# FreeDV-060 Radio Autoencoder (RADE) V2 Test Report

David Rowe VK5DGR

November 21, 2026 Git: `2b0902e` on branch `dr-v2-report-phase-d`

## 1 Introduction

This document describes the tests performed on the RADE V2 waveform, and presents the test results. Tests have been performed by the author using software simulation (Phase A), the author using radios and HF radio channels (Phase B), and by crowd sourcing stored file (Phase C) and real time PTT (Phase D) over the air results from the Amateur Radio community.

The RADE concept evolved from a discussion between Jean-Marc Valin and David Rowe, after which Jean-Marc quickly put together an initial proof-of-concept demo in late 2023. Over 2024 David built on this work to develop a practical over the air waveform for speech over HF radio channels. This waveform is denoted RADE V1. Over the course of 2025 and early 2026 David has been developing RADE V2, which has several innovations that are novel to both amateur and indeed professional radio communications:

- Using ML to handle HF channel phase distortion, fine frequency and timing offsets.
- No pilot or unique word symbols, saving bandwidth and power.
- A $N_c = 14$ carrier waveform with a 800 Hz 99% power occupied bandwidth (OBW), that is narrower and cleaner than V1.
- PAPR defined by training, making it relatively insensitive to transmitter filtering.
- Low latency frame of around 40 ms compared to 180 ms for comparable (e.g. RADE V1 or classical) waveforms.
- ML based frame sync that functions without pilot or unique word symbols.

Jean-Marc again assisted with suggestions for timing and frequency offset estimation algorithm based on autocorrelation of the received signal.

The FreeDV Project Leadership Team and many others have helped with support and testing. The contributions from David and the FreeDV PLT was generously supported by a grant from Amateur Radio Digital Communications (ARDC).

Table 1: Glossary.

| Term | Description |
|---|---|
| Acquisition | The process of finding a RADE signal and starting to decode it. |
| AGC | Automatic Gain Control. |
| AWGN | Additive White Gaussian noise - a channel model without fading. |
| loss | A dimensionless measure of distortion in our ML model, based on the distance between the vocoder feature vectors at the encoder and decoder. |
| EOO | End of Over detection system, a special frame that is used by the transmitter to tell the receiver to stop decoding immediately. |
| ML | Machine Learning. |
| MPD | Multipath disturbed - a channel model for a very poor HF channel with 2 Hz fading and 4 ms delay spread. |
| MPP | Multipath Poor - a channel model for a poor HF channel with 1 Hz fading and 2 ms delay spread. |
| OBW | (99% power) Occupied bandwidth - the bandwidth in Hz that contains 99% of the transmitted power. |
| OTA | Over the Air |
| Pilot symbols | Symbols inserted into a transmitted signal to aid in synchronisation. This approach was used for RADE V1 but consumes carrier power and RF bandwidth. |
| PAPR | Peak to Average Power Ratio, a low PAPR is desirable. |
| PNR | Peak power to noise ratio - a useful metric for SSB radio transmitters that are limited by their peak power. |
| SNR | RMS signal power to noise ratio, a useful metric for receivers. |
| RADE | Radio Autoencoder |

# 2 Phase A - Simulation Tests

## 2.1 Configuration

Table 2: Configuration for test campaign.

| | |
|---|---|
| Date | Feb-Mar 2026 |
| Git Hash | 3e06b6 8 April 2026 |
| ML encoder/decoder | 250725 |
| ML sync | 250725a_ml_sync |

## 2.2 General Performance

The script *./compare_models_inf.sh -p 260203_inf* was used to generate loss versus SNR/PNR curves (plotted in Figure 1) for the following conditions:

- RADE V1.
- RADE V2 250725 network with no timing estimator, state machine, or frequency offset estimator (labelled Genie).
- RADE V2 250725 network with real world estimators and state machine (labelled rx2).
- All waveforms were plotted for Additive White Gaussian Noise (AWGN) and Multipath Poor (MPP) channels.

The RADE V1 simulation is run using the *inference.sh* script with phase equalisation but without timing estimation. Spot checks of RADE V1 indicate the inclusion of the timing estimator has no measurable affect on loss versus SNR.

The V2 250725 curves with (labelled *rx2*) and without (labelled *Genie*) estimators are coincident, showing the estimator algorithms have a small impact on performance.

The high loss on the green V2 curve at less than -1 dB is a measurement artefact due to the state machine occasionally falling out of sync at low SNRs; decoded audio is still possible as it recovers and re-syncs.

Examining the AWGN curves on the loss versus SNR plot, for any SNR above 5 dB, the loss of V2 is less than V1.

The performance target for V2 was 3 dB over V1. Taking into account the PAPR gains (loss versus PNR curve), gains over V1 of approximately 3 dB can be observed at low PNRs on the MPP curves.

## 2.3 MPD Channel

The target channels for RADE V2 are AWGN and MPP (Doppler 1 Hz, Delay 2 ms), however we would like RADE V2 to successfully decode speech sent over Multipath Disturbed (MPD) channels, which have a Doppler of 2 Hz and Delay of 4 ms. Some additional loss over MPP is acceptable (e.g. equivalent to a few more dB for maintaining a constant loss), but complete breakdown is not. The *v2_spot.sh* tool was used for this test, and the loss from the *rx2.sh* stage reported in Table 3. Both the loss and the decoded speech from the MPD channels was quite acceptable, meeting the requirements.

## 2.4 Occupied Bandwidth and PAPR

Table 4 compares the 99% power occupied bandwidth (OBW) and PAPR for RADE V1 and RADE V2. The 56 second test file *all.wav* was used for this test, and both signals were constrained by a simulated SSB filter (roughly 400-2600

Table 3: Loss by channel

| Channel | EbNodB | SNR (dB) | Loss |
|---|---|---|---|
| AWGN | 100 | 100 | 0.080 |
| AWGN | 10 | 3.56 | 0.136 |
| MPP | 100 | 100 | 0.104 |
| MPP | 10 | 3.56 | 0.182 |
| MPD | 100 | 100 | 0.107 |
| MPD | 10 | 3.56 | 0.193 |

Hz). We measure PAPR using the peak magnitude over an entire sample (100% CCDF) as even an occasional peak will depress a typical SSB transmitter ALC for several seconds. The PAPR tends to vary with source material by a fraction of a dB. Figure 2 plots the power spectrum of both waveforms, the RADE V2 signal is considerably cleaner. The RADE V1 OBW is effectively set by the bandwidth of the SSB filter. OTA measurements of OBW in Phase D (Section 5.1) are a close match to these simulated results.

