

Telecomando di un RTX civile

Come rendere più “amatoriale” un ricetrasmittitore ad uso civile

Una premessa è d’obbligo, contrariamente al solito non si tratta di un progetto fatto e finito, ma di una serie di appunti il cui scopo è mettere in grado molti colleghi a realizzare in proprio il sistema proposto. A parte la disponibilità della strumentazione necessaria per intervenire sulla gamma di funzionamento della radio è necessario essere in grado di giocare con Arduino, cosa che non è tra le mie doti (mi sono semplicemente appoggiato a chi ne sa più di me), e avere il coraggio di intervenire relativamente a fondo su una radio che all’inizio dell’avventura funziona perfettamente. Questo per scoraggiare, senza cattiveria, ma per pura prudenza, chi “potrebbe far danni”.

Tutti modelli citati sono versioni funzionanti in VHF, resta inteso che la cosa è perfettamente attuabile sui corrispondenti modelli previsti per il funzionamento in UHF.

Su queste pagine, anche per colpa mia, sono apparsi più volte modifiche a vecchi RTX a uso civile per l’uso da parte di noi radioamatori.

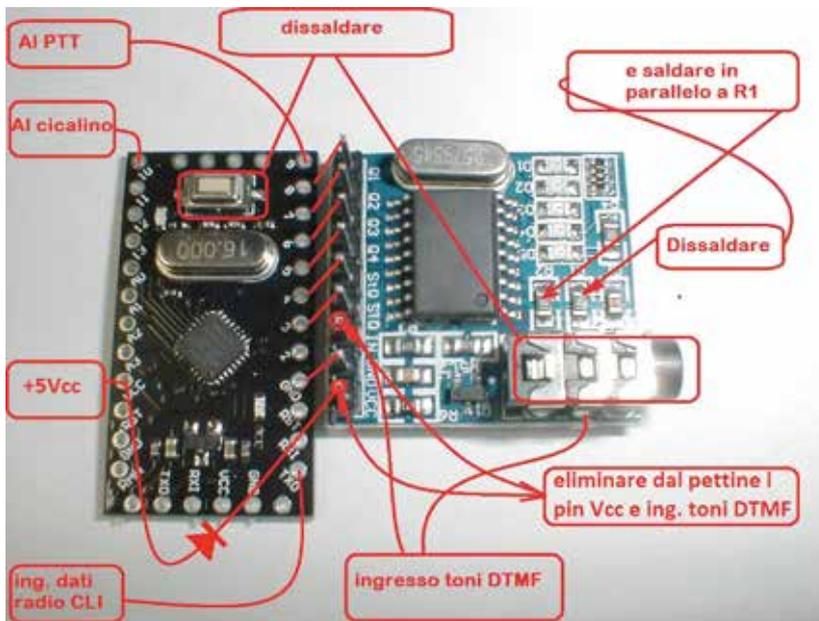
Di solito si tratta di RTX datati, che non possiedono più le omologazioni per l’uso civile e la cui conversione in gamma amatoriale è più o meno facile. Avendo a disposizione la strumentazione del caso per riallineare la radio e un PC datato per la riprogrammazione di solito la modifica è attuabile senza troppi problemi.

Questa tipologia di radio è caratterizzata da una forte limitazione in fatto di comandi. Ovvero l’utente del servizio civile non deve poter intervenire sulla radio, e le ragioni di questa necessità è evidente... non deve essere possibile modificarne la programmazione, il che ci porta a una delle differenze più evidenti rispetto alle nostre radio, non esiste un VFO, ovvero la radio funziona esclusivamente su “memorie” preimpostate e non modificabili. Ma non è la sola differenza, il frontale è sempre molto spartano, interruttore di accensione, controllo del volume, cambio dei canali, presa microfono. Niente altro.

Dato l’aspetto “dimesso”, diciamo poco evidente, il costo spesso molto basso, unito a una costruzione robusta, spesso... corazzata. Ne fanno l’oggetto ideale per l’uso in auto.

