$7

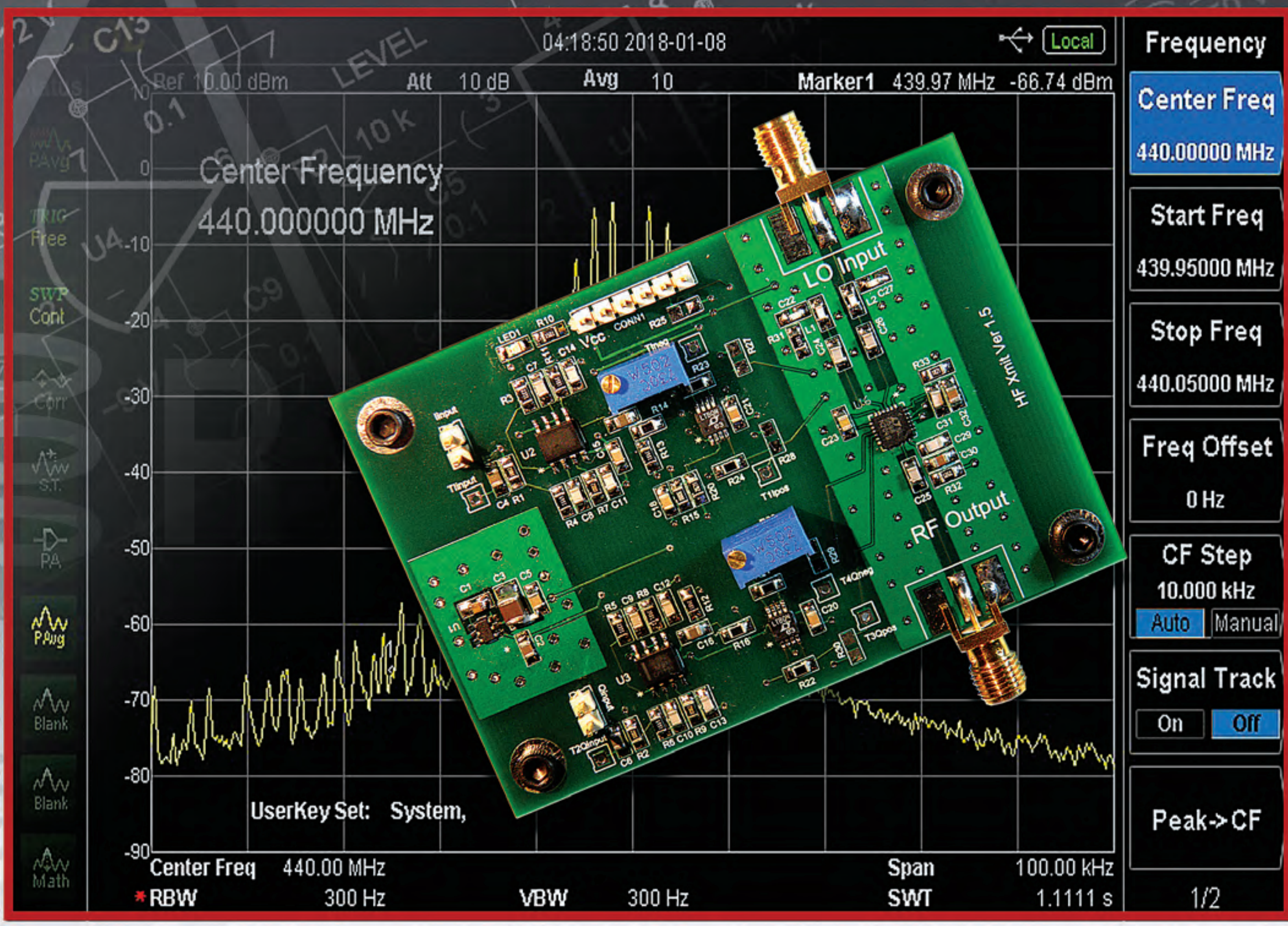
QEX

March/April 2019

www.arrl.org

A Forum for Communications Experimenters

Issue No. 313



ACØZJ describes a baseband quadrature modulator that operates over multiple amateur bands.

ARRL The national association for AMATEUR RADIO™
225 Main Street
Newington, CT USA 06111-1494

KENWOOD

3rd IMDR 110 dB*

RMDR 122 dB*

BDR 150 dB*

Performance Exceeding Expectations.

The most happy and sublime encounters happen in the worst circumstances and under the harshest conditions.

There are enthusiasts who know this all too well because of their love of HF radio.

Results born of certainty and not circumstance. Delivered through impeccable performance. This is our offering to you.

"The Kenwood TS-890S has the highest RMDR of any radio I have ever measured."
- Rob Sherwood - NC0B - December 2018

HF/50MHz TRANSCEIVER TS-890S

NEW

Top-class receiving performance

3 kinds of dynamic range make for top-class performance.

- Third order intermodulation Dynamic Range (3rd IMDR) 110dB*
- Reciprocal Mixing Dynamic Range (RMDR) 122dB*
- Blocking Dynamic Range (BDR) 150dB*

*Values are measured examples. (2kHz spacing:14.1 MHz, CW, BW 500 Hz, Pre Amp OFF)

- Full Down Conversion RX
- High Carrier to Noise Ratio 1st LO
- H-mode mixer

4 kinds of built-in roofing filters

500Hz / 2.7kHz / 6kHz / 15kHz (270Hz Option)

7 inch Color TFT Display

- Roofing frequency sampling band scope
- Band scope auto-scroll mode
- Multi-information display including filter scope

Clean and tough 100W output

Built-in high-speed antenna tuner

32-bit floating-point DSP for RX / TX and Bandscope

Customer Support: (310) 639-4200

www.kenwood.com/usa

JQA-1205 091
ISO9001 Registered
JVCKENWOOD Corporation

ADS#02119

QEX (ISSN: 0886-8093) is published bimonthly in January, March, May, July, September, and November by the American Radio Relay League, 225 Main St., Newington, CT 06111-1494. Periodicals postage paid at Hartford, CT and at additional mailing offices.

POSTMASTER: Send address changes to: QEX, 225 Main St., Newington, CT 06111-1494
Issue No 313

Publisher
American Radio Relay League

Kazimierz "Kai" Siwiak, KE4PT
Editor

Lori Weinberg, KB1EIB
Assistant Editor

Zack Lau, W1VT
Ray Mack, W5IFS
Contributing Editors

Production Department

Steve Ford, WB8IMY
Publications Manager

Michelle Bloom, WB1ENT
Production Supervisor

Sue Fagan, KB1OKW
Graphic Design Supervisor

David Pingree, N1NAS
Senior Technical Illustrator

Brian Washing
Technical Illustrator

Advertising Information Contact:

Janet L. Rocco, W1JLR
Business Services
860-594-0203 – Direct
800-243-7768 – ARRL
860-594-4285 – Fax

Circulation Department

Cathy Stepina, *QEX Circulation*

Offices

225 Main St., Newington, CT 06111-1494 USA
Telephone: 860-594-0200
Fax: 860-594-0259 (24 hour direct line)
e-mail: **qex@arrl.org**

Subscription rate for 6 issues:

In the US: $29;

US by First Class Mail: $40;

International and Canada by Airmail: $35

Members are asked to include their membership control number or a label from their QST when applying.

In order to ensure prompt delivery, we ask that you periodically check the address information on your mailing label. If you find any inaccuracies, please contact the Circulation Department immediately. Thank you for your assistance.

About the Cover

Braddon Van Slyke, ACØZJ, describes the design and construction of a quadrature modulator based on the LTC5598. The modulator takes baseband I and Q signals, such as from digital to analog converters or left and right output of a sound card, a local oscillator signal, and puts out an RF signal up to 1 mW. The RF output is filtered and can then be amplified for transmitting on the ham bands. When used as an image-reject up-converting mixer, it is suitable as a modulator building block in a transmitter or transceiver.

In This Issue

Features

Index of Advertisers

The American Radio Relay League

The American Radio Relay League, Inc, is a noncommercial association of radio amateurs, organized for the promotion of interest in Amateur Radio communication and experimentation, for the establishment of networks to provide communications in the event of disasters or other emergencies, for the advancement of the radio art and of the public welfare, for the representation of the radio amateur in legislative matters, and for the maintenance of fraternalism and a high standard of conduct.

ARRL is an incorporated association without capital stock chartered under the laws of the state of Connecticut, and is an exempt organization under Section 501(c)(3) of the Internal Revenue Code of 1986. Its affairs are governed by a Board of Directors, whose voting members are elected every three years by the general membership. The officers are elected or appointed by the Directors. The League is noncommercial, and no one who could gain financially from the shaping of its affairs is eligible for membership on its Board.

"Of, by, and for the radio amateur," ARRL numbers within its ranks the vast majority of active amateurs in the nation and has a proud history of achievement as the standard-bearer in amateur affairs.

A *bona fide* interest in Amateur Radio is the only essential qualification of membership; an Amateur Radio license is not a prerequisite, although full voting membership is granted only to licensed amateurs in the US.

Membership inquiries and general correspondence should be addressed to the administrative headquarters:

ARRL
225 Main St.
Newington, CT 06111 USA
Telephone: 860-594-0200
FAX: 860-594-0259 (24-hour direct line)

Officers

President: Rick Roderick, K5UR
P.O. Box 1463, Little Rock, AR 72203

Chief Executive Officer: Howard Michel, WB2ITX

The purpose of *QEX* is to:

1) provide a medium for the exchange of ideas and information among Amateur Radio experimenters,

2) document advanced technical work in the Amateur Radio field, and

3) support efforts to advance the state of the Amateur Radio art.

All correspondence concerning *QEX* should be addressed to the American Radio Relay League, 225 Main St., Newington, CT 06111 USA. Envelopes containing manuscripts and letters for publication in *QEX* should be marked Editor, *QEX*.

Both theoretical and practical technical articles are welcomed. Manuscripts should be submitted in word-processor format, if possible. We can redraw any figures as long as their content is clear. Photos should be glossy, color or black-and-white prints of at least the size they are to appear in *QEX* or high-resolution digital images (300 dots per inch or higher at the printed size). Further information for authors can be found on the Web at **www.arrl.org/qex/** or by e-mail to **qex@arrl.org**.

Any opinions expressed in *QEX* are those of the authors, not necessarily those of the Editor or the League. While we strive to ensure all material is technically correct, authors are expected to defend their own assertions. Products mentioned are included for your information only; no endorsement is implied. Readers are cautioned to verify the availability of products before sending money to vendors.

Kazimierz "Kai" Siwiak, KE4PT

Perspectives

The Radio Range Knob

We are well into the low portion of the sun spot cycle with relatively poorer ionospheric propagation. This will extend perhaps for a long time into the future. The common knowledge is that the most important part of the station is an efficient antenna system. That will help, as will the choice of operating band and operating times. So what else is different at the bottom of this cycle? This cycle we have a few more choices on the 'radio range knob' or 'link margin knob' on our radio systems — we can choose to operate using the new radio-link efficient operating modes. With the *WSJT-X* modes like JT9, JT65, and FT8, we can see many decibels deeper into the darker ionosphere than ever before. But how does that compare to the decades-old era of spectacular peak sun spot cycles?

The *QEX* readership have a long institutional memory. Many of you may remember operating during those spectacular sun spot cycle peaks of the past, well before the current digital era. How do your recent digital-era experiences compare with those dramatic cycles of decades ago? Let us know in a Technical Note; our more-recent ham readers may like to know!

In This Issue

We feature a range of topics in this issue of *QEX*.

Phil Salas, AD5X, builds a highly linear two-tone test generator for transceiver IMD testing.

Jim Koehler, VE5FP, automates a simple toaster oven for reflow soldering.

Braddon Van Slyke, ACØZJ, makes a base-band quadrature modulator that operates over multiple bands.

Jan M. M. Simons, PAØSIM, use noise cancelling and noise reduction techniques to extract signals from noise.

Keep the full-length *QEX* articles flowing in, or share a **Technical Note** of several hundred words in length plus a figure or two. Let us know that your submission is intended as a **Note**. *QEX* is edited by Kazimierz "Kai" Siwiak, KE4PT, (**ksiwiak@arrl.org**) and is published bimonthly. *QEX* is a forum for the free exchange of ideas among communications experimenters. The content is driven by you, the reader and prospective author. The subscription rate (6 issues per year) in the United States is $29. First Class delivery in the US is available at an annual rate of $40. For international subscribers, including those in Canada and Mexico, *QEX* can be delivered by airmail for $35 annually. Subscribe today at **www.arrl.org/qex**.

Would you like to write for *QEX*? We pay $50 per published page for articles and Technical Notes. Get more information and an Author Guide at **www.arrl.org/qex-author-guide**. If you prefer postal mail, send a business-size self-addressed, stamped (US postage) envelope to: *QEX* Author Guide, c/o Maty Weinberg, ARRL, 225 Main St, Newington, CT 06111.

Very best regards,

Kazimierz "Kai" Siwiak, KE4PT

Braddon Van Slyke, ACØZJ

13111 W. 60th Ave., Arvada, CO 80004; **blindpigsaloon@q.com**

Baseband Quadrature Multi-Band Modulator

This baseband quadrature modulator operates over multiple amateur bands.