Table 4: Occupied Bandwidth and PAPR for *all.wav*.

| Waveform | OBW | PAPR (100% CCDF) |
|---|---|---|
| RADE V1 | 2109 | 4.20 |
| RADE V2 | 867 | 3.48 |

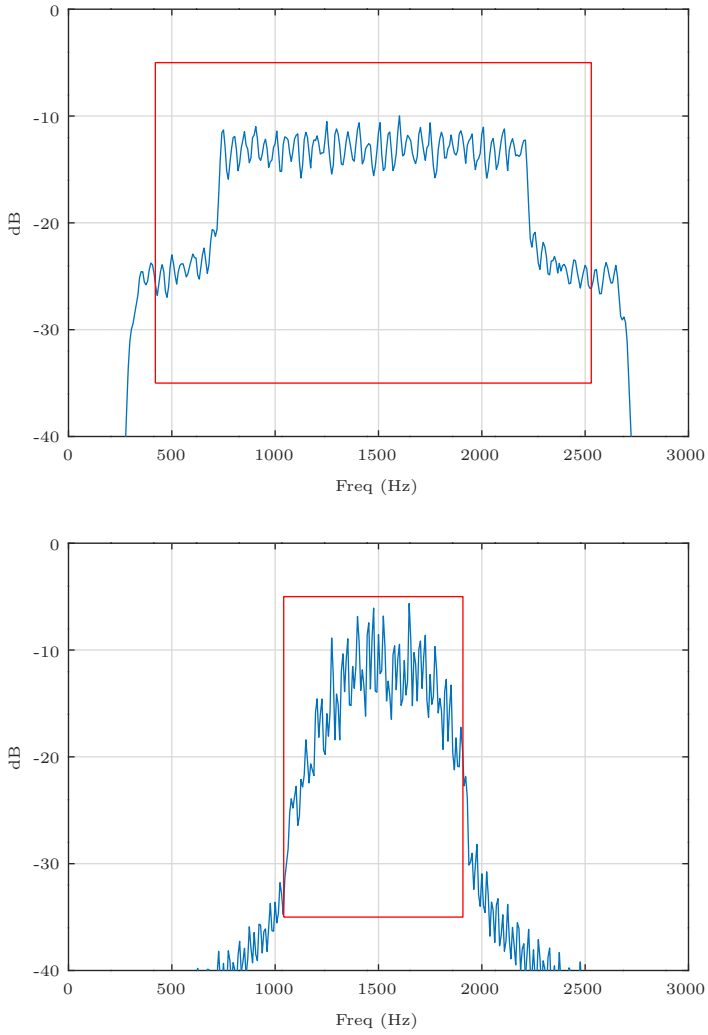
Cmd line for RADE V1

```
./inference.sh model19_check3/checkpoints/checkpoint_epoch_100.pth wav/all.wav
 /dev/null --rate_Fs --pilots --pilot_eq --eq_ls --cp 0.004 --bottleneck 3
 --auxdata --write_rx rx.f32 --tanh_clipper --ssb_bpf
```

Cmd line for RADE V2

```
./inference.sh 250725/checkpoints/checkpoint_epoch_200.pth wav/all.wav
 /dev/null --rate_Fs --latent-dim 56 --peak --cp 0.004 --time_offset -16
  --correct_time_offset -16 --auxdata --w1_dec 128 --write_rx 250725_rx.f32
  --freq_offset 25 --correct_freq_offset --ssb_bpf
```

Figure 2: Power Spectrum of RADE V1 and V2 with SSB filter.




## 2.5 Acquisition

Several properties of the acquisition subsystem were tested:

1. Acquisition time from when a RADE V2 signal starts to when it is acquired in various channel conditions.

2. Probability of false acquisition on noise or single carrier sine wave signals when no RADE V2 signal is present.

3. Ability to acquire if a sine wave signal is present, then lose sync when the V2 signal ceases.

4. Ability to re-acquire a second RADE V2 signal (with a different frequency offset and timing) just after the first V2 signal ends.

Acquisition time was characterised using *v2_spot.sh*. Note we prepend 1 second of silence in these tests which is subtracted for the reported *acq_time* to obtain the results in the Table 5. Acquisition time becomes longer as SNR drops, which is quite acceptable. High SNR acquisition time on a good channel of 200 ms is usable (similar to RADE V1) but a little slow given the decoder latency of 40 ms so as further work it would be desirable to improve this. This is partially due to state machine design that is optimised for low SNR channels. It may be possible to modify the state machine to transition quickly if the probability of a valid signal is high, for example sum the probabilities until a threshold is reached rather than using a fixed time (state machine counter) to validate the signal.

Table 5: Time to acquisition $T_{aq}$.

| Channel | EbNodB | SNR (dB) | $T_{aq}$ (ms) |
|---|---|---|---|
| AWGN | 100 | 100 | 200 |
| AWGN | 10 | 3.56 | 240 |
| AWGN | 1 | -5.44 | 1080 |
| MPP | 10 | 3.56 | 320 |
| MPP | 5 | -1.44 | 1040 |

Example command line for MPP tests:

```
./test/v2_acq.sh --EbNodB 5 --g_file g_mpp.f32
```

The probability of false acquisition on noise was tested using *v2_acq.sh* which plays 600 seconds (10 minutes) of AWGN noise into the *rx2.sh* receiver. The number of acquisitions *n_acq* is printed at the end and should be zero. Over several runs there were zero acquisitions in 10 minutes giving mean time to false acquisition $T_{false} > 600$s.

Example command lines for acquisition tests:

```
./test/v2_acq.sh --n_secs 600
./test/v2_acq.sh --n_secs 600 --sine
./test/v2_acq.sh --n_secs 60 --sine --verbose
```

The probability of false acquisition of noise plus an additive sine wave was tested with *v2_acq.sh* using the *–sine* option. This tests the tendency of the receiver to start decoding audio when there is a carrier wave (e.g a station tuning up) and no valid signal. A 1000 Hz sine wave plus noise resulted in no false acquisitions over a 10 minute run which is a pass.

The ability to acquire with an additive sinewave was tested using *v2_spot.sh*, with results presented in Table 6. In all cases acquisition (sync) is obtained with a sine wave and not held (sync was lost) once the V2 signal is removed. A sine wave outside of the V2 spectrum (e.g. 1000 Hz) has only a small impact on loss. A sine wave inside the V2 signal (e.g. 1505 Hz) has a larger impact on loss but V2 does not fall over - the desired result. As the sine wave level increases to near the V2 level (approximately 1.0) it eventually becomes impossible to obtain sync. A successful sync at a sine wave amplitude of 0.5 indicates additive sine waves at up to -6 dBc can be tolerated, an acceptable result.

Table 6: Acquiring and losing sync with additive sine wave

| Channel | Sine Freq (Hz) | Sine Amp | EbNodB | SNR | Loss | Sync | Lost sync |
|---|---|---|---|---|---|---|---|
| AWGN | - | 0.0 | 100 | 93.5 | 0.080 | Y | Y |
| AWGN | 1000 | 0.2 | 100 | 93.5 | 0.083 | Y | Y |
| AWGN | 1000 | 0.2 | 3 | -3.4 | 0.298 | Y | Y |
| MPP | - | 0.0 | 100 | 93.5 | 0.100 | Y | Y |
| MPP | 1505 | 0.2 | 100 | 93.5 | 0.179 | Y | Y |
| MPP | 1505 | 0.2 | 6 | -0.4 | 0.276 | Y | Y |
| AWGN | 1000 | 0.5 | 10 | 3.6 | 0.147 | Y | Y |
| AWGN | 1000 | 1.0 | 10 | 3.6 | - | N | - |

Example command lines for acquisition tests with additive sine wave:

```
./test/v2_spot.sh --sine_amp 0.2 --sine_freq 1505
--g_file g_mpp.f32 --verbose
```

Consider the case where signal1 follows signal2, such that there is no gap between them. This tests stresses the acquisition system, as it is difficult to design for a system that can distinguish signal2 disappearing into a fade for a short duration, from the case where signal1 replaces signal2. The ability to quickly re-acquire a new signal was tested with the following command line:

```
./test/v2_spot.sh --verbose --prepend_signal2 --prepend_noise 0.31
--EbNodB 10 --reset_output_on_resync
```

This generates signal2 (-25 Hz frequency offset) which is placed just before the target signal1 (25 Hz frequency offset) with 0 ms delay. The test then measures the loss of signal1; a low loss is expected if re-acquisition is successful. The *prepend_noise* argument was chosen so that the timing offset was just on a symbol boundary($\hat{\delta} \approx 160$), a worst case location for the acquisition system. Table 7 presents the results for a few spot tests. The losses are in all case approximately the same as for signal1 alone which indicated successful re-acquisition and a pass. The counter-intuitive lower loss for the MPP result with two signals can be explained by the MPP channel being a little more benign (on average) over the longer test period (signal2 10 seconds, signal1 is 56 seconds in length).

