

QRSS

Ovvero: “In CW chi va piano va molto lontano” (Mauro IK1WVQ)

Il QRSS è un modo LENTO, un CW con durata minima del punto maggiore di un secondo. la lentezza permette di coprire grandi distanze con potenze bassissime, e vediamo come e perchè.

Cominciamo con la definizione di QRSS.

Nel codice Q ordinario si usa il termine QRS per indicare al corrispondente di ridurre la velocità di trasmissione ("vai piano per favore").

QRSS è un'iperbole per indicare "vai MOLTO piano".

Ma perchè bisogna andare molto piano? E' una necessità se si vuole ricevere segnali debolissimi. Quindi non è un problema di conoscenza della telegrafia (qui i punti sono molto lunghi per cui questo CW non si ascolta, ma si VEDE riportato su una specie di waterfall). Quindi tranquilli!, non è necessario saper ricevere a orecchio (anche perchè qui a orecchio si sente solo rumore!).

Tutti gli OM dediti al CW sanno che riducendo la banda passante dell'RX (inserendo un filtro stretto) migliora in modo significativo il rapporto segnale rumore, e quindi si possono discernere segnali più deboli. Quindi più stringiamo il filtro e meglio è!

Si certo, però... c'è un però: qualsiasi tipo di modulazione provoca un aumento della banda occupata dalla trasmissione, e quindi a un certo punto occorrerà ridurre la larghezza del segnale, se no non starà più nella banda concessa dal filtro, con inevitabile perdita di parte dell'informazione trasmessa. (secondo teorema di Shannon. non vi tedio oltre perchè si finisce nella matematica spessa).

Provate e ricevere un segnale SSB con il filtro CW stretto e capirete cosa vuol dire "perdere parte dell'informazione"!

Comunque, siccome bisogna pur quantificare quanto appena detto, esiste questa equazione empirica che lega la velocità di una trasmissione CW (SOLO CW, OCCHIO!) alla larghezza del canale richiesto:

$$\text{canale minimo richiesto(Hz)} = 0.833 * \text{WPM.}$$

Quindi se vogliamo ricevere segnali molto bassi dobbiamo restringere il filtro, e quindi necessariamente ridurre la velocità di trasmissione.

Ecco spiegato il perchè della denominazione "QRSS".

Questa tabella riassume in pratica quanto appena detto:

WPM	Punti/min.	Sec./punto	BW_min Hz	SNR dB
12	600	0,1	10	0 (rif.)
8	400	0,15	6,67	-1,8
4	200	0,3	3,33	-4,8
1,2	60	1	1	-10
0,4	20	3	0,33	-14,8
0,12	6	10	0,1	-20

Per esempio, prendendo come riferimento una trasmissione a 12 WPM (punti lunghi 0.1 secondi) con potenza di 10W, si vede che a 0.4 WPM (punti lunghi 3 secondi) si ottiene un miglioramento di 14.8dB (= servono solo 330mW per ottenere lo stesso S/N), a condizione però di ridurre la banda passante del filtro a 0.33 Hz.

Quindi, più stringiamo il filtro più miglioriamo il S/N, purchè naturalmente si riduca la velocità di trasmissione, in modo da non sfiorare il canale disponibile.

Nel mondo QRSS si sono codificati dei sottomoduli che identificano la velocità di trasmissione, e sono nominati "QRSSxx" ove "xx" indica la durata in secondi di un punto CW. Ecco la tabellina riassuntiva:

WPM	MODO	BW_min Hz	SNR dB	W richiesti
12	CW	10 Hz	0 (rif.)	10W
1,2	QRSS 1	1 Hz	-10	1W
0,4	QRSS 3	0,33 Hz	-14,8	330mW
0,12	QRSS 10	0,1 Hz	-20	100mW
0,04	QRSS 30	33 mHz	-24,8	33mW
0,02	QRSS 60	17 mHz	-27,8	17mW
0,002	QRSS 600	1.7 mHz	-37,8	1.7mW
0,0002	QRSS 6000	170 uHz	-47,8	170uW

Normalmente in HF si usa il "QRSS3" (quello dell'esempio precedente) il limite è "QRSS6000", ove il punto è lungo 6000 secondi (100 minuti!, usato solo in ULF per i test, ma non facciamoci prendere la mano!)

La giustificazione matematica di tutto ciò la dobbiamo a Claude Shannon ed a Harry Nyquist.