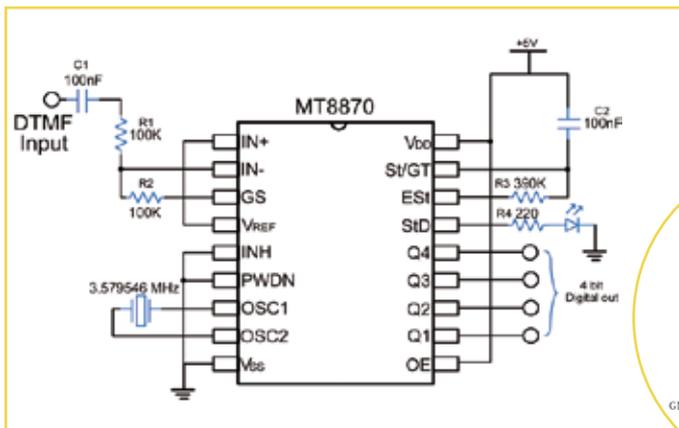
Un esempio, il Maxon modificato anni fa (è una delle modifiche più complicate in assoluto) aveva tra i dati di targa la possibilità di resistere a un rapporto di onde stazionarie pari a 50:1.

Foto Arduino e DTMF



Saldare i due stampati, dopo aver rimosso i pin di alimentazione e ing. toni, con i due lati saldature uniti, interponendo un francobollo di materiale isolante.

Portare l'alim. al modulo DTMF e alimentare Arduino tramite un diodo



Basic connection diagram

La proposta è dunque un telecomando, realizzabile in due modi diversi e applicabile a più tipologie di radio, volto a cambiare la memoria di utilizzo senza intervenire direttamente sul frontale della radio. Infatti, con pochissime eccezioni, queste radio non hanno la possibilità di allontanare il frontale dal corpo della radio. Fa eccezione, per quanto ne so, alcuni vecchi modelli Icom (IC U400), il Philips FM1000, la serie 9000 Simoco e l'Alan HM135, caratterizzati comunque da un frontale non propriamente minuscolo unito alla necessità di un cavo dedica-

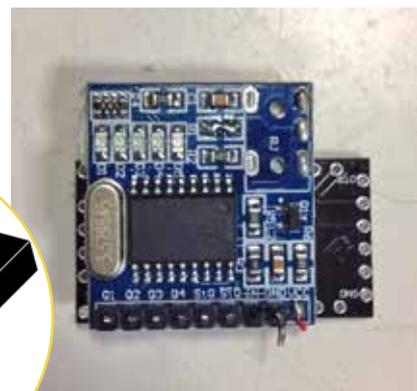
to. Attenzione però... dato che radio diverse hanno, come è ovvio, caratteristiche diverse quanto alle possibilità di comando il mio scopo è di fornire indicazioni utili a più colleghi possibile, pur soffermandomi a quanto realmente realizzato.

Confesso che utilizzare in un progetto un "ardu-coso" mi infastidisce non poco, ma non c'erano altre possibilità per realizzare queste idee... dunque tutte le soluzioni proposte richiedono l'impiego di una schedina Arduino, di solito Arduino Mini che pur non avendo a bordo l'interfaccia di programmazione USB è il più piccolo e il più economico della serie, del resto anche le versioni dotate del processore 168 assolvono perfettamente il compito che gli è richiesto.

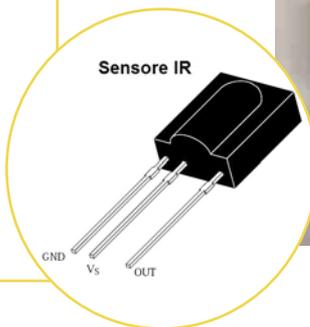
La prima soluzione è stata di ricorrere al microfono DTMF,

decodifico i toni e li utilizzo per comandare la radio. Questa possibilità sembra la più facile da attuare, ma così non è... alcune radio posseggono, come accessorio, il microfono dotato di DTMF Accessorio che potrebbe avere un costo non indifferente. Il vecchio (!) HM100TN, ora HM152TN, di casa Icom costa, come ricambio originale da 60 a 80 €. In verità sono reperibili esemplari di evidente produzione cinese a prezzi competitivi, anche sotto i 15 €, il problema che potrebbe capitarci un esemplare che emette toni DTMF fuori standard, fatto che rende il nostro acquisto completamente inutile. Altri produttori, Kenwood ad esempio, forniscono un microfono DTMF che non emette toni DTMF, mi spiego meglio... Il microfono comanda la logica della radio la quale si incarica di emettere i toni richiesti che, dunque non vengono generati nel microfono. Questo fatto aggiunge un gradino di difficoltà in più al nostro intento, dato che è necessario determinare il punto nel corpo radio in cui sono presenti i due toni DTMF.