Table 7: Re-acquisition test.

| Case | Channel | EbNodB | SNR | Loss |
|---|---|---|---|---|
| signal1 | AWGN | 100.0 | 3.6 | 0.083 |
| signal2 followed by signal1 | AWGN | 100.0 | 3.6 | 0.083 |
| signal1 | AWGN | 10.0 | 3.6 | 0.134 |
| signal2 followed by signal1 | AWGN | 10.0 | 3.6 | 0.135 |
| signal1 | MPP | 10.0 | 3.6 | 0.181 |
| signal2 followed by signal1 | MPP | 10.0 | 3.6 | 0.102 |

## 2.6 Automatic Gain Control (AGC)

ML networks tend to operate over a limited dynamic range so an AGC is required. The script *./compare_models_inf.sh -p 260830_inf* was used to generate loss versus SNR curves for the first three AGC conditions below, and *./compare_models_inf.sh -p 260831_inf* for the fourth (plotted in Figure 3) on the worst case MPP channel:

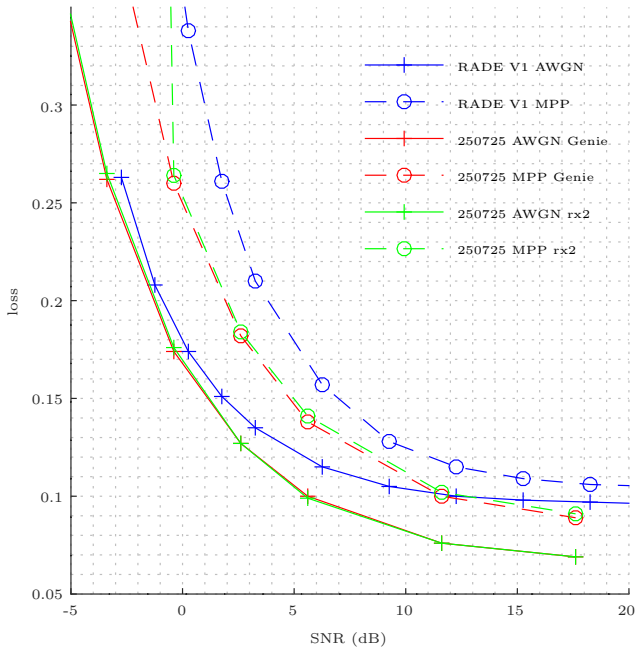
1. Baseline with no AGC.
2. AGC enabled with +20 dB gain (linear gain of 10).
3. AGC enabled with -20 dB gain (linear gain of 0.1).
4. AGC enabled with 0 dB gain (linear gain of 1.0) with three different delays (timing offsets).

This section (Figures 3, 4 and Table 8) was regenerated using *radae* commit `73f7183` and *rade_c* commit `6d89585` on 1 Sep 2026, both later than the campaign configuration in Table 5.

The AGC curves are coincident with the baseline curve, indicating AGC has a small impact on performance over a range of gains and timing offsets. Table 8 presents results for a fine grained sweep of loss against delay for a high SNR, to confirm the loss obtained from the verification procedure will be insensitive to arbitrary timing offsets found in real time systems.

AGC operation is limited to $\pm 20$ dB from the nominal peak level of $A = 1.0$ or $0.1 < A < 10.0$ . Therefore a manual gain term may be required to scale from a typical $\pm 2^{15}$ peak level common with deployments to the range $0.1 < A < 10.0$ at input of the V2 receiver. In terms of RMS levels, given a 3 dB PAPR, we would like $A_{rms} = 0.707$ at the receiver input.

Figure 1: Loss versus SNR and PNR for various RADE V1, and RADE V2 with and without estimators.



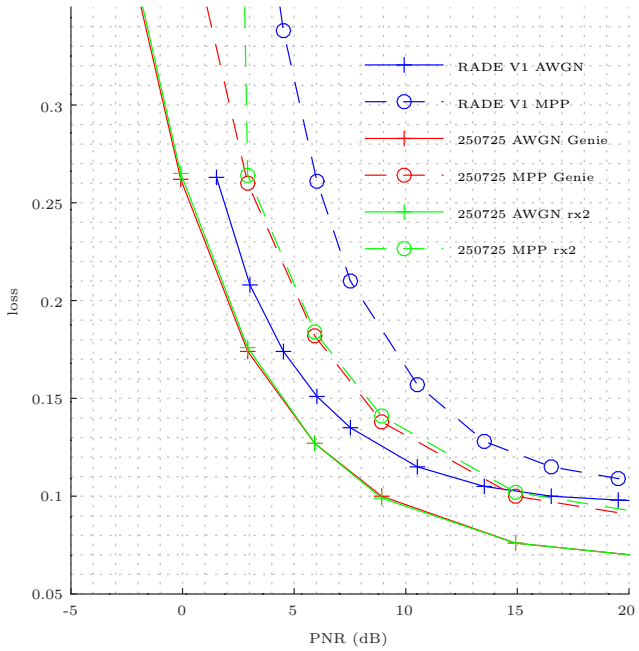
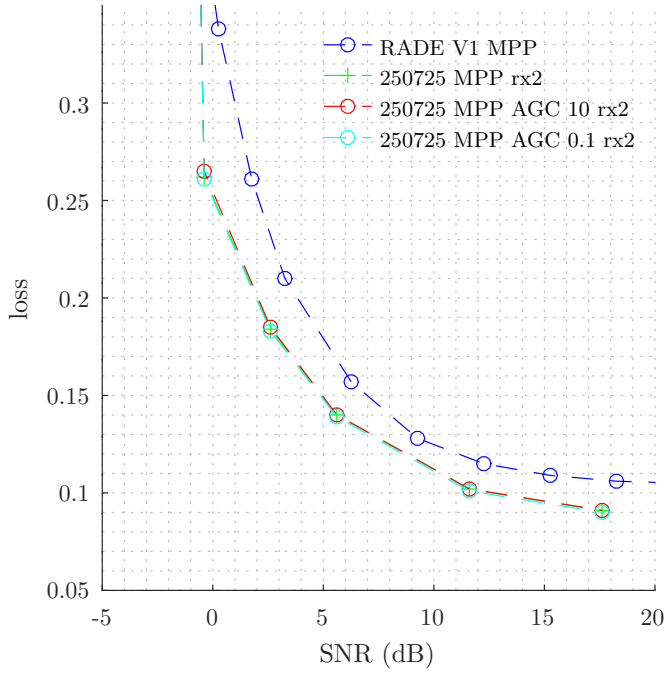



Figure 3: Loss versus SNR for various AGC conditions.



