FT8 and diversity

Iztok, S52D

January 28, 2019

Abstract

An implementation of FT8 diversity for WSJTX program is described. Time diversity is summing up two consecutive even or odd samples, space diversity is summing up samples form two synchronous receivers and two copies of WSJTX, while frame diversity is summing up decoded messages from two WSJTX instances.

Diversity, as implemented, helps decode noticeable more traffic and provides significant improvement.

1 Introduction

Diversity reception is common in all modern radio communications. Even cheapest cellular phones have two antennas and receivers. FT8 mode of WSJTX is extremely popular on HAM shortwave bands and we need better reception techniques to deal with interference and other obstacles contributing to weak signal reception.

Better FT8 decoding is achieved using diversity, by combining two or more RX signal sources. 2 to 5 dB S/N can be gained, enough to make some QSOs possible.

FT8 decoder analyzes 12 seconds of captured audio and removes all decoded signals in several passes. We try to extract more signals by summing up two audio samples. Three kinds of diversity are implemented in S52D FT8 decoding. Time diversity, also called incremental redundancy, combines two consecutive even/odd samples. Space diversity combines two samples when two phase synchronous receivers are used. Frame diversity is also implemented so that already decoded signals are not decoded again.

Type of diversity is presented to the user by adding letter f, s or t in the result annotation field.

Since time and space diversity both adds seven passes to FT8 decoder leading to longer decoding time, a fast CPU is needed. Code was tested on Linus with 3.8 GHz i7 CPU and works fine.

Diversity patch was implemented in WSJTX 1.9.1 and later in WSJTX 2.0.0. Results are excellent: S52D made several QSOs that would not be possible with standard FT8 decoder. Benefits are seen on empty band with only few decodes, as well as on crowded 20m band. Usually 5 % more messages are decoded using either time or space diversity.

2 Time diversity

No additional hardware is needed. Decoder takes two consecutive odd/even samples with residue signals and combines them with proper tone phase. For repeated messages it combines amplitude of the signal, while noise and interference is only power combined.

To put it simply, it can be said that some bits are coming from the current sample, and some from the previous sample.

Sometimes it decodes messages that were transmitted in the previous sample, occasionally both messages are decoded with current time. For example, we can see both RRR and 73 messages together. This happens when summing up helps with common bits like CALLs.

"t' marked messages are not to be trusted completely, while operating FT8, as they might be ghosts from previous sample.

Samples are being summed up, while changing bands or just receiver QRG. Thus, old band data can be decoded as being received on the new band and reported to PSKreporter.

3 Space diversity

For space diversity (called stereo diversity by W8JI) two copies of WSJTX shall be run, each monitoring same frequencies on different receiver attached to different antenna. Both radio receivers have to use exactly same oscillators.

Several SDR units (Afedri, Red Pitaya) supports synchronous reception, as well some mainstream RIGs like K3, IC-7610 and probably also FT-1000 MP and TS-990S.

Secondary WSJTX instance writes file to a specified directory, while primary reads it and adds it to decoding. Frame diversity is also used, adding messages decoded by secondary WSJTX and not by primary. File can be properly closed on time as enough delay is provided by time diversity on primary WSJTX.

Real space diversity requires two antennas separated by few wavelengths in order to fight fading. Polarization diversity with one vertical and one horizontal antenna gives good results as well.

While testing using IC-7610, benefits were visible by using any two different antennas. Having different vertical and horizontal pattern results in better reception.

Some testing is needed to get feeling of how space diversity behaves on different bands at different instances.

Files generated by space and frame diversity are named with time. So they must be removed every day to prevent false decoding.

4 Frame diversity

As implemented, it simply adds messages decoded by secondary WSJTX to the primary one.

As full decode is done by one decoder, strict RX synchronization is not required and any of the two radios can be used.

They have to be tuned to the same frequency, otherwise WSJTX might get confused and transmit on wrong frequency. If they are tuned to the same band with reasonable small offset, one TX frequency is fine to make QSOs on both RX windows. Also, when chasing DX, main WSJTX can be tuned to whole band, while secondary is using 200 Hz filers to pick out only the wanted station.

5 Synchronization

Main challenge in diversity implementation was how to synchronize audio samples for different signals. Two instances of WSJTX do not capture audio at exactly same time. TX stations also do not transmit exactly at the same time, the main reason being the latency of the operating system.

A sequence of 7 delays was found, that works fine for audio spectrum from 270 to 3000 Hz. It is based on a simple fact: if two signals are offset 50 degrees, then summing them up when one is delayed 410 degrees, or 770 degrees are same as delaying of 50 degrees.

Using just 7 delays, we cover whole audio bandwidth with any possible delay and when summing up, error from perfect match is maximum 60 degrees for 97 % of samples, maximum 30 degrees for 77 % of samples while 50 % samples are within 16 degrees.

Since FT8 symbol length of 1/6.3 seconds is long enough, small mistakes on symbol change do not degrade decoding noticeably.

6 Installation

Replace two files in src/wsjtx/lib directory and recompile wsjtx. Most users will try time diversity only and it works with no additional configuration files.

Those with two radios and antennas can benefit from frame diversity, while only few of us have possibility to use space diversity. Both demand two copies of WSJTX to run (use -r name), and each needs separate configuration file.

If primary WSJTX stops with fortran error while reading temporary file, a sollution is to disable apriori decoding (or use fast decoding) with secondary WSJTX.

7 Configuration file

If there is no configuration file, only time diversity is used. Common directory where the files are stored must be specified when two receivers are used.

Files wsdiv.txt shall be created in a writable direc-

tory where files ALL.TXT, wsjtx.log etc are stored for each WSJTX instance.

wsjtx.txt file has a simple format. The first 4 lines contain 0/1 to enable time diversity, data writing, data reading for frame diversity and data reading for space diversity. Two directory names shall then be added, one for file writing and one for reading.

Example for space diversity:

meaning	primary	secondary
time diversity	1	1
data writing	0	1
frame diversity	1	0
space diversity	1	0
write directory	.	/tmp/div/
read directory	/tmp/div/	.

8 Code implementation

Author learned FORTRAN 4 in high school back in 1974/1975, and switched to pascal in 1976. Learning a bit of modern fortran was a pleasant surprise, as well as trying to understand how ft8 decoder really works.

Only two files are changed: ft8_decode.f90 where real work is done and decoder.f90 where t/f/s annotations are generated.

There are no changes to the program structure or to procedure parameters so that porting to new versions remains simple.

9 Future work

More integration into WSJTX will move configuration file into standard WSJTX configuration menu. By propagating exact RX QRG to decoder we can stop time diversity during QRG change. WSJTX is well balanced and works fine on a different range of computers. Diversity patch is CPU hungry and works only on top range CPUs.

Faster version of diversity patch will be provided, using 4 steps for synchronization. Slightly lower number of messages can be decoded, compared to version with 7 steps.

Space diversity can be made simpler, if WSJTX is capturing audio samples at exactly same time, so there is no need for synchronization. Modifying WSJTX to work with two radios at the same time is not likely due to complexity of work to cover all possible combinations.

External SDR programs can be modified to feed multiple WSJTX instances with synchronous data.

Unlike JT65 averaging, this implementation works on audio sample level. Other modes might benefit as well. If diversity is incorporated into mainstream package, then some additional parameters can be added to decoder functions, enabling a more universal, readable and flexible approach.

10 Conclusion

Results of using diversity on daily activity are positive. Hence the patch deserves to enter mainstream code in some later release.

Beside making several QSOs including rare DX, author learned a lot by studying WSJTX code and enjoyed every step of it.