La seconda possibilità è molto più economica, si tratta di ricorrere a un telecomando infrarosso. Sono in vendita esemplari minuscoli, dotati dei tasti essenziali e normalmente forniti anche del fotodiodo necessario a realizzare il ricevitore. Il costo di questi oggetti è decisamente basso, con 10€ rischiamo di portarne a casa tre o più... Del resto nulla ci impedisce di riciclare un vecchio



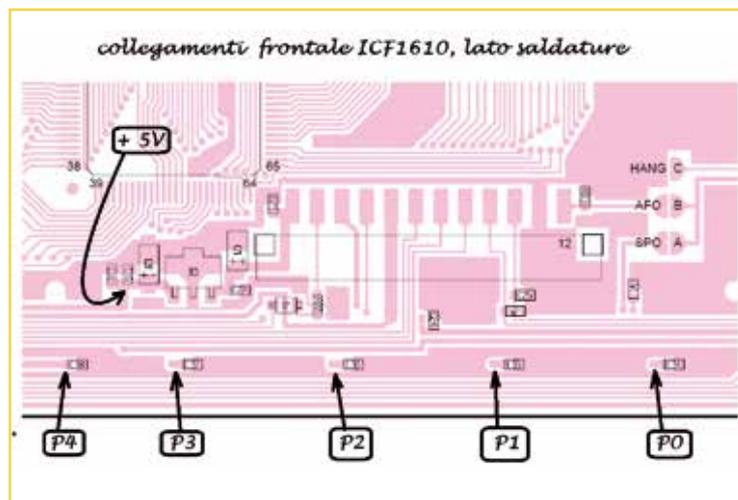
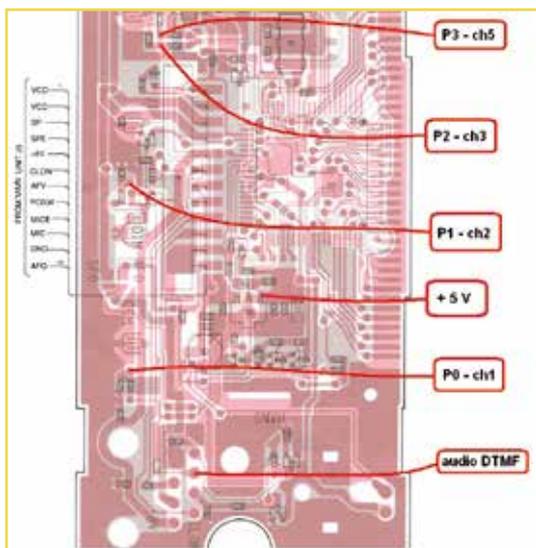
Arduino e il modulo DTMF montato



telecomando in disuso, anche quello della vecchia TV o del vecchio decoder che ci hanno costretti a buttare... Dalla medesima fonte è recuperabile anche il sensore IR, il che riduce ancor più il costo, già basso, della realizzazione. E' una buona idea acquistare, o recuperare, telecomando e sensore IR insieme, per essere sicuri che i due abbiano la medesima frequenza di funzionamento.

Una volta che abbiamo in mano il telecomando è necessario decodificare i codici emessi da questo nel momento in cui premiamo un pulsante. La cosa andrà realizzata nuovamente impiegando un modulo Arduino. In rete sono reperibili numerosi esempi adatti al nostro scopo.

Qualunque sia stata la nostra scelta è il momento di capire come comandare la nostra radio, e qui è ora necessario distinguere la tipologia, e l'età, della nostra vittima. Su esemplari relativamente recenti, e qui il mio esempio si ferma a un unico modello l'ICF110 di casa Icom, anche se il più recente ICF5022 ha caratteristiche analoghe. Questa radio prevede un comando seriale fornito quindi sulla presa del microfono, dove si collega solitamente anche il cavo di programmazione, che invia alla radio delle stringhe esadecimali che "dicono" al processore "vai sul canale XX". Il che ci porta a un nuovo problema... come faccio a sapere che stringa devo inviare alla radio? Facile, o quasi, è necessario uno sniffer, ovvero un



Collegamenti frontale icf1610 per telecomando

Collegamenti frontale ICF1010

cavo seriale costruito solo per questo uso che dirotta il segnale che il PC invia alla radio a un altro PC su cui leggeremo quanto il primo sta mandando alla radio. Resta sottinteso che è indispensabile avere due PC dotati di RS232..

La procedura di allineamento infatti prevede il collegamento della radio al PC, a mezzo dell'interfaccia di programmazione, e questo è utilizzato per cambiare la memoria in uso sulla radio secondo le esigenze della medesima procedura di allineamento. La procedura completa è riportata alla sezione 5 del service manual.