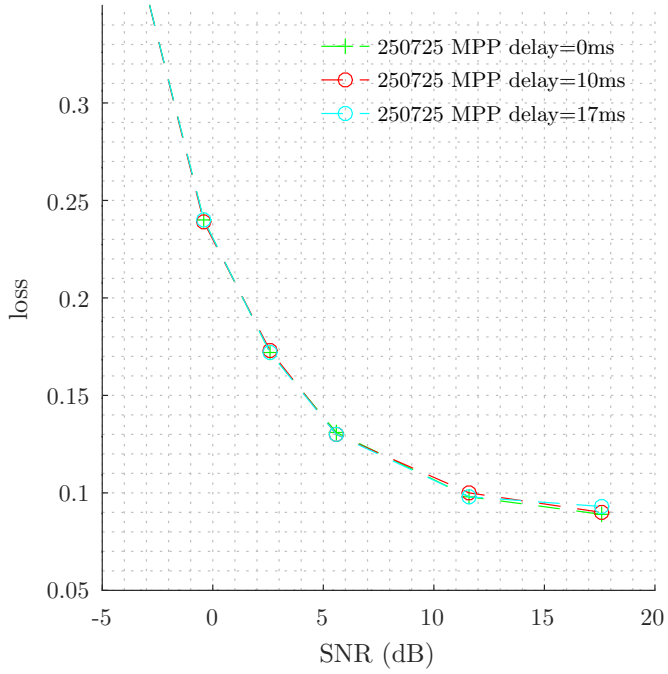



Figure 4 and Table 8 were generated with:

```
PY_OPTS="--agc" RESULTS=v2_delay_loss.txt PLOT=v2_delay_loss ./test/v2_c_timing.sh 0 0.04 40
```

Figure 4:

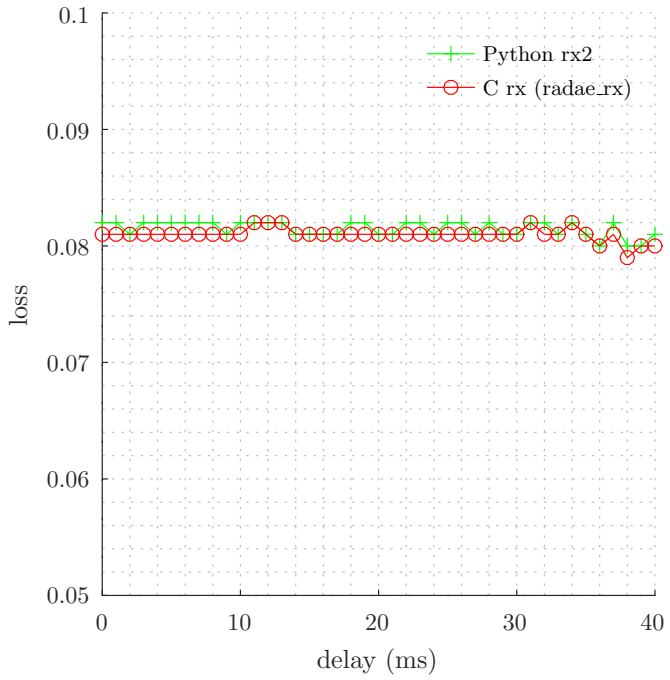



Table 8: Loss versus timing delay, EbNodB=100 (AWGN, effectively no channel noise), wav/all.wav, 0-40ms sweep (41 points, spans more than two symbol periods, $M = 128$ samples = 16 ms).

| Receiver | Min Loss | Max Loss | Mean Loss | Std Dev |
|---|---|---|---|---|
| Python (rx2.py) | 0.080 | 0.082 | 0.0815 | 0.0006 |
| C (radae_rx –v2) | 0.079 | 0.082 | 0.0810 | 0.0005 |

## 2.7 Sample clock and freq drift tests

A Tx/Rx sample clock offset of ±25 ppm (e.g. $Fs = 8000$ Hz at the Tx and $Fs = 8000.2$ Hz at the Rx) was tested over the 56 second *all.wav* sample. No significant change in loss was observed and the receiver remained in sync.

```
./test/v2_spot.sh --sample_clock_offset 7999.8 --write_Ry_smooth Ry_smooth.c64
 --write_delta_hat delta_hat.f32 --pad_samples 120
```

A Rx frequency offset drifting at -5 Hz per minute was tested. No significant change in loss was observed and the receiver remained in sync.

```
delta=0.02 ./test/v2_spot.sh --g_file g_mpp.f32 --EbNodB 10 --df_dt -0.1
```

## 2.8 End of Over Detection

This section describes the End of Over (EOO) system performance. When the transmission is about to cease, a special EOO frame is sent to instruct the receiver to immediately stop decoding. This prevents decoding of spurious received signal and unwanted "R2D2" tones emerging from the decoder at the end of a transmission. The design goal is reliable operation in high SNR benign channels. In poor channels (e.g. low SNR MPP), it is acceptable to have a lower probability of EOO detection, as clean end of over behaviour is a secondary requirement in the low SNR use case. In all channels, should the EOO frame not be detected, the hangover counter in the receiver state machine will eventually stop decoding after a few seconds.

The script *eoo_detect_prob.sh* generates $N = 20$ by 9 s duration samples (extracted from *all.wav*) each terminated in an EOO frame, transmitted through *tx2.py* $\rightarrow$ *ch* $\rightarrow$ *rx2.py* so that both the payload and the EOO frame are genuinely faded and filtered by the channel (an earlier version of this test spliced an AWGN-only EOO tail onto the channel output, so multipath fading never actually reached the EOO frame – see Issue #1). The number of successful EOO detections is counted to determine the probability of EOO detection for each condition. The 9 s duration is chosen to ensure the alignment shifts between *all.wav* and *g_mpp.f32* as the trials modulo rotate around each file. AWGN detection probability declines steadily from 65% at 4.3 dB SNR to 5% at 0.3 dB SNR; MPP is weaker at comparable SNR, declining from 55% at 8.3 dB SNR to 10% at 2.3 dB SNR, since a real fade can occasionally land on the EOO frame itself.

Table 9: End of Over Probability of Detection over $N = 20$ trials

| Channel | SNR3k (dB) | P(detect) |
|---|---|---|
| AWGN | 80.3 | 0.95 |
| AWGN | 4.3 | 0.65 |
| AWGN | 2.3 | 0.40 |
| AWGN | 0.3 | 0.05 |
| MPP | 80.3 | 0.85 |
| MPP | 8.3 | 0.55 |
| MPP | 5.3 | 0.25 |
| MPP | 2.3 | 0.10 |

The script *eoo_false_prob.sh* counts how many times the EOO detection system detects a EOO frame when none is present (false detection), e.g. due to the regular payload V2 OFDM symbols resembling an EOO frame. This is undesirable as the receiver would stop decoding mid-over and have to re-acquire. The longest source file tested was *librispeech.wav* (used for ASR testing) at 540 seconds with no false detections. Over $N = 20$ trials each, no false detections were observed on *all.wav* at 4.4 dB SNR over AWGN or 5.4 dB SNR over MPP.