I have been looking for a way to experiment with and assemble a software defined transceiver using separate components such as the transmitter, receiver, local oscillator, and so on. For transmitting, I looked at discrete baseband IQ modulators, especially the Linear Technology LTC5598 baseband IQ modulator IC with a range of 5 to 1600 MHz. This range is met with proper local oscillator port matching to a range of frequencies, but I've had reasonable success with a single match from 50 to 450 MHz.

This article describes the design and construction of an IQ modulator based on the LTC5598. The modulator takes baseband 'I' and 'Q' signals, such as from digital to analog converters or left and right output of a sound card, a local oscillator signal, and outputs an RF signal up to 1 mW. The RF output contains odd order harmonics, but with filtering the output can then be amplified for transmitting on the ham bands. When used as an image-reject up-converting mixer, it is suitable as a modulator building block in a transmitter or transceiver for such things as APRS, beacons, and voice using SSB or FM. Other modulation modes are possible as well.[1]

Design

Figure 1 shows a block diagram of the LTC5598 modulator board. Two excellent

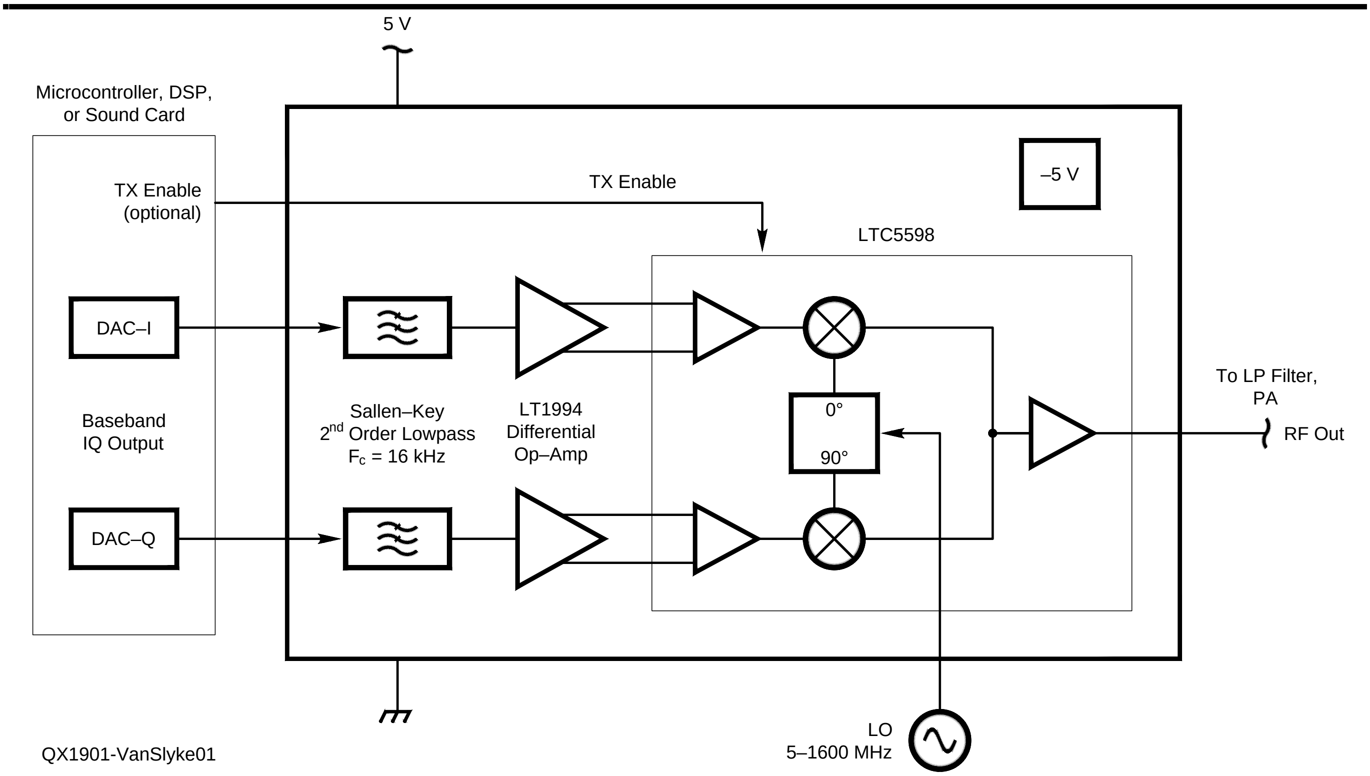


Figure 1 — Block diagram of the LTC5598 modulator board.

sources about IQ or quadrature modulation, are *Understanding the 'Phasing Method' of Single Sideband Demodulation* [2] and *Basics of IQ Signals and IQ modulation & demodulation – A Tutorial* [3]. I also encourage you to read the *SDR Simplified*, and *Hands On SDR* articles in previous *QEX* issues.

Consider this short review of IQ baseband modulator basics. Transmitting a tone signal, for instance 600 Hz, would be designated as the 'I', or in-phase signal. The quadrature signal, designated 'Q', is the 'I' signal phase shifted -90° from the 'I' signal. In digital signal processing a Hilbert Transform is commonly used to phase shift the in-phase signal, and by definition, shifts each frequency components of the 'I' signal by -90° to produce the quadrature signal. Any time/sample delay introduced by the Hilbert transform into the quadrature signal must be matched in the in-phase signal. The in-phase signal can be any arbitrary waveform such as voice. These 'I' and 'Q' signals form the baseband input. The local oscillator (LO) frequency is then set to be the transmit frequency, for instance, 144.270 MHz. At this point, a plot of the RF spectrum output at 144.270 MHz would show one upper-sideband (USB) signal, at 144.270 MHz plus 600 Hz. For the sake of this discussion let's ignore odd-order harmonic products. To create a lower-sideband (LSB) signal at 144.270 MHz minus 600 Hz just invert the 'Q' signal by multiplying by -1. That is, the phase of the 'Q' signal is shifted to be +90° to that of the 'I' signal.

A few key characteristics of the LTC5598 should be summarized. First, the LTC5598 requires the quadrature input signal to be differential, rather than single-ended. The data sheet states[4] "The baseband inputs should be driven differentially; otherwise,

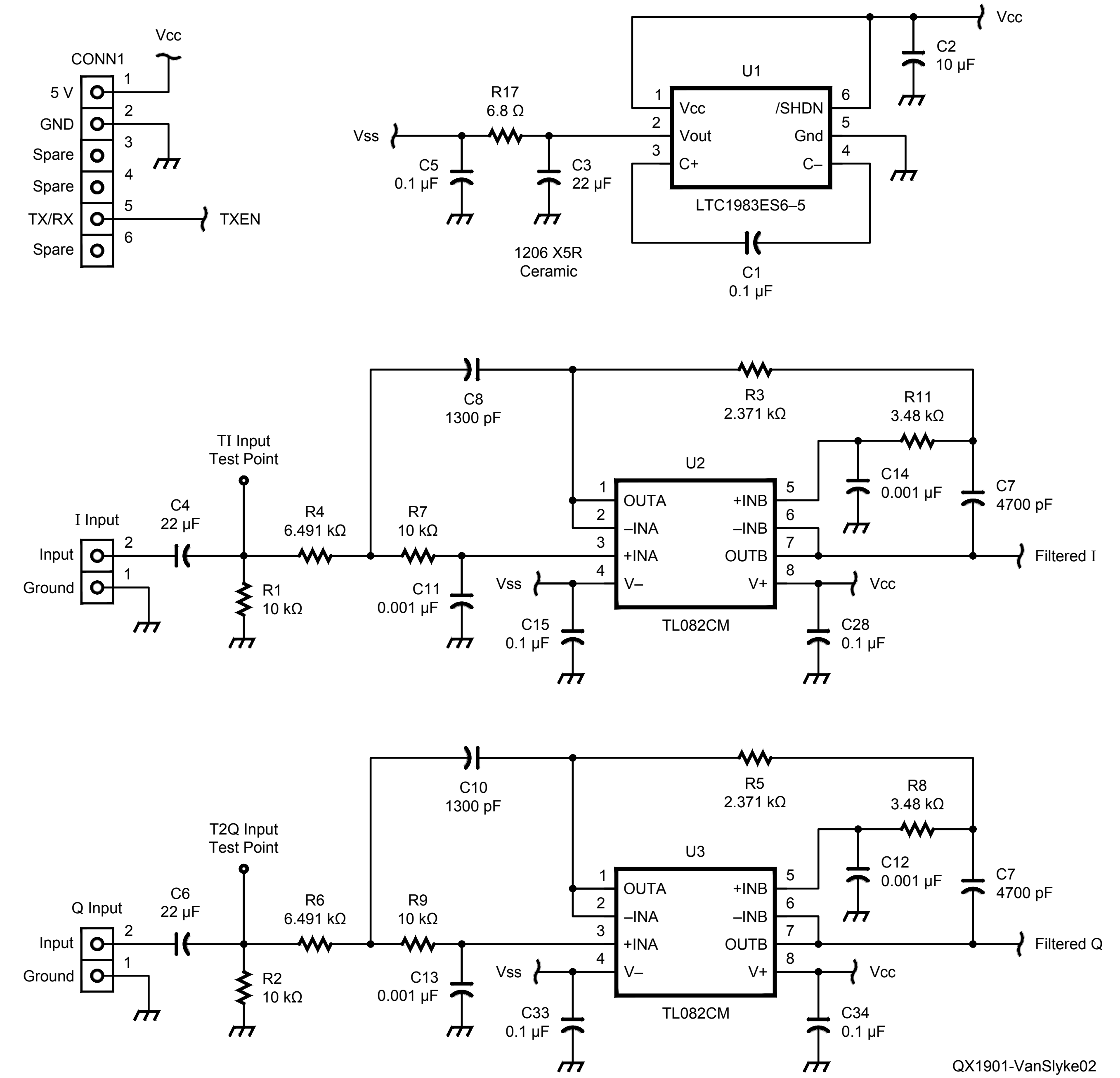


Figure 2 — Schematic of the modulator board showing anti-aliasing filters and the negative 5 V supply.

the even-order distortion products may degrade the overall linearity performance." Single-ended signals have a conductor and a ground, whereas differential signals have two conductors and ground, where the signals on the conductor are 180° apart. This project provides single-ended to differential conversion, so that single-ended DACs or sound card outputs can be used.

Second, odd-order harmonics of the modulated signal are present at the LTC5598 output, and those must be filtered prior to transmitting. Third, the local oscillator must be a sine wave, or images can occur. Images are unwanted signals present on the other side of the LO frequency from the desired signal. "The third-harmonic rejection on the applied LO signal is recommended to be equal or better than the desired image rejection performance since third-harmonic LO content can degrade the image rejection severely. Image rejection is not sensitive to second-harmonic LO content".[5] Lastly, the 'I' and 'Q' inputs of the LTC5598 can be up to 300 MHz in bandwidth. My design does not use nearly that wide a bandwidth, but is limited to 16 kHz, which is sufficient for audio and low data rate operations.

Design

The schematic for the modulator is shown in Figure 2 and Figure 3. I have a DSP evaluation board with dual DACs, which I wanted to use to produce the 'I' and 'Q' inputs to the modulator. The DAC outputs do not have anti-aliasing filters, so anti-aliasing filters are part of the board design. To design the filters, I used the Texas Instrument Filter Designer Tool[6] specifying a passband frequency of 16 kHz, a stop-band frequency of 80 kHz, unity gain in the passband, and selecting the "Linear Phase 0.05 deg" design option. The filter topology is Sallen-Key.

The maximum 'I' and 'Q' input voltage before the output begins to distort is 2 V P-P. The voltage limit is a function of the gain of the differential driver stages. The LTC5598 puts out a little over 0 dBm (1 mW) at this level.

The left and right speaker outputs from a sound card can also be used as the 'I' and 'Q' inputs to the modulator. If speaker outputs are used, the low-pass filter circuits on the modulator board could be omitted or replaced with a gain circuit using a single op-amp. Sound card line-level output voltages are generally too small, but they too could be used by changing the gain of the filters or by replacing with a single op-amp having a specific gain. An example using GNU Octave to generate baseband FM and SSB 'I' and 'Q' signals and outputting the signals to a sound card is shown in the sample Octave code *basebandIQ.m* on the **arrl.org/QEXfiles** web page.

The op-amp and differential driver design requires a negative voltage supply. The modulator needs a differential signal with a common mode voltage of 0.5 V. The differential driver requires a negative voltage supply of at least 1.1 V below the common mode voltage to function properly.

The negative supply uses an LTC1983-5 100 mA regulated charge pump inverter that provides -5 V. Pay attention to the LTC1983 data sheet regarding layout; under General Layout Considerations: "You will not get advertised performance with careless layout".[7] I built a separate negative power supply using dead-bug construction on top of a grounded copper plane on a separate PCB. My latest PCB layout has a ground plane around the chip and short connections from power supply lines with capacitors to ground. It gives good performance with spurs at ±280 kHz at -66 dB relative to the carrier.

The LTC5598 requires differentially driven 'I' and 'Q' signals. A number of ADCs provide differential outputs, but my specific ADCs do not, and neither do PC sound cards. To create the differential signals, I used a pair of LT1994 fully differential input/output amplifier driver ICs. The IC uses typical plus and minus inputs as does a standard op-amp, but it also includes a common-mode voltage input pin that allows setting the common-mode output voltage independent of the input common-mode voltage. A voltage divider sets the common-mode output voltage to 0.5 V.