Dopo aver realizzato lo sniffer la sequenza letta è questa:

FE EE EF E8 26 22 01 06 00 30
31 FD versione 6

FE EE EF E8 26 22 01 05 00 30
31 FD versione 5

Come è evidente i primi otto byte sono uguali, il successivo dipende dalla versione del firmware della nostra radio, 1.5 o 1.6, mentre il penultimo determina la posizione in memoria, 31 per il Ch 1, 32 per il Ch2, 33 per il Ch 3 ecc..

La stringa che Arduino, via seriale (la comunicazione avviene a 9600 baud) dovrà inviare alla radio affinché venga selezionata la memoria relativa al Ch 1 è questa:

{0xFE, 0xFE, 0xEE, 0xEF, 0xE8,
0x26, 0x22, 0x01, 0x06, 0x00,

0x30, 0x31, 0xFD}; //versione 6
{0xFE, 0xFE, 0xEE, 0xEF, 0xE8,
0x26, 0x22, 0x01, 0x05, 0x00,
0x30, 0x31, 0xFD}; //versione 5

Dato che non sapremo a priori la versione del firmware della nostra radio invieremo alla radio entrambe le stringhe, tanto la logica della radio ignorerà quella che non è corretta per la sua versione.

Come dicevo questo è un caso molto particolare, probabilmente altri modelli del medesimo costruttore adottano lo stesso sistema, certamente non ne hanno la possibilità i modelli più datati, e quindi reperibili più facilmente e a condizioni economiche più vantaggiose.

Uno di questi è sicuramente ICF1010 e il suo quasi gemello ICF1610, per le versioni in VHF. Queste radio hanno, sempre in fase di programmazione, la possibilità di associare alcune memorie ai tasti posti sotto il display, quattro o cinque secondo il modello. Dunque il nostro sistema può sfruttare la cosa facendo credere alla radio che uno di questi tasti sia stato premuto. Il sistema è meno versatile del precedente, essendo limitato ai quattro (o cinque) tasti a nostra disposizione ma per l'uso in auto la cosa non dovrebbe rappresentare un problema.

Un'altra possibilità di comando è la possibilità della radio di utilizzare uno dei pulsanti pro-

grammabili per portare la radio su una memoria preimpostata. Mi spiego, un pulsante porta la radio sulla frequenza memorizzata sul Ch1, ora posso "salire" tra le memorie agendo sul comando UP, con una pressione passerò sul Ch2, con un'altra pressione passerò sul Ch3, ecc.

In questo caso è possibile istruire Arduino in modo che alla pressione del tasto 4 sul telecomando venga "premuta" il tasto che porta la radio sul Ch1 seguita da tre "pressioni" sul tasto UP, che ovviamente provvederà a portare la radio sulla quarta memoria. Questo è il caso dell'Alan HM06, radio decisamente economica anche da nuova, dalle dimensioni non particolarmente ridotte, cosa che aiuta l'intervento.

Vediamo qui due brevi filmati circa il funzionamento del telecomando IR su ICF 110 e su Alan HM06:

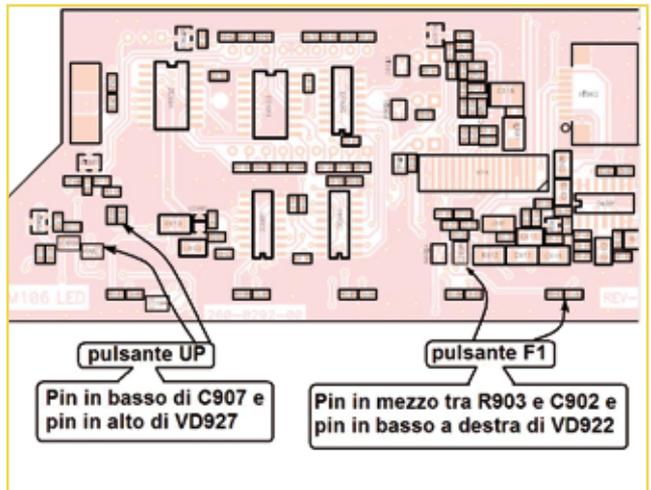
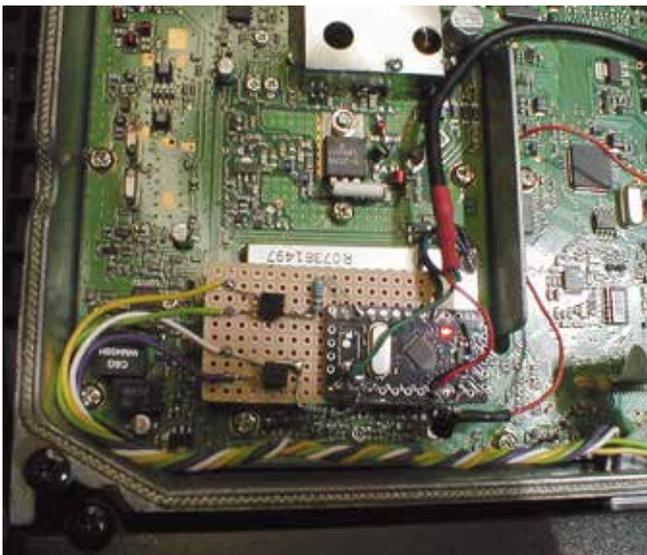
www.iw1axr.eu/scarico/alan2.mp4