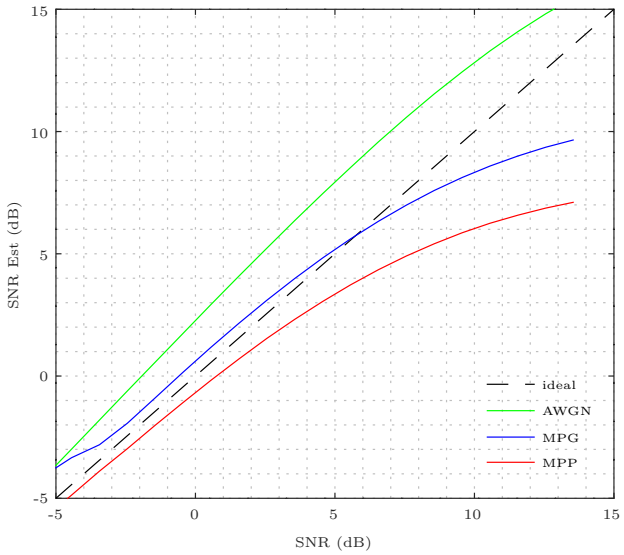
Sample command lines:

```
./test/eoo_detect_prob.sh --channel mpp --No -25
./test/eoo_false_prob.sh
```

## 2.9 SNR Estimator

Figure 5 illustrates the SNR estimator performance. SNR estimation for RADE V2 is challenging due to the lack of pilot symbols. The algorithm used is based on the autocorrelation of cyclic prefix and works well for AWGN channels, but accuracy suffers in multipath channels. A straight line correction was applied minimise the error over three common channels over the -5 to +5 dB range. It is most accurate for slowly fading channels (MPG).

Figure 5: Mean estimated SNR against actual SNR for several channels with noise measured in a 3 kHz bandwidth. Over the -5 to +5 dB interval the RMS error is 1.5 dB with worst case 2.8 dB.



Sample command line:

```
./test/snr_est_test.sh --channel mpp | tee snr_est_mpp.txt
```

### 2.10 Streaming Text

The RADE V2 waveform provides a 1 BPSK symbol/frame (25 bit/s) for auxiliary text information such as callsign and grid square. An external library is required for framing and error correction. Table 10 presents the uncoded BER performance of the BPSK bit stream. The 49 second wave file *all.wav* was used for the source speech (1400 bits total).

Table 10: BPSK bit error rate (BER) for various channels

| Channel | EbNodB | SNR (dB) | BER |
|---|---|---|---|
| AWGN | 100 | 100 | 0.00 |
| AWGN | 10 | 3.5 | 0.01 |
| AWGN | 6 | -0.5 | 0.047 |
| AWGN | 3 | -3.5 | 0.103 |
| MPP | 100 | 100 | 0.005 |
| MPP | 10 | 4.5 | 0.035 |
| MPP | 6 | -0.5 | 0.086 |
| MPP | 3 | -3.4 | 0.136 |

A modern FEC code will be able to correct errors in the several percent range and break down at about 10% BER. This suggests reliable text transfer can occur to around -3 dB on AWGN channels and 0 dB on fading channels. It is acceptable if text transfer breaks down at very low SNRs, as we would prefer to prioritise speech at low SNRs. The text information tends to be repeated, so it is not necessary to receive every encoded text frame, even an occasional decode of a text frame would be acceptable over a several minute over.

## 3 Phase B - Initial Over the Air Tests

This section describes initial OTA tests using the stored file test system implemented by the script *ota_test.sh*. These were performed by the author. The purpose of these tests was (a) see if RADE V2 actually works over real world HF channels (b) compare V2 to RADE V1 to determine if there is any improvement at low and high SNRs. The simulated results presented above predict a 3 dB increase in sensitivity at low SNRs. At high SNRs the ML network "loss" is significantly lower for V2, but we are not sure what that means in terms of speech quality.

The *ota_test.sh* script generates a file consisting of a chirp – SSB RADE V1 – RADE V2 in sequence. This was transmitted from the authors home station (IC7200 transmitter) to various KiwiSDRs around Australia. The chirp is used to estimate SNR of the channel, and SSB and RADE V1 are included for

comparison. By sending all 4 signals at approximately the same time, we have them passing over roughly the same HF channel. In addition, several tests were conducted "Over The Cable" (OTC) - using a local (FT817) receiver connected to the transmitter via attenuators and cable - providing a "perfect" channel. A total of 17 test were performed during March 2026. In each case RADE V2 worked well on real HF channels, supporting the functional test aim (a) above.

Selected speech samples are available here.

It can be difficult to discern fine differences in speech samples using ad hoc listening tests. To help with this problem, the OTA test software measures the "loss" between the vocoder features at the input of the RADE encoder, to the output of the RADE decoder. Table 11 is a table of the "loss" for a selection of OTA and OTC samples. It can be seen that when sent over the same channel at (almost) the same time, the RADE V2 samples achieve an equal or lower loss than RADE V1, supporting the simulation results above.

Table 11: Loss for various samples from the OTA test campaign. The "Software" columns are the loss from software encoding only, before transmission over the channel. A perfect channel (like v216) achieves very close to these loss values.

| Sample | Speaker | V1 software | V2 software | Channel | V1 OTA | V2 OTA |
|---|---|---|---|---|---|---|
| v208 | David | 0.102 | 0.077 | NVIS | 0.132 | 0.132 |
| v209 | David | 0.102 | 0.077 | NVIS | 0.141 | 0.125 |
| v211 | David | 0.102 | 0.077 | DX | 0.520 | 0.154 |
| v216 | Brian | 0.115 | 0.086 | OTC | 0.123 | 0.096 |
| v217 | Brian | 0.115 | 0.086 | DX | 0.311 | 0.198 |

The OTC tests (e.g. v216) show very little impact on the loss metric, demonstrating that the signal processing through two typical HF radios (USB sound hardware, SSB filters, PA, up and down conversion) has very little impact on the RADE waveforms. This suggests loss measurements can be used as a metric for "type approval" of RADE integration e.g. with a SDR, end user application, or hardware radio.

# 4 Phase C - Crowd Sourced Stored File OTA Tests

This section describes the test results collected from the crowd sourced stored file OTA test campaign. The goals of this test campaign were:

1. Gather further stored file OTA samples to support the Phase A & B results, but from a wider range of speakers, HF channels, radio hardware.

2. If any counter examples to the Phase A & B results are encountered, explore and fix bugs if found.

3. Measurements of PAPR from other radios and test equipment.

4. If encountered, explore the "underwater" voice modulation artefact often heard on V1 samples posted on social media.

5. Gather samples from real time, PTT use of RADE V2 when integrated into 3rd party applications or radios.

6. Gain experience in 3rd parties integrating and verifying RADE V2 operation in their application/radio. Use this information to improve the integration and verification procedure.

Licensed Amateur Radio operators from around the world were invited to record a 10-60 second input sample of their own voice. This was converted to a RADE V1 and V2 waveform sample, and concatenated with a Hilbert-compressed version of the same input sample to emulate an analog SSB signal with a 6-8 dB PAPR. Participants then played the concatenated sample through their SSB transmitters over real world HF Radio channels to remote KiwiSDR receivers of their choice (KiwiSDRs are SSB radio receivers that are connected to the public Internet). The received off air signal was processed to obtain a file of SSB and RADE output speech samples, and an estimate of channel SNR. The use of stored files enabled the same Tx signal to be transmitted to different KiwiSDR receivers under different channel conditions, and different transmit power levels. The *ota_test.sh* script performed the Tx and Rx processing using the reference Python implementation of RADE V2. For convenience, this was configured to run from a web server.