Fin qui la teoria.

Come al solito tanta bella gente dice tante belle cose e dice di ottenere tanti bei risultati..

MA COME FACCIO IO A REALIZZARE UN FILTRO DI SOLI 0.33Hz DI BANDA ?

Farlo hardware neppure da parlarne (ringing, stabilità, ecc.). Oltretutto obbligheremmo il corrispondente a una stabilità di frequenza (pochi centesimi di Hz) improponibile per noi hobbisti (non parliamo neppure del QRSS6000, HI!)

Quindi occorre giocoforza prendere il segnale audio dall'RX, portarlo dentro al PC (come si fa per l'FT8 ecc.), effettuare il filtraggio voluto tramite filtri digitali (FIR, i più usati) e quindi visualizzare il segnale graficamente (tempo in ascissa, frequenza in ordinata, praticamente un waterfall girato di 90°). Così facendo sparisce il problema della deriva in frequenza del RX+TX, semplificando di molto i requisiti HW.

Bando alle ciancie e saldatore alla mano! IN PRATICA.

RICEZIONE

Il programma più semplice per la visualizzazione (non DECODIFICA!) del QRSS è "ARGO.EXE" dell'amico Alberto I2PHD (<https://www.i2phd.org/argo/index.html>) facile da configurare e usare.

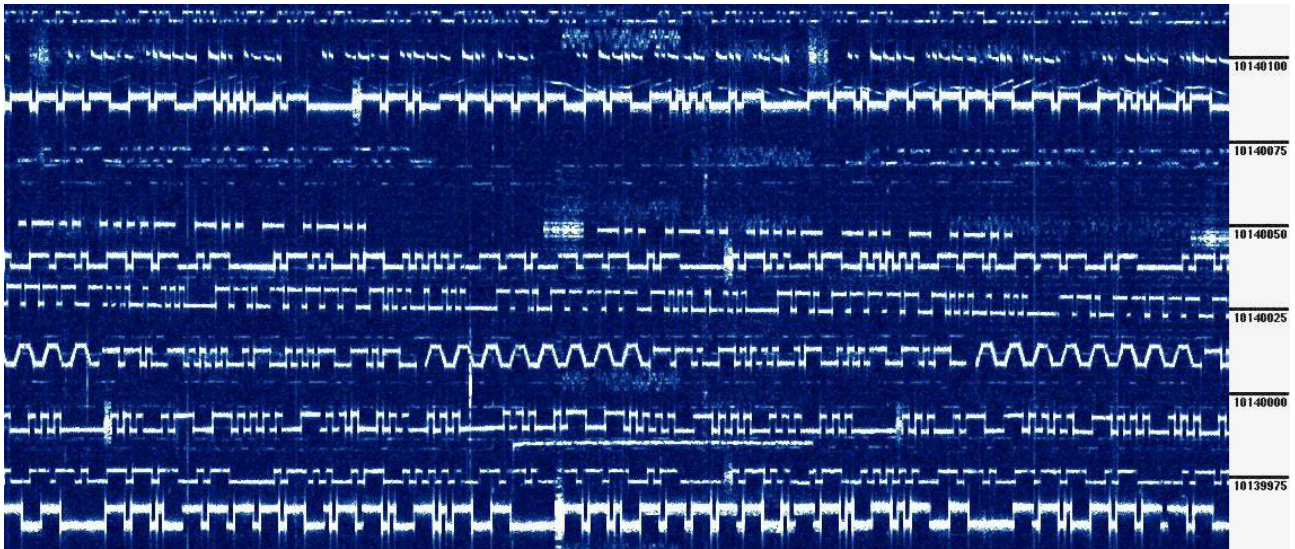
Quindi:

accendiamo il nostro RX, sintonizziamolo a 10139.5 Hz, lasciamo pure il filtro SSB, e ascoltiamo cosa riceviamo: ASSOLUTAMENTE NULLA, solo rumore.

Non scoraggiamoci e colleghiamo l'audio, tramite una solita interfaccina per modi digitali (o semplicemente con un pezzo di cavo coax tra uscita audio RX e ingresso "line" del PC).

Lanciamo Argo.exe, configuriamo alcuni parametri essenziali (scheda audio, modo QRSS3, velocità lenta) e lasciamolo andare per alcuni minuti.

cosa apparirà ?