I had an issue with setting the gain of the driver using the gain equations from the data sheet. The output of the differential driver should be,

$$V_{out} = 0.5\, V_{in} \frac{R_f}{R_{in}}$$

The 0.5 accounts for the gain split between the two differential outputs. With R_f = 806 Ω and R_{in} = 619 Ω, the gain should be V_{out} = 0.65, but empirically I found about 0.5. I can't explain this discrepancy, but for the time I'll just document the difference. Linear Technology makes single-ended to differential driver converters, which in retrospect might have been a better choice.

Looking at the 'I' signal path in Figure 3, the output of the anti-aliasing filters is coupled to the driver through capacitor C17, and connected to the drive negative input resistor R14. Resistor R13 connects the node between C17 and R14 to ground, and is used to prevent leakage current from building a up charge on C17.

Potentiometer R19 is the driver positive input resistor and is tied to ground. For equal differential balancing, its value must be that of R13 in parallel with the source (anti-aliasing filter) resistance, and in series with R14. The source resistance of the anti-aliasing filter is in the hundreds of kΩ, so the value of R19 is very close to that of R13 in series with R14. Imbalance between the resistances on the positive and negative inputs will result in LO leakage. Instead of using a potentiometer, I used a series combination of R13 and R14 at the positive input, with 1% resistors, that results in LO leakage about 40 dB below the primary output signal.

By setting the 'I' and 'Q' inputs to 0 V (or just disconnecting them), and setting the LO to a frequency within an anticipated operating range, the potentiometers can be adjusted to minimize the LO leakage. This method results in LO leakage at or better than 55 dB below the primary output signal. A ball park approximation can be done by measuring the dc voltage on each differential output and adjusting the potentiometers so that the voltages match as closely as possible.

For the circuitry that supports the LTC5598, I used the schematic for the LTC5598 demo board as a reference, part number DC1455A.[8] According to the documentation, the demo board design targets frequencies from 80 MHz to 1300 MHz. I found that the design works at 50 MHz as well. The negative LO input "LOM" is tied to ground through a 50 Ω resistor. The positive LO input requires between 0 and -10 dBm input power. LO levels below -10 dBm will result in poor image rejection. Document DC1455A, section: "Application Note – LO Input Interface" Table 2 lists LO input matching component values *vs*. LO input frequency range. This table allows the designer to optimize the LO circuit matching for a specific range of frequencies. The single-ended RF output is internally matched to 50 Ω over the entire operating range. Resistors R32 and R33 are used to prevent voltage supply ringing.

Construction

The assembled board is shown in Figure 4. The PCB is a double-sided 2.2" by 3.2" board with mostly surface-mount components. I used *DesignSpark* for the schematic and PCB Gerber file generation. The files for the *DesignSpark* project, as well as the Gerbers can be found on the **arrl.org/QEXfiles** web page. I used PCP GOGO[9] to create the PCB in FR-4 board material.

Aside from the ICs, the parts are fairly common and can be ordered from Digikey or Mouser. The resistors used have a 1% tolerance, and the capacitors, except for the 22 μF on the negative power supply, are ceramic NPO types. The Linear Technology parts can be ordered directly from Analog Devices. Linear Technology has a two component minimum when purchasing, but

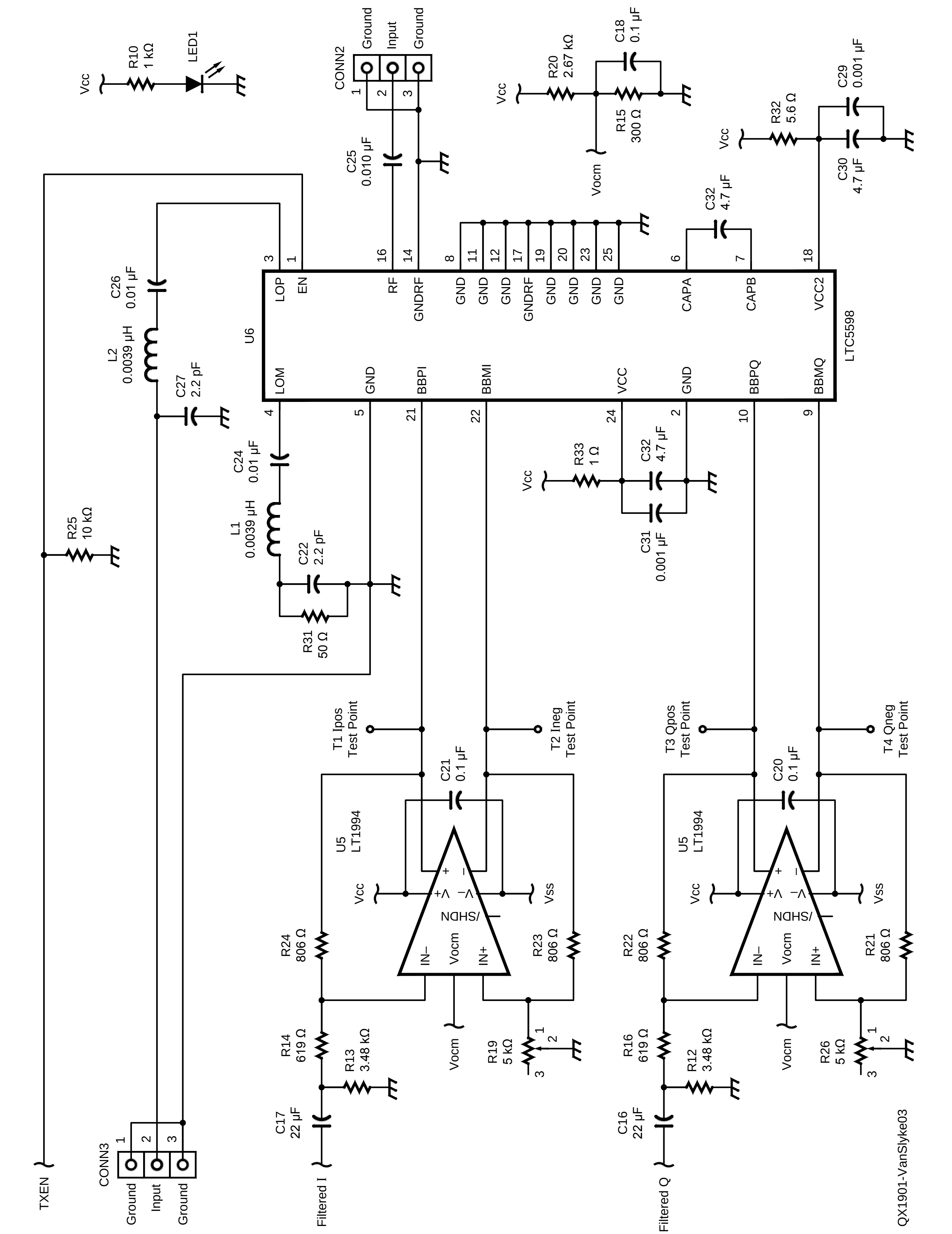


Figure 3 — Schematic of the modulator board showing single-ended to differential drivers.

their cost is much less per piece than buying from Mouser or Digikey. At the time of this writing, the modulator, the charge pump inverter and the driver, when purchased in quantities of two, costs about $24 from Linear Technology (excluding postage). Samples are an option as well, although I have found delivery time takes longer.

Performance and Results

The board current consumption when transmitting is 220 mA at 5 V. My goal when I started this project was to get the modulator working then optimize power consumption later. The LTC5598 alone consumes about 170 mA while transmitting. A similar part, the LTC5599, draws only 28 mA, but it is not a drop-in replacement to the LTC5598.

My local oscillator signal is the output from a Silicon Labs Si5338 into a low-pass filter. The output from the filter is a single sine wave at -10 dBm with third-harmonic attenuated more than 50 dB. The ‘I’ and ‘Q’ inputs are generated by software running on a TMS320F28377S configured to use two of its 12-bit DACs to create the ‘I’ and ‘Q’ signals to the modulator board. The power output of the modulator in the present configuration is about 0 dBm CW, and is fed into a dummy load with a Rigol DSA815 Spectrum Analyzer tapped into the output via a 20 dB attenuator.

The first test creates an upper sideband signal using an LO of 144.270 MHz and a tone at 2000 Hz. The spectrum analyzer output in Figure 5 shows the 144.272 MHz tone at slightly less than 0 dBm output power. The local oscillator leakage is seen to be down approximately -60 dB, and the image rejection at 144.278 MHz is at approximately -50 dB. Image rejection can be further reduced if desired by adjusting the gain and/or phase of the ‘I’ channel relative to the ‘Q’ channel.

Figure 6 shows the same signal but the span is increased 1 MHz. The spurs from the LT1994 are seen at ±288 kHz from the LO, making their way into the LTC5598. If the output of this modulator were to be used to drive an amplifier, then attention must be paid to these spurs especially if the operating frequency causes the spurs to be out of band. Bob Allison, WB1GCM, points out[10] how out of band spurs, or harmonics, must be minimized so that they do not exceed 25 μW(-16 dBm) output. That suggests the output of the modulator could be increased to (-16 dBm) – (-65 dB) = 49 dBm (79 W) before needing further filtering of the out-of-band spurs.

Figure 7 shows a full-span plot. Odd-order harmonics, a product of mixing, can be seen at 144.27 × 3, ×5, ×7, and ×9. Odd-order harmonics are to be expected, and can be easily filtered prior to amplification.

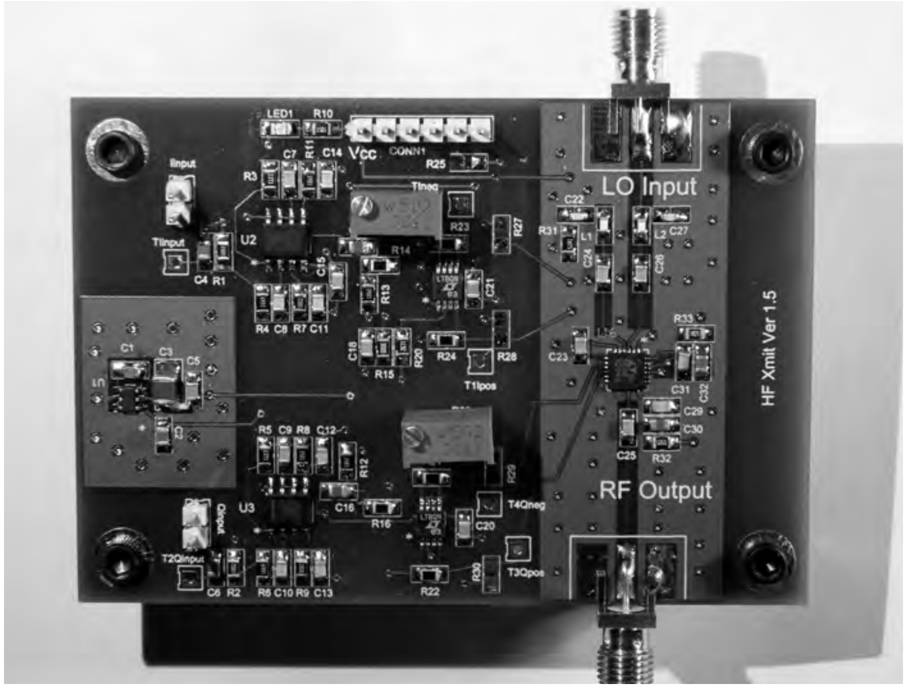


Figure 4 — The assembled LTC5598 modulator board.

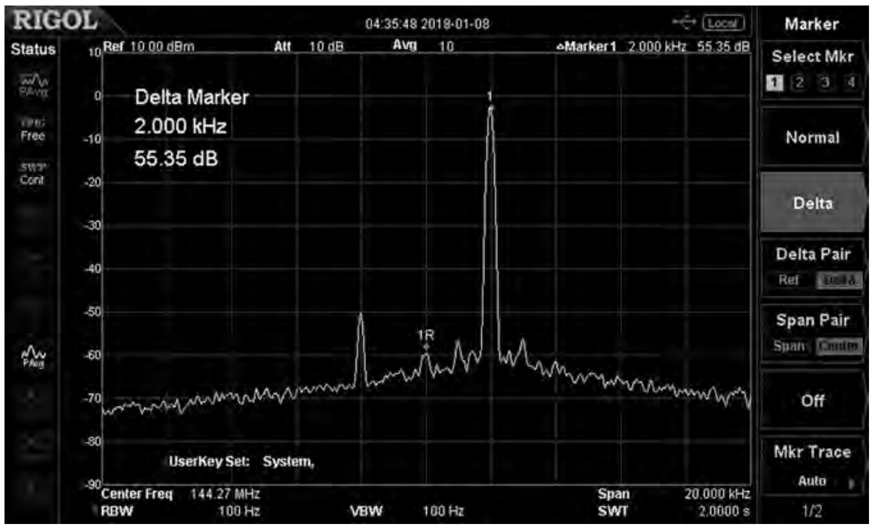


Figure 5 — Spectrum analyzer plot of USB 2000 Hz tone at 144.270 MHz, span is 20 kHz.