The procedures for stored file tests are documented here.

Samples submitted included the source audio fed to the RADE encoder, plus the received off air signal. The idea is to have enough information to reproduce the results (e.g. by passing through the RADE decoder) and measure the loss compared to a software simulation reference. They have been archived on the FreeDV project shared drive.

Table 12 summarises the test results. A total of 58 samples were collected. Special thanks to to Rick W7YC, Kanda JH0PCF (long samples), Yuichi JH0VEQ and Take JS1MAV for your samples.

Table 12: Phase C Test Results

| Goal | Target | Actual |
|---|---|---|
| Number of samples collected | 20 | 58 |
| Number of languages | 3 | 3 |
| PAPR measurements | 3 | 1 |

We achieved the goal of successful functional testing across a range of languages, radios, and HF channels, with no counter examples to the Phase A & B results. No examples were found of complete breakdown of V2 when V1 worked. These tests established the functional goal of demonstrating that RADE V2 works (classical DSP and ML subsystems) over real world HF channels. Participants were asked to rank (unblinded) each sample and a simple binomial

Table 13: Subjective ranking results ($n = 58$, 1=best, 3=worst).

| | Mean rank | Std | vs SSB | |
|---|---|---|---|---|
| | | | Win / Loss / Tie | $p$ |
| SSB | 3.00 | 0.00 | — | — |
| V1 | 1.48 | 0.50 | 58 / 0 / 0 | $< 0.001$ |
| V2 | 1.33 | 0.47 | 58 / 0 / 0 | $< 0.001$ |
| V2 vs V1 | | | 28 / 19 / 11 | 0.12 |

statistical test applied (Table 13). There was a clear preference for both RADE V1 and V2 over SSB ($p < 0.001$), however the preference for V2 over V1 did not reach statistical significance ($p = 0.12$). This result is expected, a 3dB difference is difficult for the human ear to resolve, especially over real world non-stationary HF channels. Within statistical limits, the ranking is consistent with the simulation results above. No evidence (in either V1 or V2 decoded samples) was found of the "underwater" voice modulation artefact that has previously been heard on YouTube samples of RADE V1.

Only one PAPR measurement was submitted (Table 14, thanks Take). This was on the baseband signal using Audacity and was consistent with the simulation results. As further work it would be useful to obtain PAPR measurements at the PA RF output.

Table 14: Phase C PAPR results from Take

| | CHIRP | SSB | RADE V1 | RADE V2 |
|---|---|---|---|---|
| | 0 | 7 | 4.20 | 3.50 |
| PEAK | -6.02 | -6.02 | -6.02 | -6.02 |
| RMS | -9.03 | -18.29 | -12.66 | -12.26 |
| PAPR(dB) | 0.00 | 9.26 | 3.63 | 3.23 |

# 5 Phase D - Real Time PTT OTA Tests

This section describes the test results collected from the crowd sourced PTT OTA test campaign. The goals of this test campaign were:

1. Functional testing of RADE V2 OTA in real world PTT conversations. It should acquire, sync and decode without major failures such as crashing or garbled speech.

2. Successful integration of the 40 ms delay, for example no clicks in audio (short buffers can stress sound drivers), and evaluating the PTT turn around compared to RADE V1 (180ms).

3. Gain experience in 3rd parties integrating and verifying RADE V2 operation in their application/radio (e.g. software and over the cable loss tests). Use this information to improve the integration and verification procedure.

4. Gather further examples of off air received signals and decoded V2 speech to support the previous test results, but in real time from a range of speakers, HF channels, radio hardware.

5. If any issues are encountered from *verified* integrations, explore and fix bugs.

6. Measurements of PAPR at the PA output and compare to simulation.

7. If encountered, explore the "underwater" voice modulation artefact often heard on V1 samples posted on social media.

8. It is not within the scope of this test phase to evaluate speech quality (e.g. compared to V1 or SSB), as no way of performing controlled tests is possible.

The procedure for testing is documented here. Any results must be from an application or radio that has RADE V2 integration verified using this procedure. Any applications contributing to this test campaign must be capable of exporting a received off-air signal sample file, and a decoded V2 speech file.

As real time PTT samples have no reference input, the loss cannot be measured, so the Phase D results are qualitative. The campaign ran for about three weeks. Table 15 summarises the samples collected, which fell short of the targets. Thank you to all the Hams that participated and special thanks to Rick W7YC, Yuichi JH0VEQ and Taku Kudo JK1GPC for their samples.

Table 15: Phase D Test Summary.

| Goal | Target Samples | Collected Samples |
|---|---|---|
| Number of OTA samples to collect | 20 | 12 |
| Number of languages | 3 | 2 |
| PAPR measurements | 3 | 2 |

The results against each goal were:

1. Functional testing: five testers reported a pass, and no failures were reported, across conversations with many other stations. Some testers used Thetis, an integration that has not been verified, with similar results to freedv-gui.

2. 40 ms delay and PTT turn around: Rick W7YC reported good PTT turn around times, faster than RADE V1, including rapid Tx/Rx exchanges over a 66 km NVIS path with flat and frequency selective fading. No issues such as audio clicks were reported.

3. 3rd party integration: integration and verification of freedv-gui and FreeDV Neo (internal to the FreeDV project) found bugs in V2 and helped tune the verification procedure. No 3rd party applications were verified.

4. Off air samples: Rick W7YC provided samples from several radios (Flex, IC-7300, IC-7100) on 20 and 80 m, including 2 to 200 W signals received by remote WebSDR and KiwiSDR receivers, and two way contacts. Yuichi JH0VEQ and Taku Kudo JK1GPC provided Japanese language samples. The decoded samples were consistent with the previous test results.

5. Issues: one issue was found, a false sync when the tuning error exceeds the $\pm 32$ Hz range of the V2 frequency offset estimator. In Yuichi's Japan to Australia sample, a $+44$ Hz tuning error was aliased to $-18.5$ Hz, and V2 synced confidently on the wrong frequency, producing garbled speech, while V1 decoded the same signal. After correcting the frequency offset in software, the V2 samples decoded successfully. This is a corner case affecting a small number of contacts. It is in the classical DSP, so a fix would not break compatibility (no retraining).

6. PAPR: Taku JK1GPC and Babis SV1UH provided PAPR measurements after the campaign, see Section 5.2.

7. "Underwater" artefact: some of Rick's samples may contain examples of this artefact, however the root cause is still unknown. Simulations suggest it may be speaker dependent, but the results are not conclusive. Notably, the operator in question described the RADE speech quality as very good.

Apart from the frequency offset false sync corner case, we conclude RADE V2 is working for real world PTT OTA conversations.