Scusate, ma non c'era solo del soffio?

Sì, per le nostre orecchie umane sì, col filtro SSB questi segnalini sono completamente cancellati dal rumore, ma per le orecchie di un software, semplicemente dopo adeguato filtraggio stretto, questi sono ben visibili e interpretabili.

Magia dei filtri digitali? già!

Adesso cerchiamo di capire cosa sono questi strani segni che si vedono.

Per la maggior parte dei casi si tratta di FSK/CW o di DFCW (Dual Frequency CW).

Vediamo in dettaglio il "FSK/CW":



la parte superiore della "onda quadra" rappresenta la manipolazione vera e propria (punti e linee), mentre la parte inferiore è la "frequenza di parcheggio" dell'oscillatore durante le pause.

Il motivo di questa codifica deriva dal fatto che ai tempi (anni 80) gli oscillatori erano a quarzo, e se si manipolava in modo ON-OFF, la frequenza di oscillazione variava (di pochi Hz, ma qui anche pochi Hz possono disturbare la leggibilità), quindi si lasciava l'oscillatore sempre attivo, e si variava la frequenza tramite un comune diodo che svolgeva funzioni di VARICAP.

questo è un esempio di brutta manipolazione ON-OFF, da evitare:



Oggi i generatori sono digitali, senza questo problema, ma l'abitudine è rimasta.

Altra codifica, meno comune, è la DFCW (Dual Frequency CW): LINEE in alto e PUNTI in basso. Ha il vantaggio che punti e linee hanno la stessa durata, ottima cosa per una trasmissione lenta.



Potendo modulare a piacimento la frequenza, si sono anche implementate modalità curiose, tipo

quella di "disegnare" il nominativo (che è poi la codifica HELLSCRIBER), oppure creare dei loghi grafici, il limite è la fantasia. Negli anni abbiamo visto veramente di tutto!

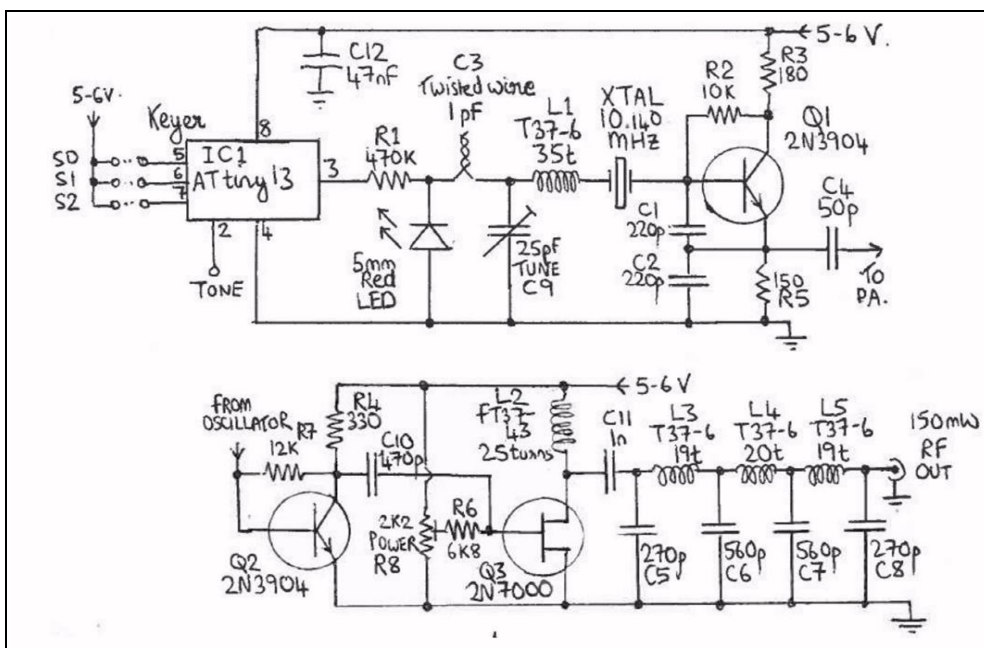


Esistono in rete dei ricevitori SDR orientati specificamente alla ricezione del modo QRSS, sono detti "GRABBER", e qui trovate un completo elenco: <https://swharden.com/qrss/plus/>

E ADESSO TRASMETTIAMO!