www.iw1axr.eu/scarico/icom.mp4

qui alcuni esempi dei file utilizzati per programmare Arduino:

www.iw1axr.eu/scarico/esempi%20TLC.rar

Purtroppo il comando che proviene da Arduino non sempre è possibile utilizzarlo direttamente. Per Icom i pulsanti posti sul frontale portano a massa il comando corrispondente, dunque con il tasto a riposo sul pin sarà presente un uno logico, mentre a tasto



Mod TLC IR frontale Alan HM06

Foto del telecomando montato nell' HM06

premuto questo passerà a zero logico. Per Alan la cosa è un poco più complicata e sarà prudente utilizzare un fotoaccoppiatore il cui transistor di uscita simulerà la pressione del tasto corrispondente. In verità la disponibilità di relè reed particolarmente minuscoli potrebbe portare a utilizzarli levandoci ogni dubbio in merito al sistema di comando. Sistema decisamente poco elegante, ma indubbiamente più sicuro, se non conosciamo il metodo adottato dal costruttore. Resta inteso che la ricerca del manuale di servizio della nostra radio porta a una minore incertezza in merito, intervenire senza avere la docu-

mentazione del caso rimane una pessima idea. Come sempre www.radiomanual.info è fonte sicura di documentazione di questo tipo.

Telecomando su Alan HM06

La modifica consiste nell'applicare un telecomando a infrarossi atto al cambio del canale.

La sequenza che il telecomando invia alla radio è così strutturata:

- Da programmazione sulla radio è memorizzato in canale 1 come canale prioritario, richiamabile come home chan-

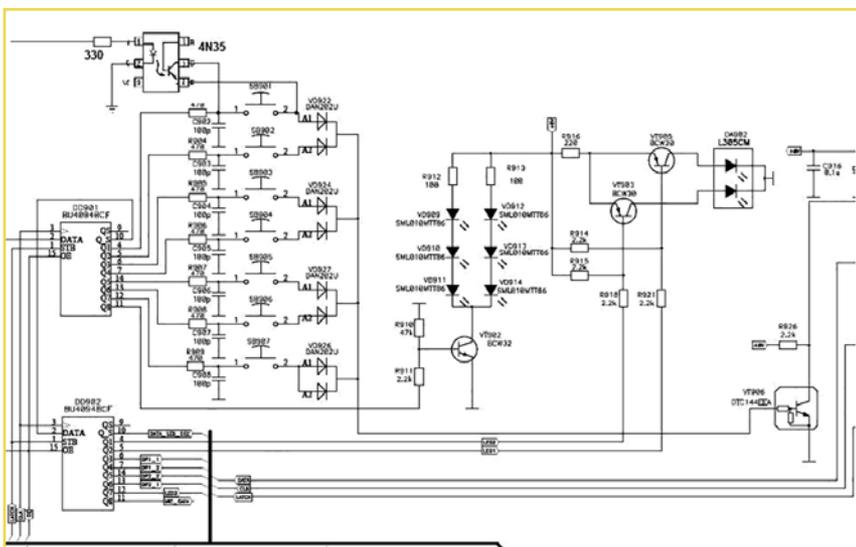
nel tramite la pressione di F1. In queste condizioni premendo F1 la radio va sul canale 1 e a ogni azione sul pulsante UP la radio incrementa la sintonia di un canale tra quelli memorizzati in sede di programmazione.

