

## Kontrola audio przy pomocy QSORecorder

SVXLink posiada możliwość nagrywania prowadzonych łączności przy pomocy QSORecorder'a. Możemy wykorzystać QSORecorder do sprawdzania poprawnego poziomu ustawień w svxlink. QSORecorder nagrywa audio takie jakie jest transmitowane poprzez sieć FM a nie przez radio co pozwala poprawnie zweryfikować poziom audio transmitowanego do sieci. Ważne aby transmitowane audio nie przekraczało poziomu -6dB. Po przekroczeniu tego poziomu svxlink który będzie transmitował takie audio będzie podawała kompresji.

Aby używać QSORecorder należy w svxlink.conf w sekcji [SimplexLogic] lub [RepeaterLogic] mieć ustawione:

**QSO\_RECORDER=8:QsoRecorder**

**co pozwala aktywować nagrywanie poprzez kod DTMF 81# i wyłączyć nagrywanie kodem DTMF 80#**

Ponadto musimy mieć w svxlink.conf przykładową sekcje:

```
[QsoRecorder]
REC_DIR=/tmp
MAX_TIME=240
SOFT_TIME=300
MAX_DIRSIZE=100
DEFAULT_ACTIVE=0
TIMEOUT=120
#QSO_TIMEOUT=120
ENCODER_CMD=
```

W katalogu podanym w **REC\_DIR** są zapisywane pliku WAV z nagraniem QSO. Nazwy plików przykładowe są w postaci:

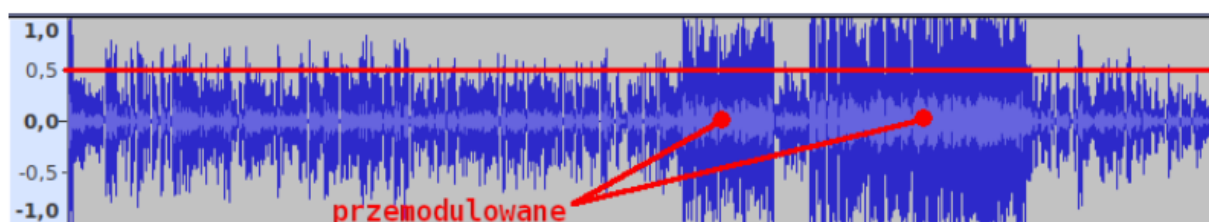
**qsorec\_SimplexLogic\_2023-10-05\_195623\_2023-10-05\_195845.wav**

gdzie w nazwie jest data i czas nagrywania.

Pliki możemy pobierać kiedy będzie zakończone nagrywanie (kod DTMF 80#)

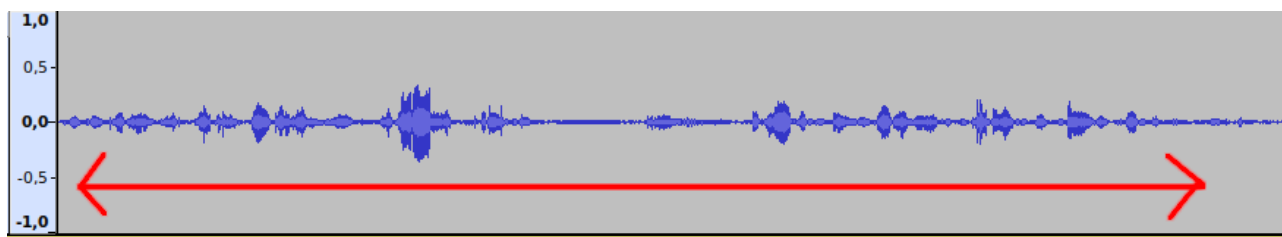
Plik należy pobrać z komputera gdzie jest svxlink np. używając programu WinSCP (<https://winscp.net/eng/download.php>) na lokalny komputer i otworzyć go w programie darmowym AUDACITY <https://www.audacityteam.org/>

Przykładowy analiza wgranego pliku z AUDACITY:



Audio które przekracza na skali +/- 0.5 (czyli 6dB) jest przemodulowane i **będzie podlegać kompresji** (LIMITER\_THRESH=-6 w [Tx1] i [Rx1], uwaga bez tej opcji audio będzie przesterowane) podczas transmisji przez nadajniki svxlinka a nie chcemy tego i **będzie zbyt głośne do optymalnego poziomu**. Należy ustawić poziom audio tak aby mieścił się w skali +/- 0.5 jak pokazane na powyższym rysunku.

Jeśli audio jest za ciche będzie wyglądało tak:



Polecam stosować to formę kontroli poziomów wysyłanego audio do sieci FM pozwoli ona że podłączone nody będą miały podobny poziom audio i da to komfort jakościowy odbieranego audio wszystkim użytkownikom sieci FM

Musimy mieć na uwadze że nagrywanie audio z radia które już odebrało audio z svxlink nie jest odpowiednią drogą do kontroli poziomów audio osoby która transmituje przez sieć FM ponieważ podlega obróbce poziomów ustawień lokalnych karty dźwiękowej nadajnika. Jedynie QSORecoder pozwala zarejestrować audio jakie jest dostarczone do sieci FM.