## 5.1 Measured Occupied Bandwidth

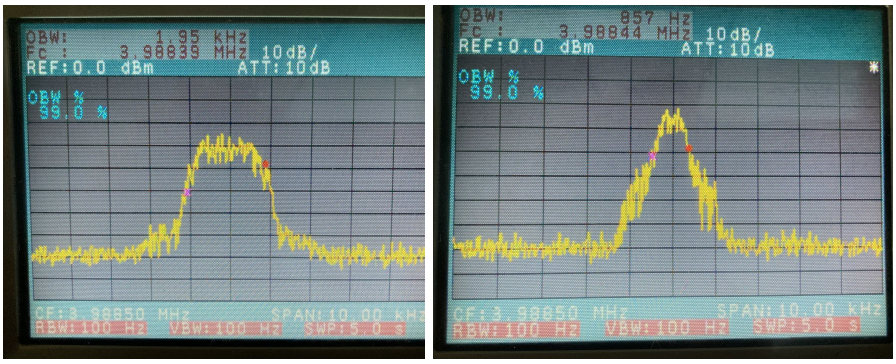
Although not one of the Phase D goals, Rick W7YC independently measured the 99% power occupied bandwidth (OBW) of the transmitted RADE V1 and V2 signals using an Advantest U3641 spectrum analyser (Figure 6). The signals were transmitted on 80 m by an IC-7300 with a 2000 Hz (500-2500 Hz) transmit SSB filter. The analyser settings were 10 kHz span, 100 Hz resolution and video bandwidth, and a 5 second sweep. Multiple runs varied slightly, but not significantly.

Table 16 compares the measurements to the simulated results in Table 4. As expected, the RADE V1 OBW is set by the bandwidth of the transmit SSB filter, which is a little narrower than the simulated SSB filter. The RADE V2 measurement is a close match to simulation. The OBW will vary a little with the source material (e.g. different speakers) and the averaging time.

Table 16: Measured and simulated 99% Occupied Bandwidth.

| Waveform | Measured OBW (Hz) | Simulated OBW (Hz) |
|---|---|---|
| RADE V1 | 1950 | 2109 |
| RADE V2 | 857 | 867 |

Figure 6: Measured 99% OBW of RADE V1 (left, 1.95 kHz) and RADE V2 (right, 857 Hz) transmitted by an IC-7300, courtesy of Rick W7YC.



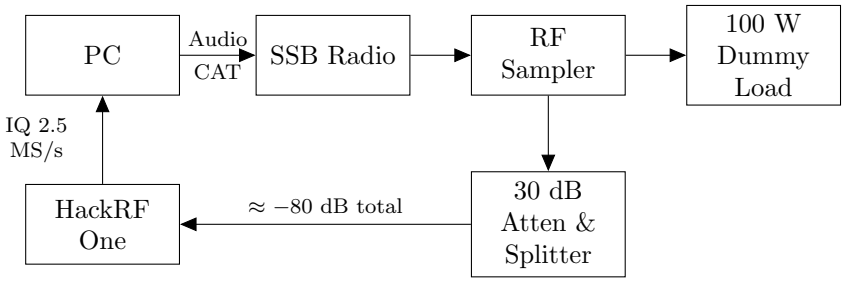

## 5.2 Measured PAPR

After the campaign, Taku JK1GPC and Babis SV1UH measured the PAPR of the transmitted V2 signal by sampling the RF with an SDR, which prompted the measurements in this section (thanks also to Take JS1MAV, Helmut DC6NY and Rick W7YC for the discussion). The goal was to measure the PAPR at the PA output of real SSB radios, and to separate the effects of the radio from the waveform.

Figure 7 shows the test setup. The test signal was *david_vk5dgr.wav* (11.6 seconds) modulated with `rade_tx_wav --v2`, played into the radio's USB audio interface with `aplay`, with the radio in data mode (LSB-D) on 7.175 MHz, the radio's speech compressor off, and the RF power control at maximum. The drive level was set by the audio level. PTT and the radio's meters (power, ALC, SWR, supply current) were controlled and logged over CAT with Hamlib. An RF sampler, attenuator and splitter provided $\approx 80$ dB of coupling to a HackRF One (`hackrf_transfer -f 7100000 -s 2500000 -a 0 -l 32 -g 20`), keeping its input below $-25$ dBm at 100 W. The coupling was calibrated with a tracking generator, and the absolute power is accurate to about $\pm 0.5$ dB. Two 100 W radios were tested, an IC-7200 and an IC-7300.

The IQ samples were mixed to baseband, decimated to 100 kHz, and band limited to 4 kHz (zero phase) centred on the signal. In simulation the PAPR is unchanged for measurement bandwidths above 1.6 kHz, and the measured SNR in 4 kHz was 46 to 55 dB, enough to keep the noise contribution to PAPR below

Figure 7: PAPR test setup.




0.02 dB. The first second after PTT was excluded, as the gain of both radios settled by about 1 dB over this period. As a check, at full drive a single tone measured 0.06 dB PAPR (ideal 0 dB), and a two tone signal was within 0.1 dB of the ideal 3.01 dB. The captures were also time aligned with the transmitted signal, so we could measure the radio's AM-AM and AM-PM response, and decode the captured signals to measure loss.

Table 17 lists the configuration used for these tests.

Table 17: Configuration for PAPR tests.

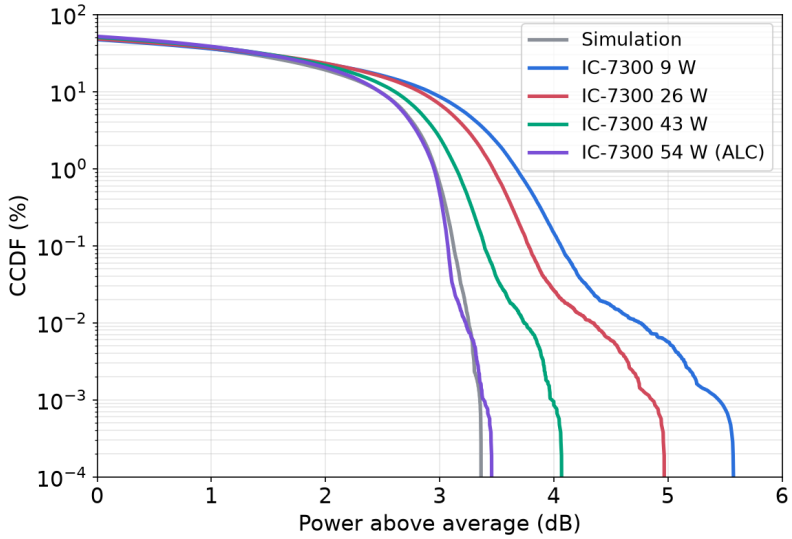
| | |
|---|---|
| Date | November 2026 |
| RADE Tx | rade_c `rade_tx_wav --v2`, git 54613bb (signal identical to main 4a31f25) |
| Loss measurement | radae `loss.py`, git e5d4c15 |
| ML encoder/decoder | 250725 |
| Test signal | *david_vk5dgr.wav*, 11.6 s |
| SDR | HackRF One, `hackrf_transfer` git-1408473 |
| Radio control | Hamlib 4.3.1 `rigctl` |

Table 18 summarises the results, with 100% CCDF (peak) and 0.1% CCDF PAPR values, ranging from low to high power. Figures 8 and 9 show the IC-7300 CCDF and AM-AM results at four power levels.

Table 18: Measured PAPR of RADE V2 at 100% and 0.1% CCDF, ranging from low to high average power.

| Source | Transmitter | Power (W) | 100% (dB) | 0.1% (dB) |
|---|---|---|---|---|
| Sim. *all.wav* | - | - | 3.48 | 3.06 |
| Sim. *david_vk5dgr.wav* | - | - | 3.36 | 3.12 |
| Taku JK1GPC | IC-7300MK2 | 10 - 80 | 5.5 - 4.5 | - |
| Babis SV1UH | - | 0.5 - 60 | 8.4 - 6.5 | 5.3 - 4.6 |
| This work | IC-7200 | 7 - 44 | 5.0 - 3.5 | 4.2 - 3.2 |
| This work | IC-7300 | 8 - 54 | 5.6 - 3.5 | 4.1 - 3.1 |

Figure 8: Measured CCDF of the IC-7300 RADE V2 signal at four average power levels, compared to simulation. The PAPR is the power at which the curve meets the bottom axis (100% CCDF).