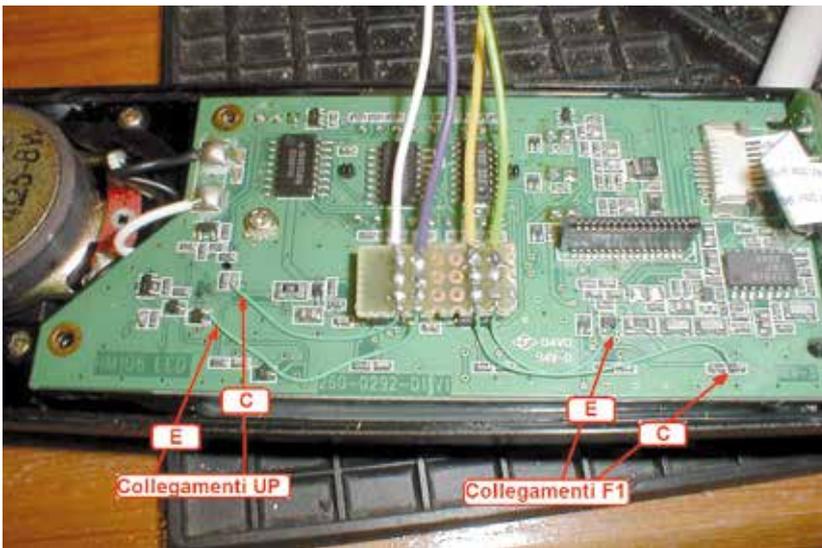
- Quando viene trasmesso un comando dal telecomando IR (da 1 a 9) il sistema "preme" F1 a cui segue alcuni azionamenti del pulsante UP, nello specifico:
 - Ch 1 un comando su F1
 - Ch2 un comando su F1 e uno su UP
 - Ch3 un comando su F1 e due su UP
 - Ch4 un comando su F1 e tre su UP
 - Ch5 un comando su F1 e quattro su UP
 -
 - Ch 9 un comando su F1 e otto su UP

Sulla radio vengono prelevati dal frontale i quattro collegamenti dei due pulsanti:

- Pulsante F1 (SB901) dal punto di unione tra R903 e C902 (al collettore) e dal A1 del doppio diodo VD922 (all'emettitore)
- Pulsante UP (SB906) dal punto di unione tra R908 e C907 (al collettore) e dal A2 del doppio diodo VD927 (all'emettitore). Nella figura e nelle foto sono evidenziati i punti di collegamento.

Schema Frontale HM06





Modifica al frontale Alan HM06.

Il comando dei due pulsanti avviene a mezzo di due fotoaccoppiatori NPN, 4N35 o simili, i collettori sono collegati ai gruppi resistenza/condensatore, gli emettitori sono collegati agli anodi dei diodi. Il LED del fotoaccoppiatore è acceso dal modulo Arduino tramite una resistenza da 330 – 560 ohm

Ad ogni azione sui pulsanti la radio questa emette in segnale acustico, un "beep", a verifica che l'azione richiesta è stata realmente eseguita.

Modifica IC-F110 e IC-F1010 per impiego del microfono dotato di tastiera DTMF

Programmare la radio con i canali necessari e la potenza prevista.

Interventi sulla main della radio

Provvedere a una sistemazione stabile dei due moduli e dell'eventuale cicalino. Collegare a massa il pin di massa di Arduino.

Effettuare le connessioni tra Arduino e il modulo DTMF, avven-

do cura di NON occupare nello specifico:

- alimentazione, +5V e massa
- le uscite dal modulo DTMF, D1, D2, D3, D4 rispettivamente ai pin 8,7,6 e 5 di Arduino
- l'uscita di decodifica corretta del tono (STD), in uscita dal modulo DTMF al pin 4 di Arduino

L'uscita STD negata è collegata al pin 3 di Arduino, anche se NON è usata. Il pin GND collega le due schede.

Sulla scheda DTMF dissaldare la presa jack femmina, l'ingresso audio è il pin centrale dei tre della fila esterna ora rimasta libera. In un primo momento si era intervenuti anche sul livello audio di ingresso dissaldando e risaldando una resistenza, modifica poi non realmente necessaria.