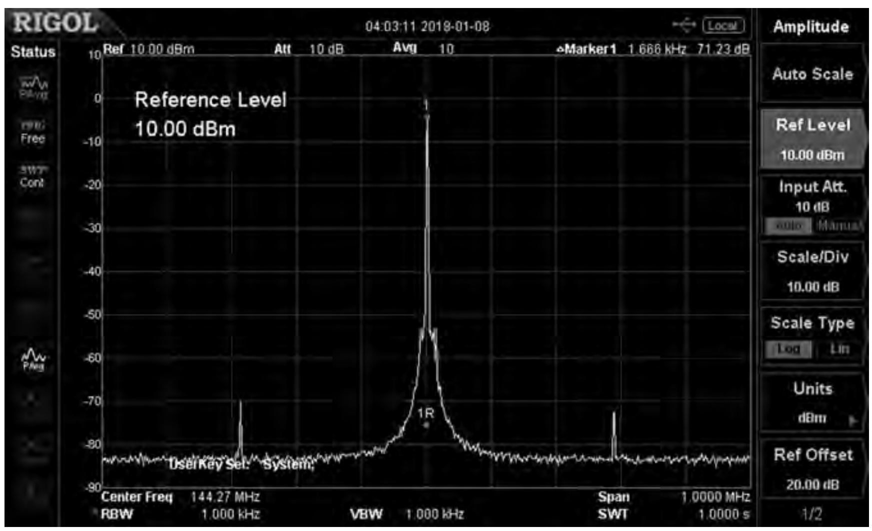


Figure 6 — Spectrum analyzer plot of USB 2000 Hz tone at 144.270 MHz, span is 1 MHz.

Switching to lower sideband is as simple as changing the polarity of the *Q* signal, that is, multiplying by -1, prior to DAC output. Figure 8 shows the lower sideband output 2 kHz below the suppressed carrier frequency of 144.27 MHz. Figure 9 shows a lower sideband signal with the tone below the LO set frequency of 440 MHz. Again, results are very similar to when the LO is set to 144.270 MHz. Results are similar for 50 MHz, with the output slightly lower than 0 dBm even though the LO match isn't optimized for that frequency.

I used an older Wavetek 3520 to generate an output signal at 1290 MHz, albeit with more phase noise than with the Si5338. Figure 10 shows an upper sideband plot at 1290 MHz, which shows that the device should be useable up to the 23 cm band.

Figure 11 shows the output of a frequency modulated (FM) tone. FM modulation with the LTC5598 is relatively straightforward. In the DSP, a 10 kHz cosine wave carrier is modulated with a tone — a voice signal will work just as well. The modulated 10 kHz signal is converted into a composite signal with 'I' and 'Q' outputs, these 'I' and 'Q' signals are sent to separate DAC channels. The DAC outputs go to the modulator board. In this case I applied a 1.9 kHz signal with a modulation index of 5.54, resulting in the expected suppressed carrier. This same tone can be picked up by my two-meter handheld radio with no apparent distortion in the audio.

Conclusion

The LTC5598 is an inexpensive and easy way to modulate baseband signals using SSB, FM, and other modulation methods. With the proper local oscillator, the chip will easily modulate baseband signals from the 6 m band up to 23 cm with an output power up to 1 mW. With the addition of a Raspberry Pi, USB sound card, and filter/amplifier, it could form the core of a beacon or APRS transmitter. It could be used as the modulator for a multi-mode transceiver for operation on the lesser-used 33 cm and 23 cm bands.

Braddon (Brad) Van Slyke, ACØZJ, received his Amateur Extra class license in 2011 and has been obsessed with building and testing anything related to radio ever since. He lives in Arvada, Colorado with his wife and four boys, and is a Digital Signal Processing Engineer employed at LGS Innovations, Inc. He holds undergraduate and graduate degrees in Electrical Engineering from the University of Wyoming and Texas A&M respectively. Brad had been writing software for medical devices including pulse oximeters and surgical generators for over 20 years, then recently switched to writing software for wireless applications.

(Continued on page 12)

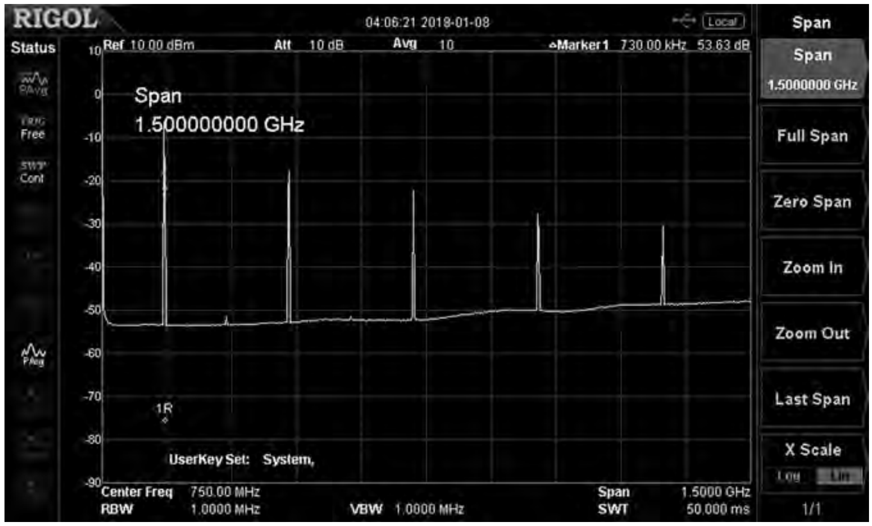


Figure 7 — Spectrum analyzer plot of USB 2000 Hz tone at 144.270 MHz, full span of 1.5 GHz.

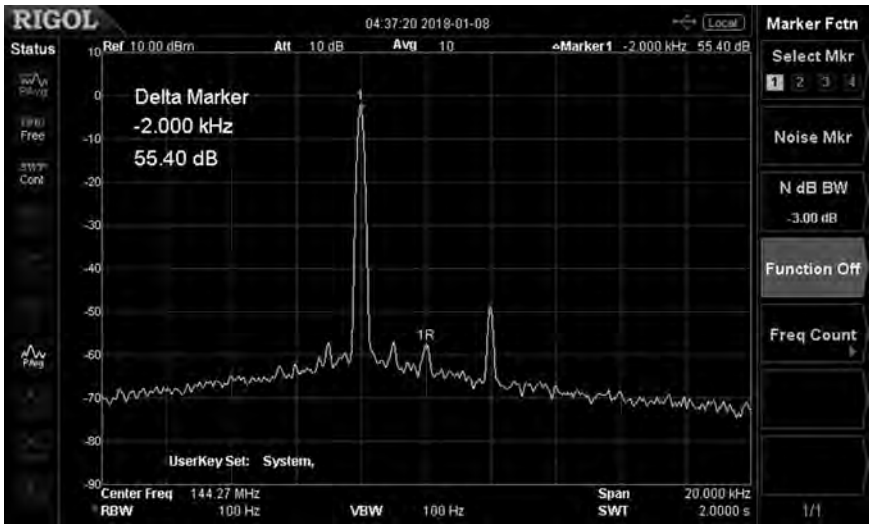


Figure 8 — Spectrum analyzer plot of LSB 2000 Hz tone at 144.270 MHz, span is 20 kHz.

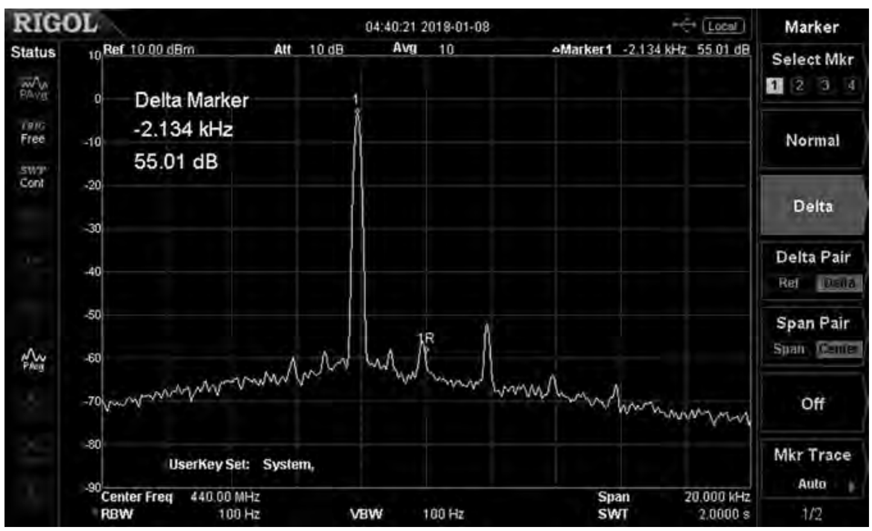


Figure 9 — Spectrum analyzer plot of LSB 2000 Hz tone at 440 MHz, span is 20 kHz

Phil Salas, AD5X

1517 Creekside Dr., Richardson, TX 75081; **ad5x@arrl.net**

Low-Cost Low-Distortion 2-Tone Test Oscillator for Transmitter Testing

This two-tone test generator has highly linear sinusoidal outputs for transceiver IMD testing.

I needed a 2-tone (700 Hz/1.9 kHz) test oscillator for transmitter testing, and went to a popular auction site to see if someone might be selling a Pacific Antenna, Elecraft or PreciseRF unit[1,2,3]. While none of these showed up, a 1 kHz audio oscillator kit did show up — at less than four dollars shipped from Hong Kong. This very popular kit is offered by many suppliers. I ordered two and received them within two weeks.

Description and 2-Tone Generator Assembly

The units are Wien Bridge oscillators, which are known for good frequency and amplitude stability and low distortion[4]. Referring to the example schematic of Figure 1, you'll note that the units are Zener-regulated, reverse voltage protected, and include two potentiometers to adjust distortion and level. Kit assembly is easy, and the average ham should be able to put these together in less than 30 minutes. Before building the units you may wish to use one of the pc boards as a template to mark the mounting holes for your desired enclosure.

The units include 0.01 μF capacitors for *Cf* resulting in a 1 kHz oscillator frequency. In order to achieve the standard 2-tone frequencies, the 700 Hz oscillator uses the included 0.01 μF capacitors paralleled with 3,900 pF capacitors for *Cf*. For the 1,900 Hz oscillator a parallel combination of 2,500 pF and 2,700 pF is used for *Cf*. The oscillator frequency *f* is determined by

$$f = \frac{1}{2\pi RC}$$

where *R* is the existing 16 kΩ resistor and the capacitor is *Cf*. If desired, you can vary the 16 kΩ resistors instead of the capacitors. For example, leaving capacitors *Cf* at the supplied 0.01 μF value, 22.6 kΩ 1% resistors should give you close to 700 Hz, and 8.45 kΩ 1% resistors should give you close to 1,900 Hz. I had a good assortment of capacitors on hand so I left the resistors alone.

For both oscillators, one set of capacitors is mounted normally on the top side of the pc board. The paralleled capacitors are tack-soldered on the back-side of the pc boards. The completed units are shown in Figure 2 (top view) and Figure 3 (bottom view). Nice multi-turn potentiometers are used for the level and distortion adjustments.

Once the pc boards are complete, apply +10 to +16 V dc, and using a frequency counter or oscilloscope, verify that each unit oscillates at the correct frequency. The actual frequencies are not critical — just that they are not harmonically related. Next, mount the assemblies back-to-back using 1/2" threaded standoffs as shown in Figure 4.

Wire the units together as shown in Figure 5. It is easiest to wire the controls and switches before mounting the oscillator assemblies into an enclosure. Interfacing to the oscillator assemblies is easily done through the screw-contact barrier strips that are part of the pc board assemblies. A 3.5 mm (1/8") stereo jack provides both the audio output and a PTT output. The PTT switch makes it easy to key your transceiver for testing.

For radio testing you will need to build a 3.5 mm stereo-to-transceiver microphone input plug adapter. My Icom IC706mkIIg uses an 8-pin RJ45 modular plug. Purchase a RJ45 pre-made cable from your local computer or electronics store. Cut off one end of this cable to a length appropriate for your test set-up, and strip and tin the wires that go to pins 4-7 of the RJ45 plug. It is easy to visually determine the correct wires by examining the RJ45 connector. The Icom IC703/706/7000 series 8-wire RJ45 microphone connections to a 3.5 mm plug are shown in Table 1.

Figure 6 shows an internal view of the assembly mounted in an aluminum box. Table 2 lists all the necessary components. I referenced Mouser (**mouser.com**) part numbers, but part substitutions from online auctions and surplus houses for the switches, connectors and enclosure can significantly reduce your costs.

Table 1
Icom IC703/706/7000 Microphone Plug Adapter wiring.