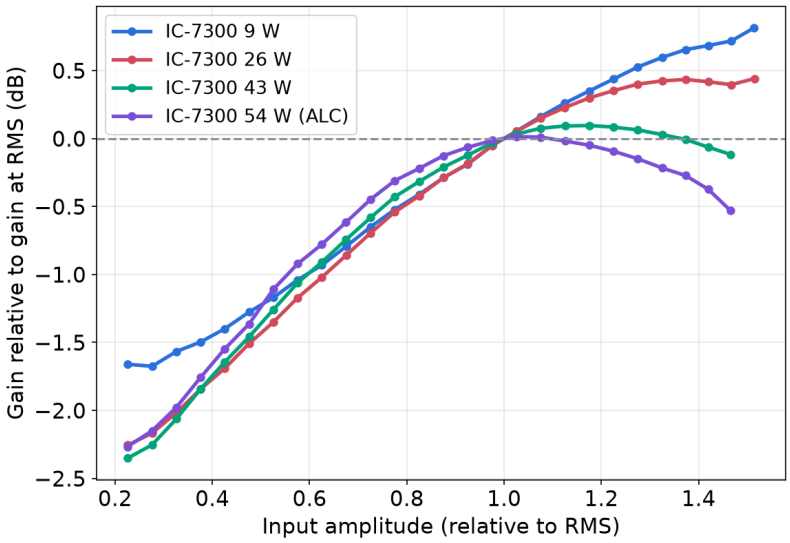

Our results were:

1. The measured PAPR falls with increasing power, consistent with Taku's results. At low to medium power both radios show gain expansion (Figure 9), the gain rising by $\approx 2.5$ dB from low amplitudes to the peaks, which raises the PAPR by $\approx 1$ dB. Near full drive the PA compresses the peaks, cancelling the expansion, and the PAPR approaches the simulated value. The radios' transmit filtering adds 0.2 to 0.5 dB, and the AM-PM is negligible ($< 2°$). Two tone tests gave 3rd order IMD of -23 to -35 dBc, typical of HF SSB transmitters.

2. With 100 W peaks and no ALC action, both radios delivered 43 to 45 W average power, i.e. the peak power less the PAPR. With the drive increased until the ALC just responds the IC-7300 delivered 54 W, and 59 W when driven 6 dB into the ALC, with mild ALC "pumping" ($\approx 1$ dB gain variation).

3. The decoder loss of the captured signals matched the clean simulated signal at all drive levels (including 6 dB into the ALC) and at all SNRs tested. On these radios at least, RADE V2 appears insensitive to the drive level. A simple set up procedure is to set the RF power control to maximum, and increase the audio drive until the ALC just responds.

Figure 9: Measured AM-AM response of the IC-7300 at four average power levels. The rising curves show gain expansion; at 43 and 54 W the PA compresses the peaks.




4. For comparison we transmitted the same speech as SSB through the IC-7300 at the same peak power. Uncompressed SSB averaged 2 to 4 W, the Hilbert compressed SSB used in our simulations (`ota_test.sh`) 21 to 27 W, and the author's voice via the microphone with the Icom speech compressor 12.5 W. RADE V2 delivers about 6 dB more average power than real world compressed SSB. The SSB peaks also briefly exceeded the ALC set point (117 to 163 W), a source of splatter.

5. Babis measured off air with an SDRplay receiver, so his results include noise and HF channel effects that tend to increase the measured PAPR.

6. PA heating: for a class B PA the DC current is proportional to the RF amplitude $a$ and the RF power to $a^2$, so the dissipation is proportional to $a - a^2$, which peaks at $a \approx 0.64$ of the maximum amplitude. With a PAPR of $\approx 3$ dB, the RMS amplitude of RADE V2 is $\approx 0.7$ of the peak, close to this worst case. From the IC-7300's supply voltage and current, the radio dissipated 178 to 190 W with RADE V2 at 43 to 59 W average, compared to 134 to 141 W for a steady carrier at 89 to 100 W. Like other continuous duty digital modes, long overs at high average power stress the PA thermally more than SSB speech.

Recommendations for operating RADE V2 with an SSB radio:

1. Use data mode with the speech compressor off, set the RF power control to maximum, and increase the audio drive until the ALC just responds. Overdriving a little costs little.

2. Ensure a low resistance DC supply connection and a good match (e.g. check any internal tuner). During these tests a high resistance DC lead (11 V at the radio) and a mistuned internal tuner (SWR 2:1, 2 dB loss) were found and fixed.

3. Expect $\approx$ 45 to 55 % of the rated peak power as average power, and for the radio's power meter to fluctuate. The first second of each over is not representative, as the radio's gain settles.

4. For long overs, consider reducing the RF power to manage PA heating. The decoded speech quality is not very sensitive to power on good links.

5. Further work: test radios with different PA and ALC designs (e.g. non-Icom, SDR based).

# 6 Conclusion

In Table 19 the results from Phase A, B, C and D are summarised and compared to requirements extracted from FreeDV-038. Only performance and functional requirements have been extracted (e.g. we don't consider implementation issues like a C port or packaging).

## 6.1 Further Work

Topics for future development. It may be possible to attend to these in parallel with crowd sourced V2 testing, as they have little impact of core requirements such as speech quality and robustness.

1. Reduce acquisition time for benign channels.

2. Improved EOO performance at lower SNRs.

3. Extend range of frequency offset estimation to $\pm$50Hz.

Table 19: Test results compared to performance requirements

| Requirement | Met | Comment |
|---|---|---|
| Speech quality v SNR $\geq$ RADE V1 on AWGN & MPP. | Pass | $\approx$ 3 dB gain of V2/V1 in simulations, but statistical significance not reached in OTA tests (expected result). |
| Handles MPD channel without breaking down | Pass | |
| PAPR $<$ 1 dB— through a typical SSB band pass filter. | Partial | This requirement was found to be in error, real world PAPR of RADE V1 was $>$ 4 dB. We have achieved a PAPR of $\approx$ 3 dB with V2, a 1 dB gain on V1. Measured 3.1 to 3.5 dB at the PA output at full drive. |
| OBW $<$ 2000 Hz | Pass | V2 OBW measured as $\approx$ 860 Hz, much cleaner spectrum. |
| Sample clock offsets $\pm$125 ppm | Pass | No significant loss at $\pm$125 ppm, syncs and decodes at $\pm$250 ppm. |
| Frequency Offsets $\pm$50 Hz | Partial | Current algorithm natively handles $\pm$32 Hz, should be possible to extend this with further work. A Phase D sample with a +44 Hz offset caused a false sync. |
| Clean end of over detection within 100 ms | Pass | EOO works well at moderate to high SNRs. |
| 25 bit/s streaming text for callsign | Pass | Waveform can send 1 bit/frame over channel, streaming text supported by external library. |
| SNR estimation | Pass | Error of 1.5 dB RMS ($\pm$3dB max) over -5 to +5 dB. |