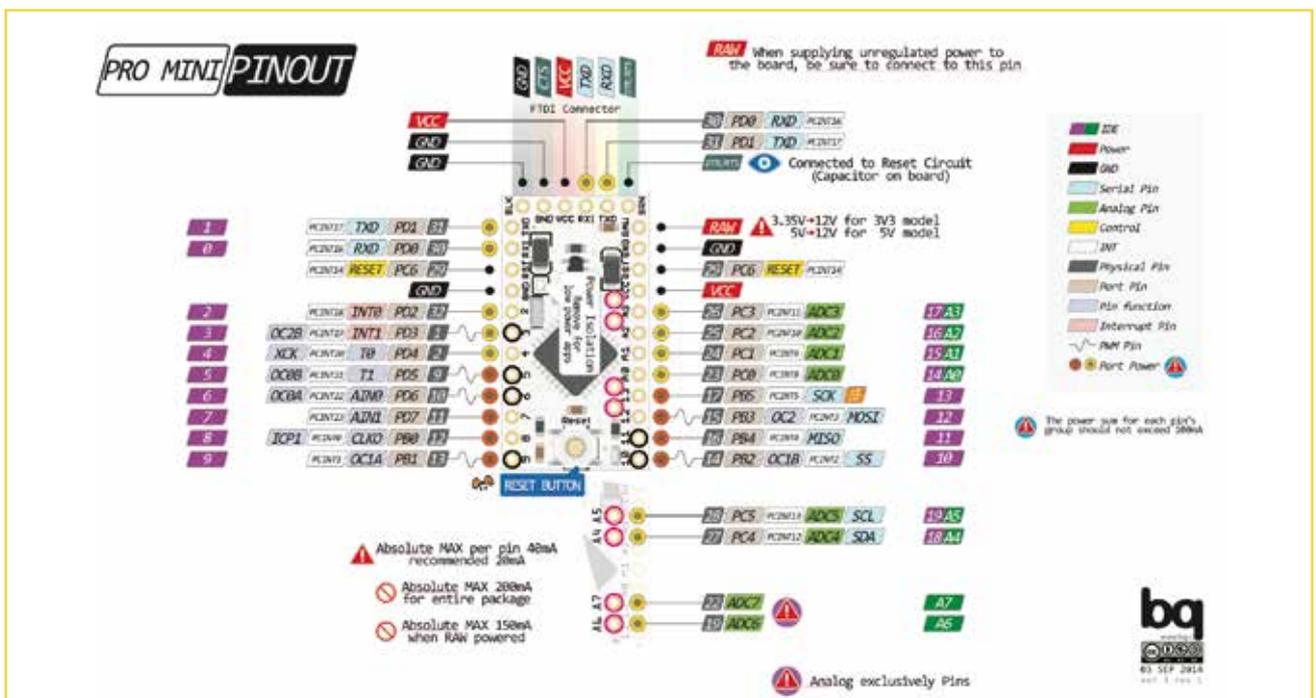
Sul pettine della scheda DTMF eliminare i pin di alimentazione, +5V e l'ingresso audio.

Sull'Arduino promini eliminare il pulsante di reset.

I due stampati vanno montati con i due lati saldature che si guardano, interponendo un cartoncino (non si sa mai...).

Collegare il pin di alimentazione Vcc tra Arduino e la scheda DTMF.

Prelevare i 5V di alimentazione dall'uscita del 7805 della radio.



Interventi sul frontale della radio, per ICF1010

Smontare il frontale e collegare un filo rispettivamente al pulsante del frontale:

P0 al punto di unione tra R32 e C25 (comando ch 1)

P1 al punto di unione tra R31 e C24 (comando ch 2)

P2 al punto di unione tra R30 e C23 (comando ch 3)

P3 al punto di unione tra R29 e C22 (comando ch 4)

+ 5V dall'uscita di IC4, è il lato dove troviamo C12, C13 e C34

Ingresso audio del modulo DTMF al pin 6 della presa MIC, è il secondo pin della fila interna partendo dal centro della scheda.

Collegare l'ingresso 08 di Arduino al pin PTT della presa MIC (pin 4)

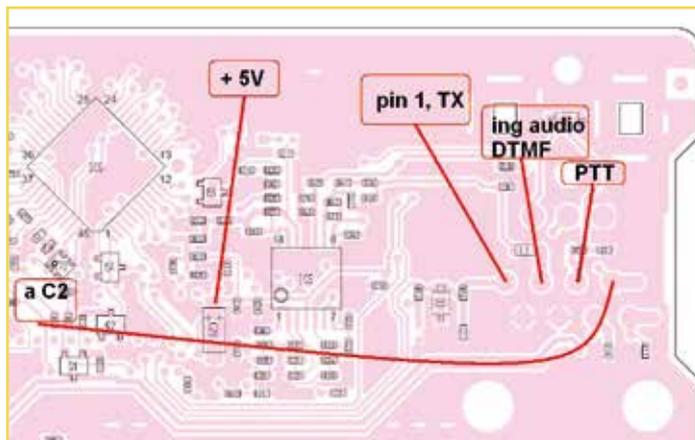
Interventi sul frontale della radio, per ICF110