Signal	*8-wire RJ45 plug*	*3.5 mm plug*
PTT	Pin 4	Ring
GND	Pin 5/7	Sleeve
MIC	Pin 6	Tip

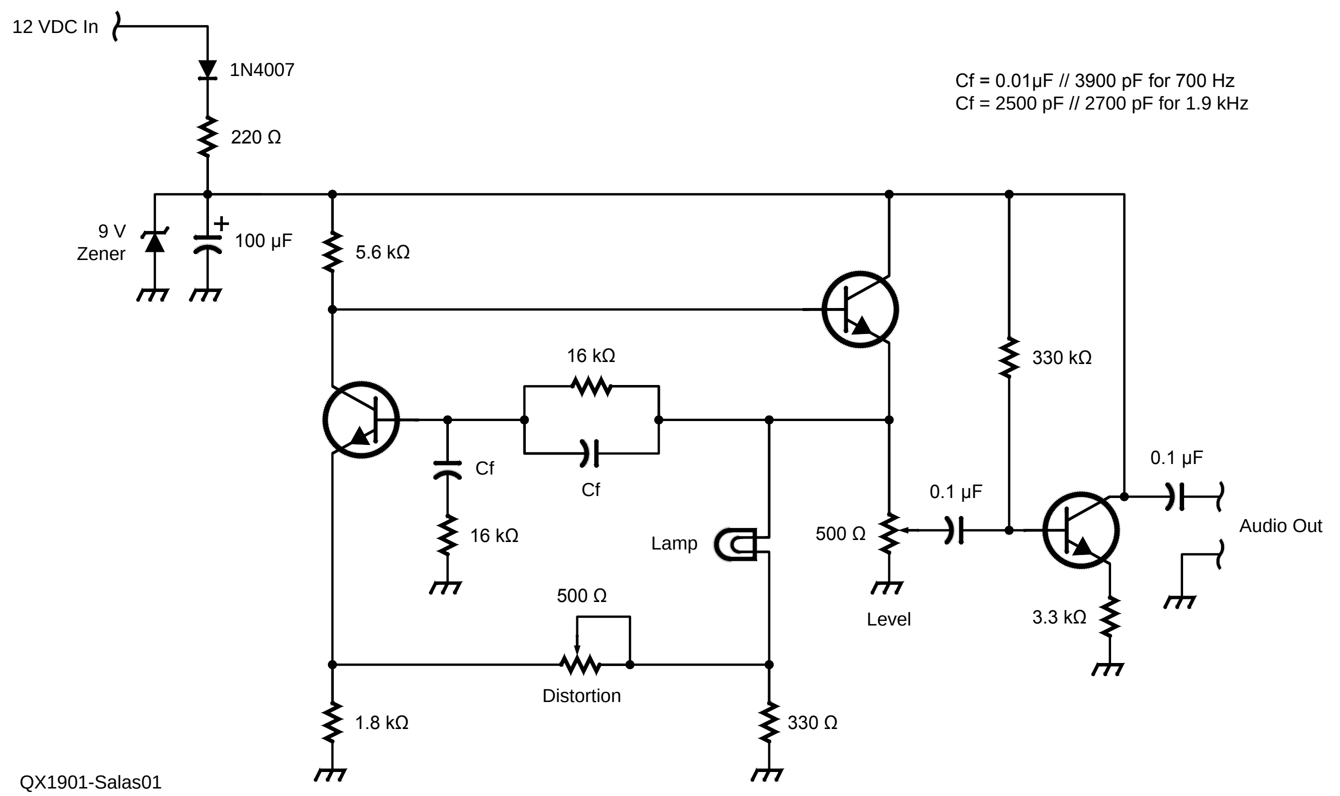


Figure 1 — Schematic diagram of a Wien bridge oscillator.

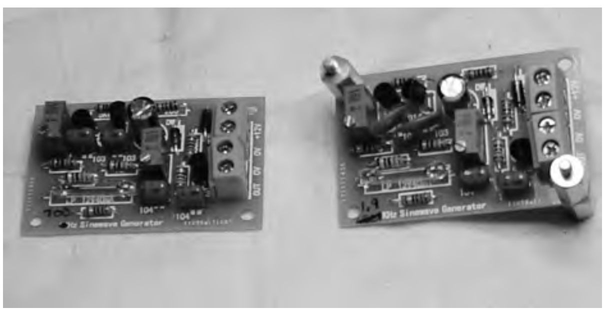


Figure 2 — Top side view of two assembled units.

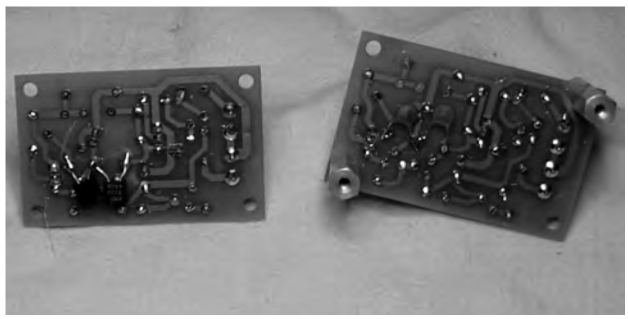


Figure 3 — Bottom side view of two assembled units.

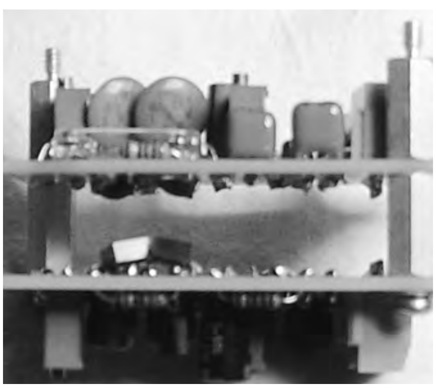


Figure 4 — Oscillators mounted back-to-back.

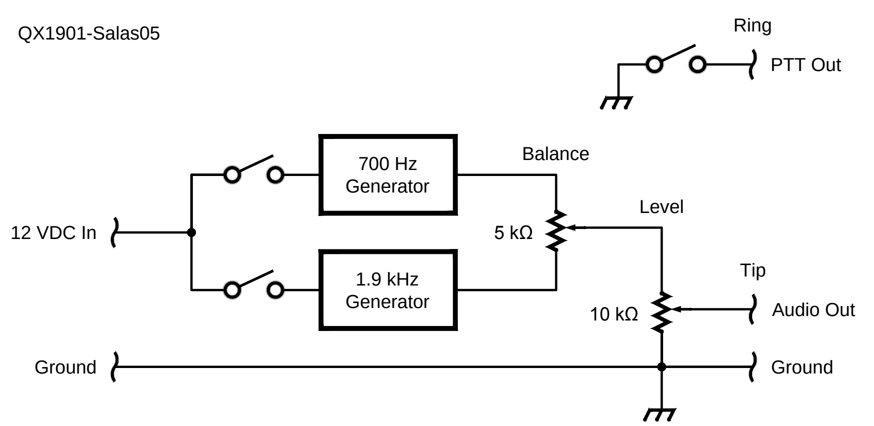


Figure 5 — Assembly interconnect diagram.

Figures 7, 8, 9 and 10 show external views of the completed unit. Casio black-on-clear labeling tape provides the labeling.

Set-up and Testing

Since you are going to be using this test generator for 2-tone transmitter testing, you probably have an oscilloscope with Fast Fourier Transform (FFT) capability, which will give you a spectrum display. Begin by terminating the 2-tone test output in 620 Ω — close to the 600 Ω typical transceiver microphone input impedance. Next set the balance control to mid-point and the main amplitude control to maximum. Then adjust each individual level control for 300 mV P-P output. This should be more than enough level to drive any modern transceiver to full power.

Next adjust the internal distortion controls for minimum harmonic content. Use your oscilloscope's auto-set capability, and then select FFT. Now adjust the distortion controls for minimum harmonic content. When finished, re-adjust the output levels if necessary. Figure 11 and 12 show the 700 Hz output in time and frequency. Figures 13, and 14 show the time and frequency domain displays of the 1,900 Hz output of my unit. The 2nd harmonic is more than 50 dB down for each tone.

Figure 6 — Internal view of the oscillators mounted in an aluminum box.

Figure 7 — Top view of the 2-tone generator.

Figure 8 — Bottom view of the 2-tone generator.

Figure 9 — A 3.5 mm output connector is on the side of the box.

Figure 10 — The dc input power connector is on the side of the box.

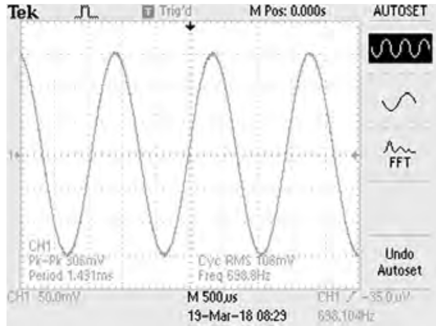


Figure 11 — Time domain display of the 700 Hz tone.

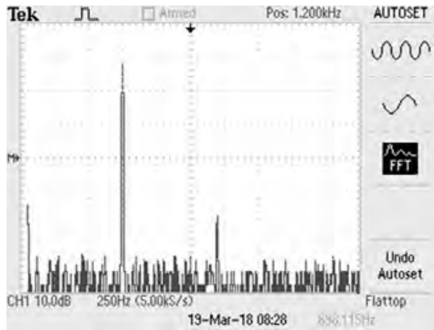


Figure 12 — Spectrum display of the 700 Hz tone.

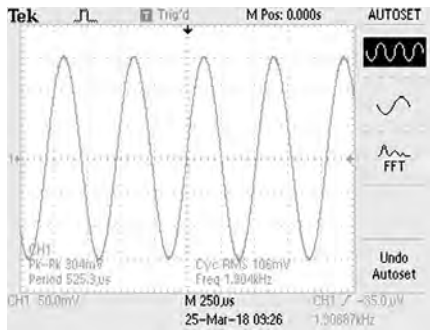


Figure 13 — Time domain display of the 1.9 kHz tone.

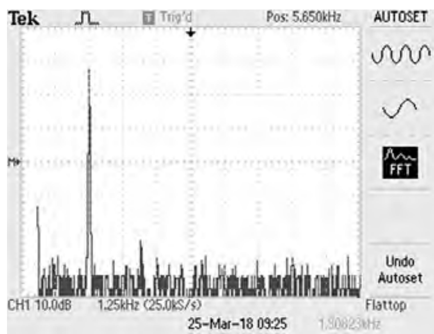


Figure 14 — Spectrum display of the 1.9 kHz tone.

Table 2.
2-Tone Test Oscillator Parts List

QTY	*Description*	*Source/PN*
2	Oscillators	Online auctions
1	Aluminum Box	Mouser 563-CU-2101B
1	Potentiometer, 5 kΩ	Mouser 31VA305-F3
1	Potentiometer, 10 kΩ	Mouser 31VA401-F3
1	Jack, 2.1 mm	Mouser 502-PC722A
3	Switch, Toggle	Mouser 633-M201201
4	1/2" #4 threaded standoff	Mouser 728-FC4505-440-A
2	2500 pF capacitor	Mouser 75-WYO252MCMBF0KR
2	2700 pF capacitor	Mouser 810-FA28C0G1H272JNU6
2	3900 pF capacitor	Mouser 810-FG28C0G1H392JNT6
2	4-40 nuts	—
2	4-40 x ¼" screws	—
4	#4 split-ring lock washers	—

Powering Option

This 2-tone generator draws just 30 ma when both oscillators are on, it lends itself to battery operation. If desired, a larger enclosure could be used to house a 9 V battery. In this case, you should short the 1N4007 diode and the 220 Ω resistor, and remove the 9 V Zener diode.

Conclusion

I've described a two-tone test generator with highly linear sinusoidal outputs for transceiver IMD testing. It is easy to build, and inexpensive (less than $20 with careful shopping), giving you another alternative to commercially available units.

ARRL Life Member Phil Salas, AD5X, has been licensed continuously since 1964. His interest in ham radio led him to pursue BSEE and MSEE degrees from Virginia Tech and Southern Methodist University respectively. He had a 35 year career in RF, microwave and light-wave design holding positions from Design Engineer to Vice President of Engineering. Phil is now fully retired and enjoys tinkering with ham radio projects and spending time with his two grandsons.

Notes

[1]Pacific Antenna, **www.qrpkits.com/twotonetest.html**.
[2]Elecraft 2T-Gen, **https://www.elecraft.com/collections/test-equipment/products/2t-gen-2tone-generator**.
[3]PreciseRF TTG1, **preciserf.com/shop/ttg1-two-tone-test-generator/**.
[4]**https://www.electronics-tutorials.ws/oscillator/wien_bridge.html**.

Baseband Quadrature Multi-Band Modulator (Continued from page 8)

Notes

1Linear Technology "LTC5598 5MHz to 1600MHz High Linearity Direct Quadrature Modulator", **cds.linear.com/docs/en/datasheet/5598f.pdf**.
[2]R. Lyons, "Understanding the 'Phasing Method' of Single Sideband Demodulation", **https://www.dsprelated.com/showarticle/176.php**.
[3]A. Wolke, W2AEW, "Basics of IQ Signals and IQ modulation & demodulation – A tutorial", **https://www.youtube.com/watch?v=h_7d-m1ehoY**.
[4]Note 1 p. 9.
[5]Note 1 p. 12.
[6]Texas Instruments Filter Design Tool, **www.ti.com/design-tools/signal-chain-design/webench-filters.html**.
[7]Linear Technology, "LTC1983 – 100 mA Regulated Charge-Pump Inverters in ThinSOT," p. 10, **cds.linear.com/docs/en/datasheet/1983fc.pdf**.
[8]Linear Technology, "Quick Start Guide For Demonstration Circuit 1455A 5MHz To 1600MHz High Linearity Direct Quadrature Modulator, " p. 7, **cds.linear.com/docs/en/demo-board-manual/dc1455A.pdf**.
[9]PCB GOGO, **www.pcbgogo.com**.
[10]Bob Allison, WB1GCM, *Technical Correspondence*, "ARRL Laboratory Handheld Transceiver Testing", *QST*, Nov. 2015, pp 74-76.