Questo è il regno dell'autocostruzione, le potenze richieste sono bassissime (poche decine di mW) per ottenere risultati di tutto rispetto. Non superate per nessuna ragione il Watt, non ce n'è alcun bisogno.

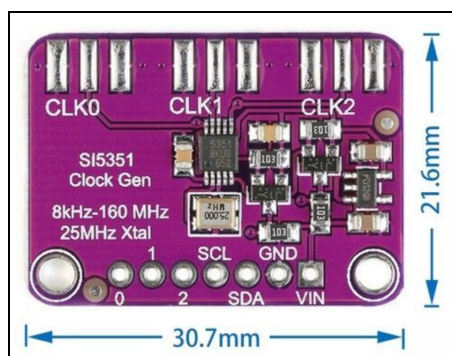
Ai tempi eroici i trasmettitori erano di questo tipo:



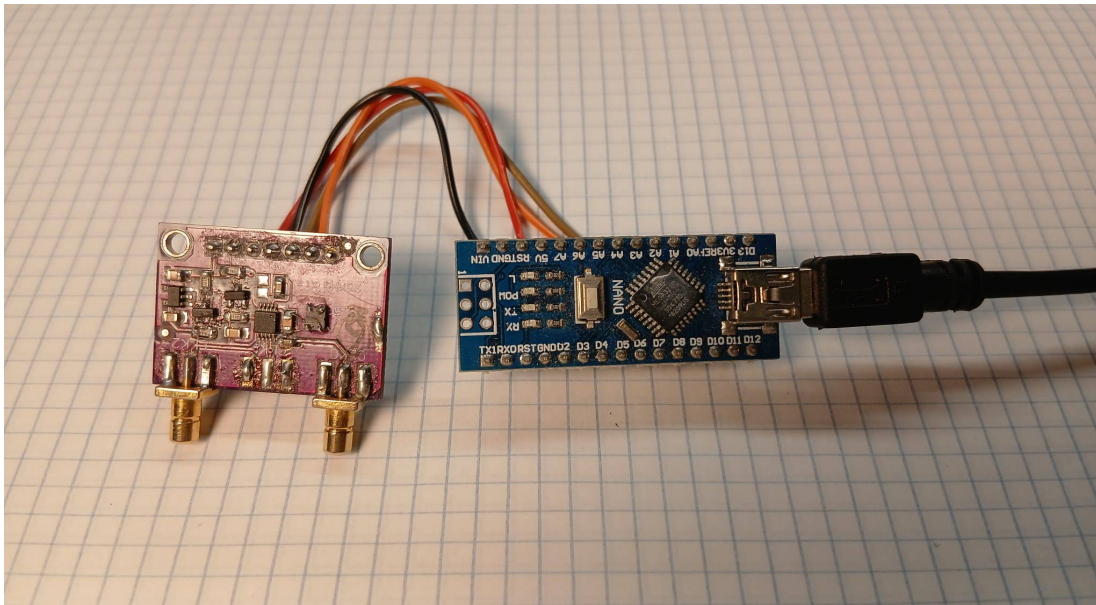
il LED faceva da varicap per variare la frequenza (occhio! LED ROSSO, non verde!).

Spesso il microprocessore era addirittura rimpiazzato da un NE555 che generava un'onda quadra (magari asimmetrica) per potersi riconoscere nei Grabber in rete!

Purtroppo oggi i quarzi non si trovano più, e quindi ci si deve appoggiare a generatori di clock digitali. Per esempio l'arcinoto Si5351 (ne ho parlato a lungo nei numeri scorsi della rivista) ha semplificato di molto le cose, ma richiede però un microprocessore, tipicamente un Arduino, per programmarlo. E' in vendita per pochissimi euro uno schedino "viola" totalmente assemblato, stessa cosa vale per il microcontrollore "ARDUINO NANO" (o qualsiasi altro della famiglia).