Smontare il frontale e liberare il circuito stampato quindi eliminare R 15. Saldare un pezzo di filo al lato caldo di D2 e fare in modo che risistemando al suo posto lo stampato l'estremità libera sia utilizzabile nell'eventualità di dover riprogrammare la radio. In questa eventualità si deve collegare questa estremità, tramite una resistenza da 1 kΩ al pin 2 della presa micro. Questo collegamento libera un filo proveniente dal microfono e lo utilizza per accendere la radio. Alcuni microfoni cinesi hanno i pulsanti UP e DOWN sul microfono che possono essere utilizzati per accendere e spegnere la radio direttamente dal microfono.

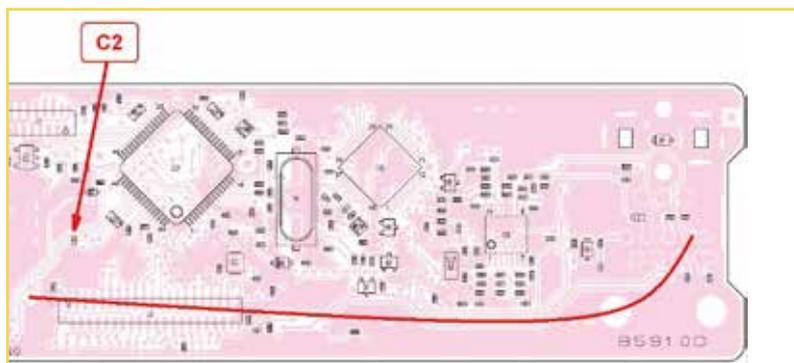
Prelevare i +5V dal lato basso di C29, verso l'integrato IC5, e portarli al pin di alimentazione di Arduino.

Unire con un filo il lato caldo di C2 (fa anche capo a una micro piazzola a lato del connettore per il flat cable) e il pin 2 della presa MIC (CLO).

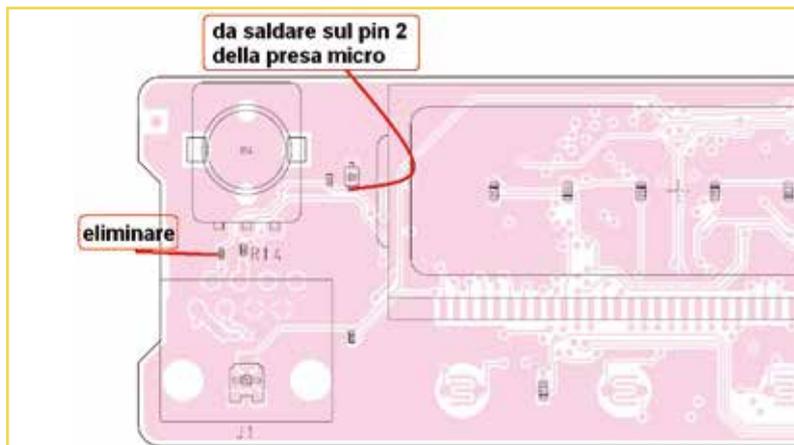
Portare l'uscita seriale di Arduino, (pin 1, TX) al pin 8 CLI della presa MIC, il primo verso D3, è quello della fila interna verso il centro del CS. Questo collegamento andrà rimosso in fase di riprogrammazione della radio



Modifica sul frontale 1 ICF110



Modifica sul frontale 2 ICF110



Modifica sul frontale 3 ICF110

Collegare l'ingresso audio DTMF all'uscita audio del microfono (pin 6 della presa MIC), è la piazzola accanto a quella CLI utilizzata prima, sempre della fila interna.

Collegare l'ingresso 08 di Arduino al pin PTT della presa MIC (pin 4), è il pin accanto a quello utilizzato al punto precedente. Questo collegamento impedisce alla radio di cambiar

canale quando si è in trasmissione.

In questo momento tutti i pin della fila interna del connettore MIC sono occupati dai collegamenti effettuati.

E' stata una lunga chiacchierata, descrivere il progetto completo per ogni radio diventerebbe ingombrante... Inoltre, al fine di evitare che colleghi possano causar danni... mi fermo qui! ■