di proprietà degli autori che li sottoscrivono. La loro eventuale riproduzione deve essere preventivamente concordata con la Redazione di AMSAT-I News e con la Segreteria di AMSAT Italia. Gli articoli non firmati possono considerarsi riproducibili senza previa autorizzazione a patto che vengano mantenuti inalterati.



AMSAT Italia

GRUPPO DI VOLONTARIATO

Registrazione Serie III F. n. 10 del 7 maggio 1997 presso Ufficio del Registro, Sassuolo (MO)

Riferimenti:

Indirizzo postale:

Segreteria: segreteria@amsat.it
Internet WEB: http://www.amsat.it

Consiglio Direttivo: cd@amsat.it

Presidente: iz0fno@amsat.org
Segretario: ik0wgf@amsat.org
Consigliere: i0kpt@amsat.org
Consigliere: iw3qbn@amsat.org
Consigliere: iw8qku@amsat.org

Pagamenti:

Tutti i pagamenti possono effettuarsi a mezzo:

Conto Corrente Postale: n° 14332340
Intestato a: AMSAT Italia

Codice IBAN: IT35 M076 0102 2000 0001 4332 340

Codice Fiscale: 930 1711 0367