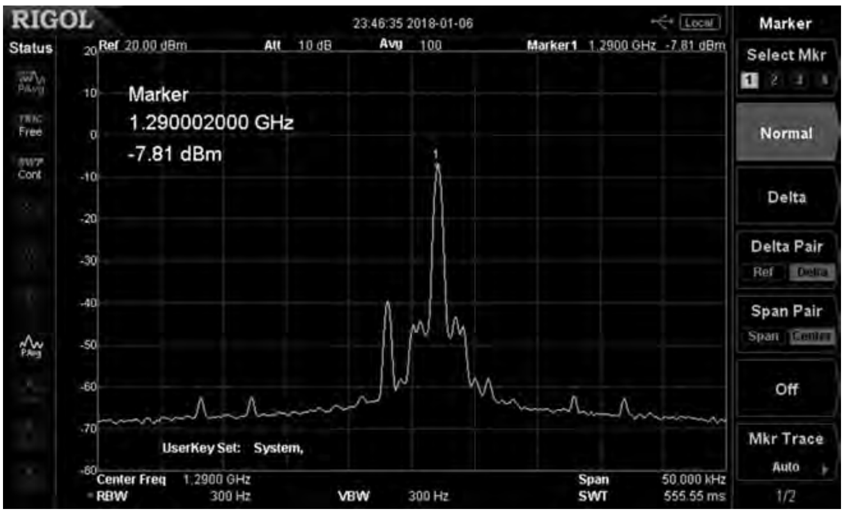


Figure 10 — Spectrum analyzer plot of USB 2000 Hz tone at 1290 MHz using a Wavetek 3520 for local oscillator, span is 50 kHz.

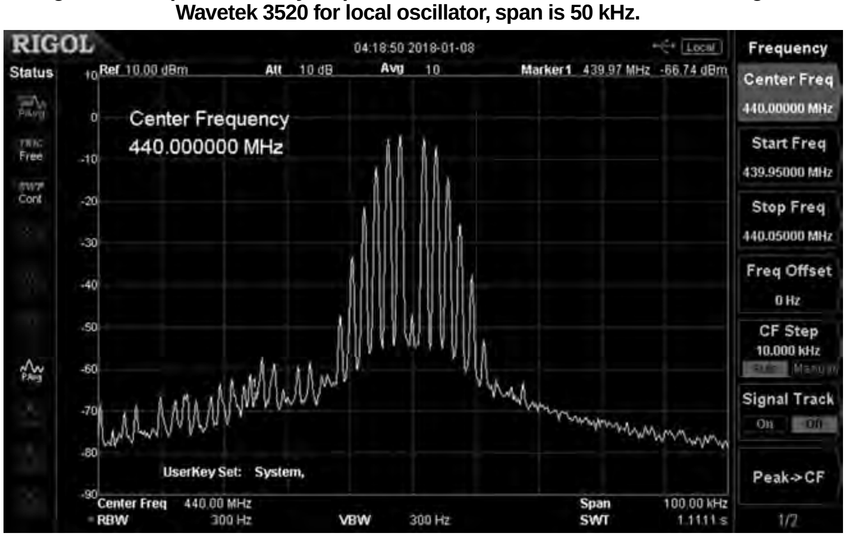


Figure 11 — Spectrum analyzer plot of FM 1.9 kHz tone with a modulation index of 5.4 at 440 MHz.

Jim Koehler, VE5FP

2258 June Rd., Courtenay, BC V9J 1X9, Canada; **jark@shaw.ca**

Reflow Soldering for the Radio Amateur—Revisited

Automate a simple toaster oven for reflow soldering.

In a previous article[1] I described how a simple toaster oven could be used for reflow-soldering by radio amateurs. There have been a number of similar descriptions on the internet but, in my opinion, the secret of doing it successfully is to monitor the oven temperature. In the previous article, I described how to do this using a thermocouple-based electronic thermometer. It worked nicely but it quickly becomes tedious with manually turning the oven on and off to get the correct temperature profile. In this article, I describe how to automate the process so that it is necessary only to push a button to start and a simple controller does the rest.

The simple controller I used is the Raspberry Pi Zero W (R-Pi). This wonderful little computer costs only $10 yet it includes a Wi-Fi interface so that program development is very easy and the controlling program can be written in a high-level language. It is a gross example of over-kill to use such a powerful computer to control something as simple as an oven but it is hard to beat the price and the ease of development. Total parts cost will depend on what you have in your junk box but, even in the worst case, can't cost you more than about $100. For that price, you get a reliable reflow-solder oven, which can be controlled by just three push-button switches.

Description

For this project you will need a toaster oven, a solid-state switch capable of handling 20 A, an R-Pi, a small dc motor with a fan blade, a tiny temperature sensor, a temperature sensor interface, and a few LEDs and push-button switches. Suitable toaster ovens can often be found for under $25 new, or less for used ones at garage sales. Adafruit[2] sells the temperature sensor interface ($15) as well as the R-Pi ($10). The rest of the small parts and a dc power supply can be bought from Digikey or Mouser, or may be found in your junk box. For the dc power, I use an old laptop computer power supply. These are ubiquitous and supply around 18 V at up to 2 A, but any dc supply of 12 V or so at 1 A is sufficient. The dc supply goes to two TO-220 linear regulators that power the dc motor and the R-Pi.

The small dc motor is mounted on the back of the oven with the fan blades inside. It is not meant to create a strong wind inside the oven but rather to just stir the air so as to keep the temperature more or less constant across the volume of the oven. I found a 9 V dc motor with a fan blade already on it in my junk box. Similar small motors can be found in surplus stores or on the internet.

I used a Tyco SSRT-240D25 solid-state switch. It is rated to control up to 240 V ac at 25 A. Since my toaster oven takes only about 1 kW from 120 V ac, this switch is operated conservatively. Similar switches can be found on eBay for around $10 postpaid.

The circuit diagram is shown in Figures 1 and 2. Part of the circuit (Figure 2) is built on a pc-board[3], a "Pi-hat" that plugs onto the R-Pi. The rest (Figure 1) is just wiring on a chassis. The chassis (Figure 3) is constructed partly from sheet aluminum and partly from 0.1" thick ABS sheet. It is not totally metal because the R-Pi communicates with your local area network (LAN) via Wi-Fi, so it should not be inside a shielded enclosure. The metal part holds the ac receptacle that your oven plugs into as well as the dc power input to the circuitry.

After you have constructed the instrument, you must connect to your LAN in order to control the R-Pi when the oven is being calibrated. You will not need to have a separate keyboard, mouse and monitor to operate the R-Pi; it can all be done through your LAN from your computer. After calibration, the network connection is no longer required and you can control your oven with just the push-button switches mounted on the instrument. So, you can use the oven in your shop even if you do not have Wi-Fi or a computer there.

A very good description of the reflow-solder process and a discussion of the various parameters can be found in a paper by R.C. Lasky[4]. For the toaster oven running at 1 kW, a solder profile similar to Lansky's Figure 1 is easily achieved. On my oven I was able to get a temperature rate of increase of about 1º C per second. This is about the middle of Lansky's recommended heating range. The desired temperature profile is to steadily increase temperature until the solder melts and flows, hold at this temperature for 10 or 20 seconds and then cool. To do this, the controller turns the oven on and when the temperature reaches the desired peak temperature, the oven is switched off and an LED is lit to alert the operator to open the oven door to allow the pc-board to cool. This sounds too easy, and it is! When the oven is switched off, the temperature will continue to 'coast' upward and the actual peak temperature might be greater than what is desired. To compensate for this, the oven must be calibrated so that the controller switches the oven off before reaching the desired peak temperature. From that point

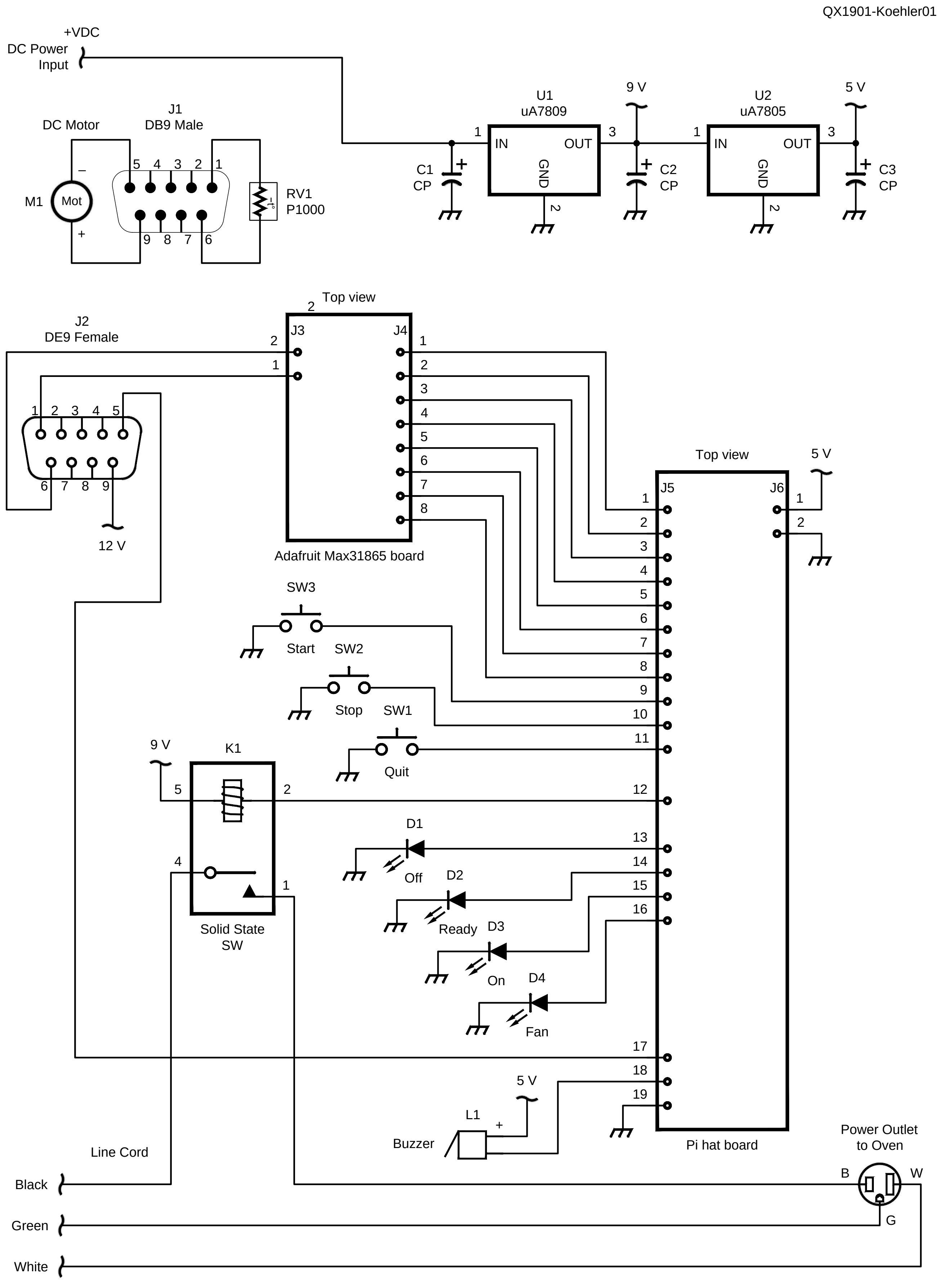


Figure 1 — Schematic portion of the point to point wiring on the chassis.

it will just coast to the desired peak. I have written a software routine — described later — to do this calibration very simply. The desired peak temperature will depend on the type of solder you use. The commonly used is 60/40 solder (60% lead, 40% tin) has a melting point of 183° C and a 'liquidus temperature'[5] — the temperature at which it flows freely — of 188° C. We want to be sure that the pc-board temperature reaches at least this temperature, so normally the oven is set to raise the air temperature to something greater than this value; say 210 to 215° C, and to hold it at this temperature for 10 or 20 seconds or so. Most integrated circuits can easily withstand these temperatures for tens of seconds. Indeed, in Europe, lead-bearing solder is no longer allowed in production electronics, so modern integrated circuits must tolerate much higher temperatures than what non-leaded solders require.