Scheda Si5351



Il risultato discusso in questo articolo è stato ottenuto con questo TX

Per chi ha un minimo di conoscenze nel mondo Arduino, questo è un Firmware minimale ma perfettamente funzionante per ottenere una frequenza voluta.

```

/*
QUARZO "DIGITALE" con scheda Si5351 e ARDUINO  Mauro IK1WVQ
Per Arduino Nano: A4 -> SDA  A5 -> SCL
RF OUT sulla porta CLK0 della scheda Si5351
su IDE Arduino installare la libreria "Etherkit Si5351"
*/

#include "si5351.h"

#define F_Hz      10136000      // Frequenza voluta in Hz
#define F_Rif_Hz 26000000      // Frequenza quarzo installato sullo schedino Si5351 in Hz

Si5351 si5351;    // attiva la libreria

void setup(void)
{
    si5351.init(SI5351_CRYSTAL_LOAD_10PF, F_Rif_Hz, 0);      // init del Si5351
    si5351.drive_strength(SI5351_CLK0, SI5351_DRIVE_8MA);  // out circa 10mW (2mA/4mA/6mA/8mA)
    si5351.output_enable(SI5351_CLK0, 1);                  // abilita l'uscita sulle due porte 0 e 1
    si5351.set_freq(F_Hz*100, SI5351_CLK0);                 // setta la frequenza voluta
}

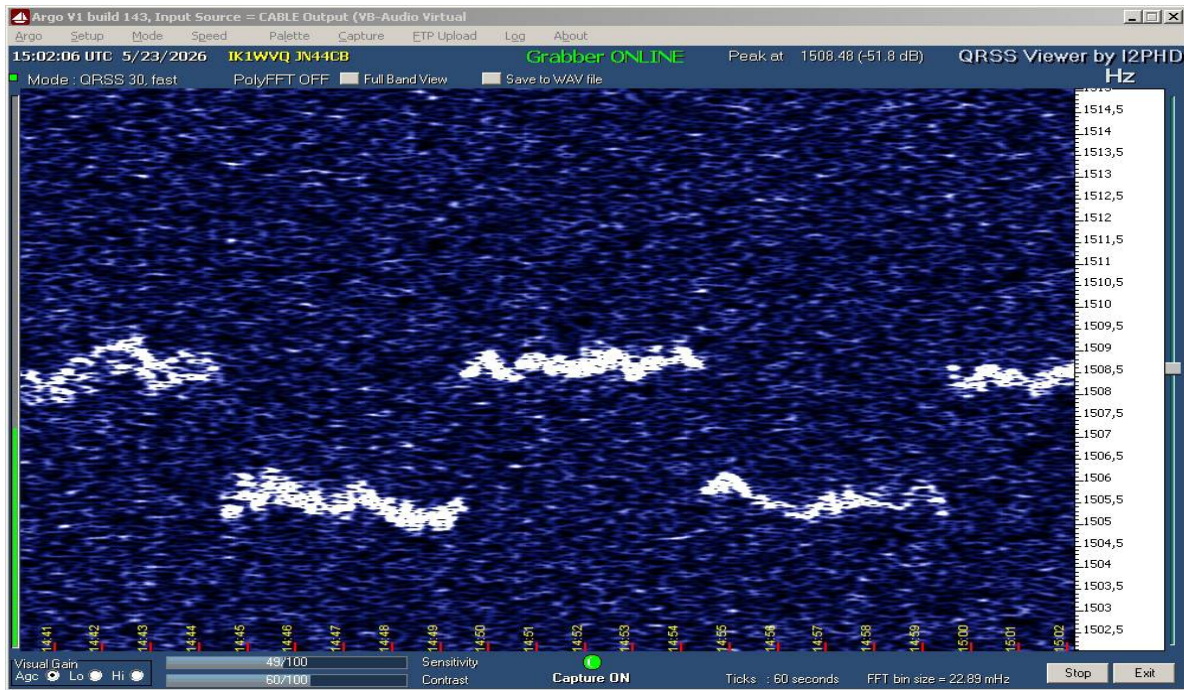
void loop(void) {}

```

FACCIAMO UN ESPERIMENTO, per renderci conto delle potenzialità di questo sistema:

- > riducete quanto più possibile la potenza del vostro TX (max 500mW)
- > trasmettete una portante continua in 30metri (o altra banda a vostra scelta)
- > tramite il WEB-SDR di Twente (Olanda), o altro, ricevete il vostro segnale
- > trasferite l'audio ad Argo tramite un “cavo audio virtuale”

Questo è quanto ricevuto dal RX di Twente, ottenuto trasmettendo un'onda quadra con periodo di 10 minuti, potenza di **590 uW** (MILIONESIMI di W) antenna dipolo:
(Le ondulazioni non sono colpa mia ma sono dovute al WEB-SDR)



Notate che sono visualizzati solo 13Hz in totale e che l'onda quadra ha espansione di soli 3Hz!
Come si può vedere i risultati fanno venir voglia di provare!!