Korzystanie z QSORecodera i AUDACITY poleca sam autor svxlink SM0SVX

<https://groups.io/g/svxlink/message/937>

Aby nagrać własne audio należy aktywować QSORecorder kodem DTMF **81#** i następnie wybrać grupę TG 9990 (ECHO w sieci FM POLAND) kodem DTMF **919990#** i następnie zacząć mówić do radia przez hotspot po odsłuchaniu audio z ECHO zakończyć nagrywanie kodem DTMF **80#**. Pobrać plik WAV z nagraniem i ocenić go w AUDACITY. Staraj się aby czas nagrania był w miarę reprezentatywny do oceny tzn przynajmniej **15 do 20 sekund**.

### **Opis wszystkich parametrów QSORecorder**

#### ***QSO Recorder Section***

*The QSO recorder is used to record all received audio to files on disk. All audio from receivers, modules and logic links are recorded. Announcements are not recorded.*

#### **REC\_DIR**

Use this configuration variable to specify in which directory to write the audio files. A good place is /var/spool/svmlink/qso\_recorder.

#### **MIN\_TIME**

If the duration of the recorded content for a file is less then MIN\_TIME milliseconds, the file will be deleted when the file is closed. Default: 0 (empty files will be deleted).

#### **MAX\_TIME**

Setting this configuration variable will set an upper limit for the file size of a recording. No more than MAX\_TIME seconds of content will be recorded to a single file. When the maximum time have been reached, the file is closed and another file is created. Note that it is not the maximum time that the recording has been active that we are setting a limit for but rather how much content that have been recorded to the file. If nothing is recorded, the file can stay open indefinitely.

Default: 0 (no limit)

#### **SOFT\_TIME**

To not get abrupt breaks in recordings it is possible to set a soft break time. Let's say that MAX\_TIME is set to 3600 seconds (one hour). If we set SOFT\_TIME to 300 seconds (five minutes) the QSO recorder try to close the file on a squelch close somewhere between 55 and 60 minutes. In this way we may avoid getting transmissions split up between files. Default: 0 (no limit)

#### **MAX\_DIRSIZE**

Specify the maximum total size in megabytes of the files in the recording directory. If the limit is exceeded, the oldest files are deleted. The directory size is checked upon file close so the size may

grow temporarily past the limit with at most the size of one recorded file. Only files which have a filename starting with "qsorec\_" will be considered for deletion. If using an ENCODING\_CMD, make sure that the "qsorec\_" prefix is not removed from the target filename unless you really want the MAX\_DIRSIZE feature to skip them. Default: 0 (no limit)

### **DEFAULT\_ACTIVE**

If this configuration variable is set to 1, the QSO recorder will be activated by default when SvXLink start. Default: 0 (default inactive)

### **TIMEOUT**

If a timeout is specified, the activation state of the QSO recorder will return to the value specified in the DEFAULT\_ACTIVE configuration variable when the node has been idle for the specified number of seconds. When DEFAULT\_ACTIVE is unset or 0, if the QSO recorder is manually activated it will be automatically deactivated after the specified amount of time of inactivity. When DEFAULT\_ACTIVE is set to 1, if the QSO recorder is manually deactivated it will be automatically activated after the specified amount of time of inactivity. Default: 0 (no timeout)

### **QSO\_TIMEOUT**

Set this configuration variable if you want to close the currently opened file and open a new one after each QSO. The number of seconds the node should be idle before closing the file should be specified. Default: 0 (no QSO timeout)

### **ENCODER\_CMD**

Specify a command to be executed after a new wav file have been written to disk. This makes it possible to use an external encoder utility to encode the wav file to another format. Even though this configuration variable was added to run an external encoder it could do more complicated things with the file if needed. A couple of examples would be to transfer the file to another computer or to send a notification e-mail. If the command line get too complicated it may be a good idea to write a script instead.

The encoder command will be run under a shell so normal shell operators like redirects and pipes may be used. The shell specified in the SHELL environment variable will be used and if not set, /bin/sh will be used. The "-c" command line option will be added so the complete command will look something like: \$SHELL -c "\$ENCODER\_CMD". A number of %-codes can be included in the command. They have the following meaning:

- %f - The full filename with full path
- %d - The directory part (what REC\_DIR is set to)
- %b - The basename, that is, the filename without path and extension
- %n - The filename without path but with extension

## FM POLAND komunikacja radioamatorska w sieci FM

---

The encoder will be started in the background and it will not be stopped even if SvxFLink exits. It will run in the background until it's done. As long as SvxFLink is running it is monitoring the encoding processes. If a process run for longer than one hour it will be killed.

Note that SvxFLink will never remove the original recording so that have to be done in the encoder command. Here are a couple of examples:

```
ENCODER_CMD=/usr/bin/oggenc -Q \"%f\" && rm \"%f\"
```

```
ENCODER_CMD=/usr/bin/lame --quiet \"%f\" \"%d/%b.mp3\" && rm \"%f\"
```

```
ENCODER_CMD=/usr/bin/speexenc \"%f\" \"%d/%b.spx\" 2>/dev/null && rm \"%f\"
```

```
ENCODER_CMD=/usr/bin/opusenc \"%f\" \"%d/%b.opus\" 2>/dev/null && rm \"%f\"
```