Construction

The metal bottom and back plate were

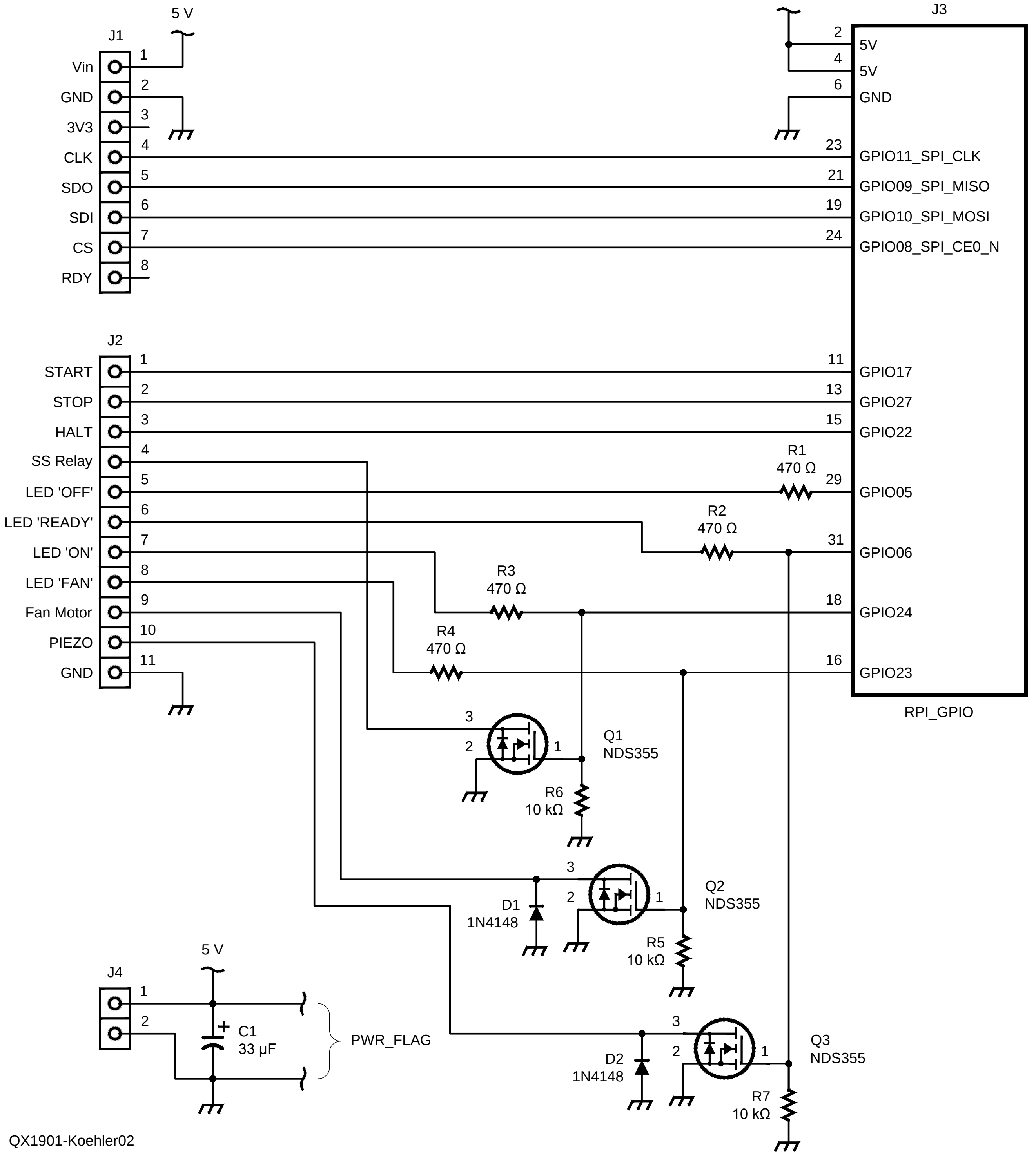


Figure 2 — Schematic portion that is built on a pc-board.

folded from a single sheet of aluminum (Figure 3). The 120 V ac receptacle that the oven plugs into is mounted on this back surface. Also mounted there is the dc jack compatible with the dc power supply, in my case the laptop power supply. Two wires go to the dc motor and another two wires go to the temperature sensor mounted in the oven. These four wire connections from the controller to the outside were made using four pins of a DE-9 connector. The male connector is on the four conductors going to the motor and the sensor, and the female connector is on the chassis. Also on the back plate of the chassis is a 120 V ac input cable that goes to the oven receptacle via the solid-state switch. I cut a piece of 0.1” ABS sheet to cover the bottom of the chassis and to make the front panel, and used more 0.1” sheet ABS to make an outside cover. ABS is nice to work with and it is easy to glue together. You may use either Krazy Glue® or ABS cement. The ABS sheet I had was a bilious yellow so I spray-painted it with a less garish pale yellow color and used water-slip decals to make the labels. The LEDs used in the project were just press-fitted into holes drilled in the front panel.

I had previously used thermocouple sensors and their interfaces for sensing temperature but for this project I decided to use a platinum temperature sensor and the MAX31865 as the interface. PT1000 platinum temperature sensors use a thin film resistor made from a platinum alloy that has a resistance at 0° C of exactly 1000 Ω. The resistance change with temperature is very well known and the MAX31865 uses a ratio-metric method that results in remarkably stable, accurate and well defined temperatures. In general, the accuracy surpasses that of thermocouples. Inside the oven, we want the sensor temperature to be very close to that of the air surrounding the target pc-board. This means that the sensor should have very little thermal mass. I chose US Sensor PPG102C1 (DigiKey part number 615-1045-ND). This is a small chip measuring about 0.08” by 0.12”. It must be mounted inside the oven so that it is close to the pc-board being soldered, and it needs to be insulated electrically from the oven. I used an approximately 6” piece of 1/4” Teflon™ rod that I drilled with a 1/8” bit through its length. I used a ¼-28 die to thread one end of the Teflon rod. Figure 4 shows the rod (top image), the sensor soldered to a twisted pair of Teflon-insulated stranded wire with the solder joints covered with small heat-shrink tubing (middle image), and (bottom image) the sensor pulled into the end of the rod.

The Teflon rod acts as a heat insulator so the inside solder joints to the sensor should not actually melt when the air in the oven reaches solder melting temperatures. The threaded end of the rod is used to fasten the rod to the back surface of the oven using a ¼” hole. This hole should be placed so that the sensor end of the rod is only about ¼” to ½” above the grill where the pc-board will be placed.

The small dc motor is mounted on the outside back surface of the oven using long hex spacers. The back surface of the oven gets very hot and the spacers somewhat insulate the motor from it. The motor was mounted so that the fan blade was just an inch or so away from the inside back surface of the oven. Figurer 5 shows the motor mounted on the back surface of the oven (upper image), an inside view showing the fan and the temperature sensor (middle image), and an outside back view of the assembly (lower image) showing the motor and sensor mounting.

I had replaced my badly rusted oven grill with a thin aluminum plate used as a shelf. I made a small table from wire mesh that sits on this shelf. The pc-board to be soldered is put on this little mesh table, that way both the top and bottom surfaces of the pc-board are exposed to the heated air inside the oven. If your grill is clean, you can dispense with the aluminum shelf and the mesh table, and just place your pc-board on the grill.

The ‘Pi-hat’ pc-board is assembled using 0805-sized surface-mount passive components. This size is not too difficult to solder by hand. There are also three SOT-23 sized HEXFET transistors. For those who

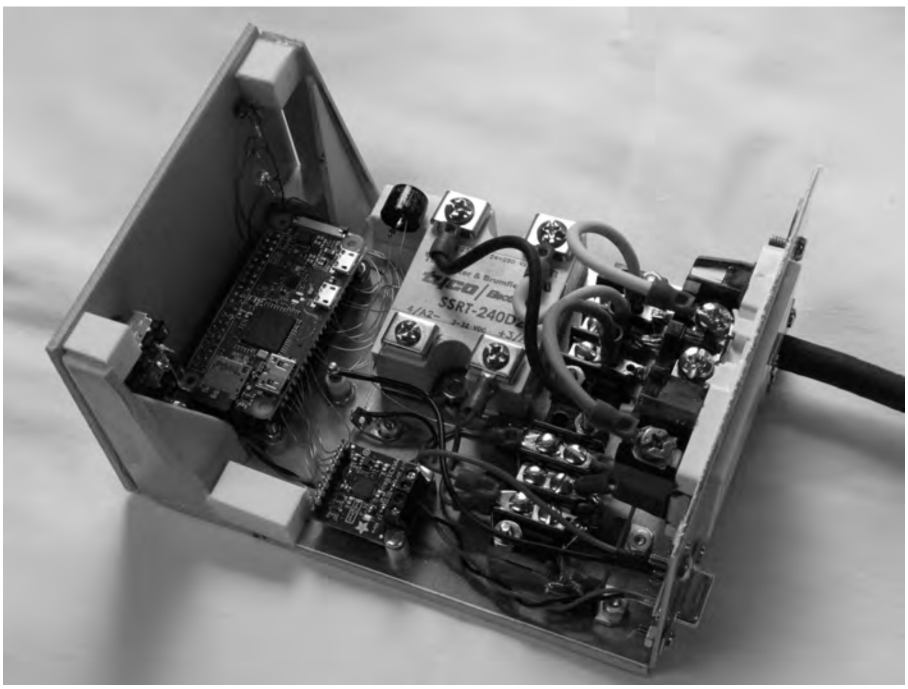


Figure 3 — The chassis is partially constructed from sheet aluminum.

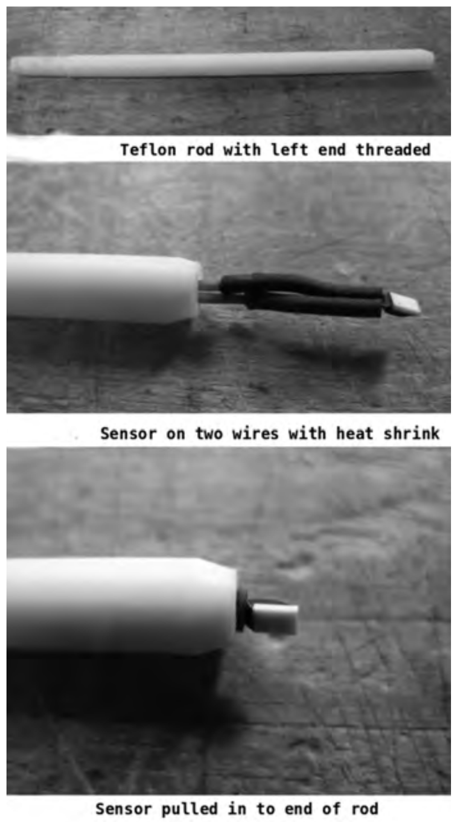


Figure 4 — Teflon rod (top image), the sensor soldered to a twisted pair of Teflon-insulated stranded wire with the solder joints covered with small heat-shrink tubing (middle image), and (bottom image) the sensor pulled into the end of the rod.

Figure 5 — The motor mounted on the back surface of the oven (top image), an inside view showing the fan and the temperature sensor (middle image) and an outside back view of the assembly (bottom image) showing the motor and sensor mounting.

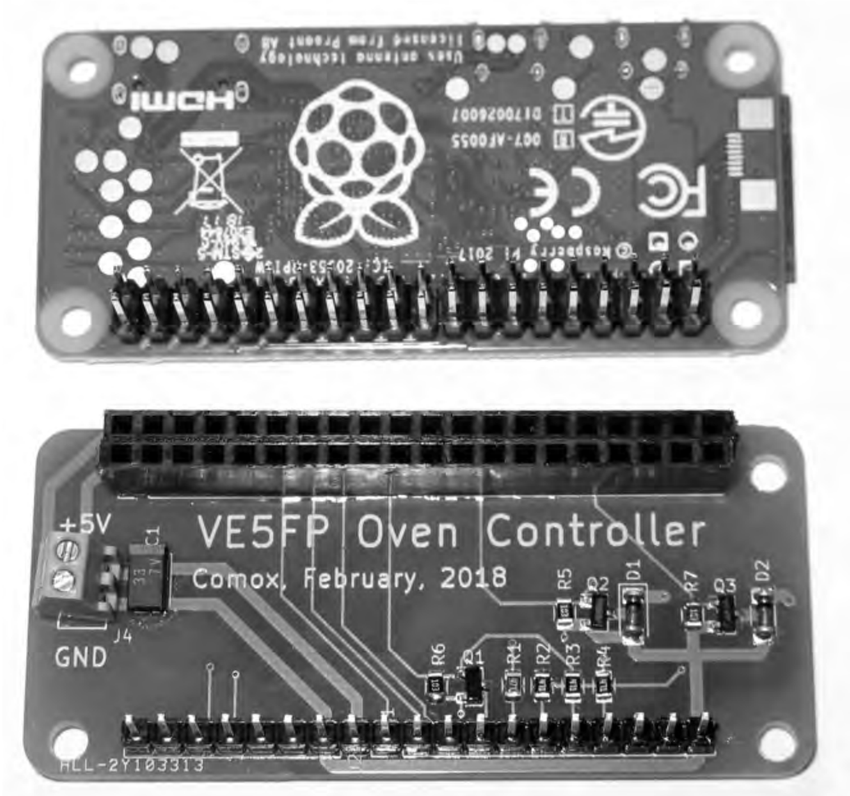


Figure 6 — The Pi-hat mates to the R-Pi via the female 40-pin header on top of the Pi-hat and the male 40-pin header mounted on the bottom of the R-Pi.

prefer not to work with SMD components, the assembled board is available, see Note 3. To mate the Pi-hat to the R-Pi, I have put the female 40-pin header on top of the Pi-hat and the male 40-pin header is mounted on the bottom of the R-Pi (Figure 6). This way the R-Pi sits on top with its Wi-Fi antenna clear of obstructions. The normal way of mounting would have obscured the antenna of the R-Pi. The Pi-hat on the bottom has two single-row male headers on it for connection to the rest of the circuit.