Per generare le onde quadre o altro, nel FW riportato sopra sostituite la riga:

```
void loop(void) {}
```

con queste:

```
void loop(void)
{
    si5351.set_freq(F_Hz*100, SI5351_CLK0); // setta frequenza "bassa"
    delay(300000); // attesa 5 minuti = 300000 msec

    si5351.set_freq((F_Hz+5)*100, SI5351_CLK0); // setta frequenza "alta" (+5Hz)
    delay(300000); // attesa 5 minuti = 300000 msec
}
```

Comunque, per chi non vuole addentrarsi nell'autocostruzione, segnalo che Hans GOUPL di QRP-LABS propone un ottimo kit: <https://qrp-labs.com/ultimate3.html>

Però, volete mettere il fascino del saldatore ??? HI!

SI PUO' FARE QSO IN QRSS ?

Certo che sì, con TANTA TANTA pazienza, compattando al massimo i dati scambiati, un pò come si fa con il Meteor Scatter o con l'FT8.

Essendo il QSO necessariamente programmato, consiglio di accordarsi per sostituire i nominativi con una, al massimo due lettere e codificare il rapporto con una lettera (non numero, troppo lungo)..

Queste sono le frequenze di trasmissione preferite per il traffico QRSS (centro banda, +/- 100Hz)

600m: 476.100 kHz	160m: 1837.900 kHz	80m: 3569.900 kHz
60m: 5288.550 kHz	40m: 7039.900 kHz	30m: 10140.000 kHz
22m: 13555.400 kHz	20m: 14096.900 kHz	17m: 18105.900 kHz
15m: 21095.900 kHz	12m: 24925.900 kHz	10m: 28125.500 kHz

Link utili:

SDR-WEB <http://websdr.ewi.utwente.nl:8901/>

“Virtual Audio Cable”: <https://vb-audio.com/Cable/>

Mail list dei cultori del QRSS: <https://groups.io/g/qrssknights>

software consigliati per la ricezione QRSS:

ARGO <https://www.i2phd.org/argo/index.html> il software entry level. Usatissimo

LOPORA di Onno PA2OHH <https://github.com/swharden/Lopora> moderno, scritto in Python , bello ma in po' macchinoso da installare.

SPECTRUM_LAB <https://www.qsl.net/dl4yh/spectral1.html> Veramente impressionante, potentissimo ma MOLTO difficile da usare. Consigliato a utenti esperti

Chi vuole provare può contattarmi a questo indirizzo email per delucidazioni:

ik1wvq@stmb.it

Termino con questo (per chi non riesce proprio a fare a meno dei logaritmi! HI!)

Teorema di Shannon

Teorema di Shannon: la massima velocità di trasmissione o **Capacità**, in un canale rumoroso la cui larghezza di banda è H Hz con rapporto S/R (segnale/rumore), è data dall'equazione:

$$C = \text{massimo numero di bps} = H \log_2 (1 + S/R).$$

Poichè la larghezza di banda H della linea telefonica è di 3 KHz e il rumore presente è di 30 dB (= 1000), il limite è

$$C=3100 \cdot \text{Log}_2 (1001) = 3100 \times 9,96 = 30876 \text{ bit/sec}$$

mentre per un canale televisivo si parte da 6MHz

Rinnovo Quota Sociale annuale

Come ogni anno siamo a ricordarvi di rinnovare l'iscrizione alla nostra Associazione.

La quota associativa di base è la stessa dell'anno scorso, ovvero € 15, compresa l'assicurazione antenne, da versare tramite bonifico bancario al IBAN: IT41N0623051010000015401758 (Credit Agricole) intestato a: Amateur Radio Society

Occorre poi inviare la ricevuta scannerizzata del bonifico a: segreteria@arsitalia.it

Vi preghiamo di effettuare i versamenti entro il mese di gennaio.

Non vi sono more, ma a far data 28 febbraio dell'anno successivo l'A.R.S. Italia considererà receduti coloro che non hanno inteso rinnovare la quota sociale.

Ricordiamo che solo chi è in regola con il versamento può accedere ai servizi offerti.

l'A.R.S. propone anche alcuni servizi in opzione:

- Servizio “QSL Bureau”: € 40.

- Polizza per la “tutela legale” in caso di controversie.

Contattare la segreteria per maggiori informazioni.