Referring to Figures 2 and 6, the eight connections of J1 connect to the Adafruit MAX31865 board. For J1, pin 1 is on the right side of the left-most 8 pins. The right-most 11 pins, J2 in Figures 2 and 6 connect to the LEDs, switches, the motor and optionally a buzzer. J2 pin 1 is adjacent to J1 pin 1. I placed these two headers adjacent to one another so that a single un-cut 19-pin piece of male header can be used to make both headers J1 and J2. To interface between the Pi-hat and the Adafruit MAX31865 board you can use jumpers, solder wires or wire-wrap. Similarly, from J2 to the LEDs and switches the end going to J2 can be jumpered or soldered or wire-wrapped. I used wire-wrap for both. To prevent the pins of the header on the Pi-hat from contacting and shorting to conductors on the bottom side of the R-Pi, I glued a strip of plastic on the bottom of the R-Pi and I also nipped off the tops of the Pi-hat header pins. Instead of using straight male headers as I did, you could also use a row of right-angle male headers. That way the male pins would point away from the Pi-hat board instead of sticking up from it.

There is a pin on the Pi-hat to connect an optional buzzer. This buzzer is activated at the same time as the READY LED so it will sound when the reflow-solder process controlled by the R-Pi has finished.

Power to the oven comes into the enclosure via a strain relief. Be sure to use an appropriate size of line cord and the wires from the solid state relay to the ac outlet. They must handle about 10 A if your oven needs 1 kW. The oven is turned off and on via the solid-state relay. Please be very careful about this portion of the wiring!

Setting up the R-Pi

The R-Pi is normally used with a separate keyboard, mouse and monitor. However, it is also possible to use it ‘headless’. That means you can connect to it using your own computer with its keyboard mouse and monitor through your LAN using the R-Pi Wi-Fi capabilities. To do this, you must connect automatically to your LAN when it is turned on. The first step is to load the pre-configured Linux operating system onto a micro-SD card. It is available as a file you can download from the Raspberry web page[6]. The needed image file is *Raspbian Stretch* with *Desktop*. This image file is more than 4 GB in size. You then edit a few files on it, according to the instructions in an Adafruit tutorial[7] and then ‘burn’ this image file into a micro-SD card using a free program called *Etcher*[8]. There are free versions of *Etcher* available for Linux, Windows and Mac OS. The micro-SD card should be at least 8 GB in size and fairly fast. There is no maximum size limit. These days it is hard to find micro-SD cards smaller than 16 GB and the speed of these is sufficient for the R-Pi operating system. I used the most inexpensive 16 GB card I could find. To transfer the image file to the micro-SD card you must access the card from your computer. The easiest way to do this is to put the micro-SD in a USB adapter. Then, when you plug in this adapter your operating system will detect it. You can then burn the image onto the card using *Etcher*.

If you then plug the micro-SD card into the socket on the R-Pi and power it up, either with the R-Pi just by itself using a mini-USB power supply or by putting it into the circuit you have just built, you will see some activity on the little LED close to the mini-USB socket on the R-Pi. After 20 seconds or so, it will have booted up and connected to your LAN. Your router will assign the R-Pi an address on your LAN. But what is that address? There are programs to find out and I recommend the *Angry IP Scanner*[9] available for free for Windows, Mac and Linux operating systems. When you do a scan of your LAN, you will see the R-Pi appear with a URL. On my home LAN it appeared at **192.168.1.26**.

Use *SSH*[10] to connect to the R-Pi. *SSH* is a secure protocol for connecting from one machine to another on a network. If your main machine is Linux or a Mac, you will already have *SSH* on it and you can access it by opening a terminal. If you have Windows 10, then it is already part of the operating system but it is not enabled by default. You can find instructions on the internet to do so[11]. In the terminal, to connect to the R-Pi, you simply type: **ssh pi@192.168.1.26** where you replace my address with the address that you got from the *Angry IP Scanner*. You will have to answer a question and then you will be connected to the login screen of the R-Pi asking for a password. The password is ‘raspberry’. After logging in, you will be the user named ‘pi’ sitting at a terminal with full control of the R-Pi.

To power down the R-Pi — don’t do it just now — just type the command: **sudo halt** and the R-Pi will disconnect itself from the network and power down. You can tell when it is done as the little green LED will go out.

We can also access the R-Pi by using a protocol called Virtual Network Computing (VNC). This is all described in another Adafruit tutorial[12]. These instructions tell you how to install a VNC server in the R-Pi, how to install the VNC client in your computer and how to set up the R-Pi so that the VNC server starts up automatically on boot-up.

From now on, you can use the VNC client installed on your computer to log in to the R-Pi and you will then have a GUI interface to the R-Pi. In the material that follows, I assume that you are interacting with your R-Pi using VNC. The R-Pi appears on your computer screen as a window as shown in Figure 7. This is a generic GUI interface. If you left-click on an icon, it does the default thing and if you right-click on an icon, you get a list of possibilities. To start with, if you left-click on the third-from-left icon on the top row, you are invoking the file displayer and it will open a window and show you the contents of the home directory on the R-Pi. If you then close the file displayer window and left-click on the fourth-from-left icon on the top row, you will invoke the terminal and if you now type ‘**ls**’ on the terminal line, you should see something like the listing shown in Figure 7. Notice that the response is not anything like that of your main computer. I recommend that you change the host-name of the R-Pi and the default password especially if you operate the oven within range of your LAN.

Finally, we must install some software that is used in the various oven programs. Python, a high-level language is already installed. We need to also install *gnuplot*, a program for plotting data and *feh*, a program for displaying graphics files. We install these from the internet by first invoking a terminal and then typing,

Figure 7 — The R-Pi on the computer screen is a generic GUI interface.

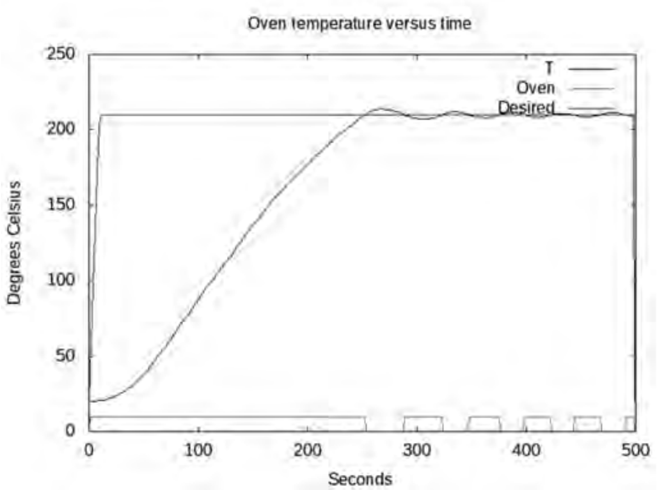


Figure 8 — Temperature variation vs. time with Kd equal to zero.

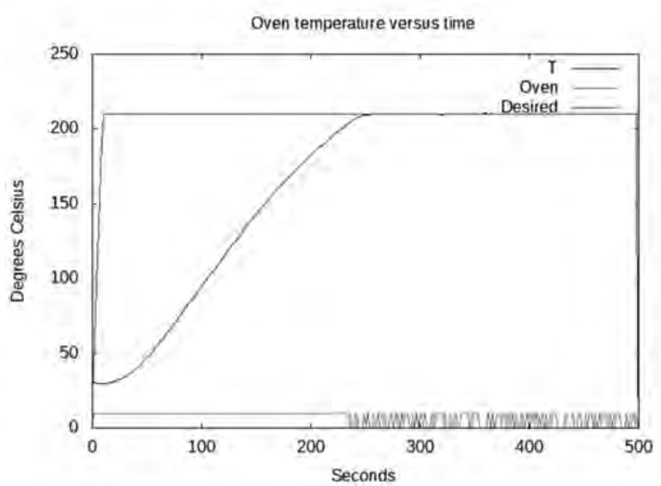


Figure 9 — Temperature variation vs. time with Kd set -9.

sudo apt-get install gnuplot feh

We also need the software I've written to handle the oven. This is available in the file, '*pi_oven.zip*', from the **www.arrl.org/QEXfiles** web page. You can access using the R-Pi web browser and download it into the R-Pi. It will appear in the R-Pi download directory. Using the R-Pi file handler, extract all the files in it into the R-Pi **/home/pi** directory. The R-Pi is now completely set up.

Setting Up the Oven

The oven operates as a negative feedback system. The controller determines the error between the desired temperature and the actual temperature and switches the oven on and off so as to minimize the error. The particular algorithm is called the *proportional integral and differential* (PID) method. In the case of this oven controller, because the entire solder-reflow process takes just a few minutes, we dispense with the integral of the error so, properly, this algorithm is the PD method. The temperature error *e* is the desired temperature *Td* minus the actual temperature *Ta* at any time,

$$e = Td - Ta$$

so the value *s* used to turn on the oven is,

$$s = Kp \cdot e + Kd \cdot \frac{de}{dt}$$

If *s* is greater than zero, the oven is turned on. If *s* is zero or less, the oven is turned off. Here *Kp* is a constant of proportionality and *Kd* is the differential constant. The differential *de/dt* is the rate of change of error per second. We can arbitrarily assign a value of 1 to *Kp* and make some measurements to determine the value to use for *Kd*. Its value will depend on your particular oven.

In the program used to control the oven the value of *Kd* is read from a file named "k_d.txt". This file consists of one line holding one number. Click on the file from within the VNC window to open it with a text editor and you will see that is just the number -9.0. This is the value that works for my oven. I have written a test program, called "test_oven" that you can use to determine the appropriate value for your oven. Using a text editor, open "k_d.txt" and replace the number with zero (0.0). Then, with the oven connected, run the test_oven program by double-clicking its icon in the VNC window and then clicking on "Run in terminal". This will run the oven with a desired temperature of 210° C for five minutes. When the program has completed, you will see a graph displayed in the VNC window showing how the temperature varied with time. Figure 8 shows what I saw with *Kd* equal to zero. The "Desired" is the flat trace at 210°, the "Oven" is the trace at the bottom, and the "T" is the sloping trace of oven temperature. The temperature overshot the desired value by about 5°, then oscillated back and forth around the target temperature. Change the value in the k_d.txt file by trial and error so that the temperature comes up to the desired temperature and just overshoots a fraction of a degree. *Kd* will be some negative number. If it is too large a negative number, the temperature will approach the desired temperature too slowly. If it is too small, it will overshoot too much. Figure 9 shows the graph when the value in k_d.txt is -9.0, the value that works best for my oven.

The oven is ready to go once the correct value of *Kd* has been determined. I have written a Python program *oven.py* that does the job of turning on the oven and going to an appropriate temperature and then turning the oven off. The program is designed to have the oven follow a temperature 'profile' that is just a file containing a time sequence, one second at a time, of what the desired temperature is. Since the whole profile might last over many minutes, this file will contain many lines of data with one new line every second. It would be tedious in the extreme to have to write out this file for every new situation. I wrote a second program that takes an abbreviated profile which just has an entry every time the desired temperature changes and produces the final temperature profile that the *oven.py* program requires. My abbreviated short profile file for eutectic or 60/40 solder contains the lines in Table 1.

At time zero, the desired temperature is 20° C, at 1 second, the desired temperature is 210° C, this is desired until time 301 seconds when it drops to 20 again and finally, the whole process ends at 360 seconds. This is the normal file that I use for normal boards using 60/40 solder. I wrote a Linux script file, named *oven* that first invokes the program to read the short eutectic solder file and write the desired profile file and then invokes *oven.py* to follow that profile.

Doing the Reflow-Soldering

Once you have gotten the best value for *Kd* and written that value into the k_d.txt file, the oven is ready for use. After you have prepared the pc-board by applying the solder paste and placed the components, you just need to put it in the oven and then run the program. Double-click on the "oven" icon, select "Run from terminal" and from then on the process will be controlled by the three push-buttons. Start (see Figure 10) begins the solder-reflow process. The oven will come up to 210° C. Then the Ready LED will start to flash and the oven will turn off. At this time, you should open the oven door to allow the pc-board to cool. If you built in the optional buzzer, it will sound simultaneously with the Ready LED. That

Table 1.
Abbreviated short profile file for eutectic or 60/40 solder.

0	20
1	210
300	210
301	20
360	20

Figure 10 — Front panel of the Pi-Oven controller. Push Start to begin the solder-reflow process.

is all that there is to it. The Stop button will stop the process at any time that you press it. The Quit button is used when you are done for the day. It makes the R-Pi go through its shut-down process. Note that you must Start the process first for the Quit button to be active. When the program has gone through its sequence, a graph of temperature *vs*. time will be displayed in the VNC window.

Although it is nice to see the graph of temperature *vs*. time for each solder process, it is not very convenient to have to control the oven using a computer. It is very simple to have the oven program run every time that the R-Pi boots up. That way, you can still control it using the three push-button switches and you do not need an internet connection. To make the program run by default upon boot up, you just edit a file in the home directory called **.bashrc**. Notice the period before the name: it is important. From a terminal we can use the simple text editor built into the R-P nano and open the file for editing by typing,

sudo nano .bashrc

This brings up a window showing the existing contents of the file. Simply scroll down to the bottom and add the following line followed by an *Enter*,

./oven&

Then save the file by typing *control-x*, and answering yes. Now when the R-Pi is rebooted it will automatically run the oven program. This means that you can control it using just the push-button switches.

Figure 11 shows the oven next to the controller. When you're done using the oven program, it is considered good practice to shut down the R-Pi gracefully by pressing first the Start button and then when the oven comes on, hold the Quit button for a few seconds. Give it ten seconds or so to shut down and then you can unplug the power to the unit.

Figure 11 — The oven shown next to the controller.

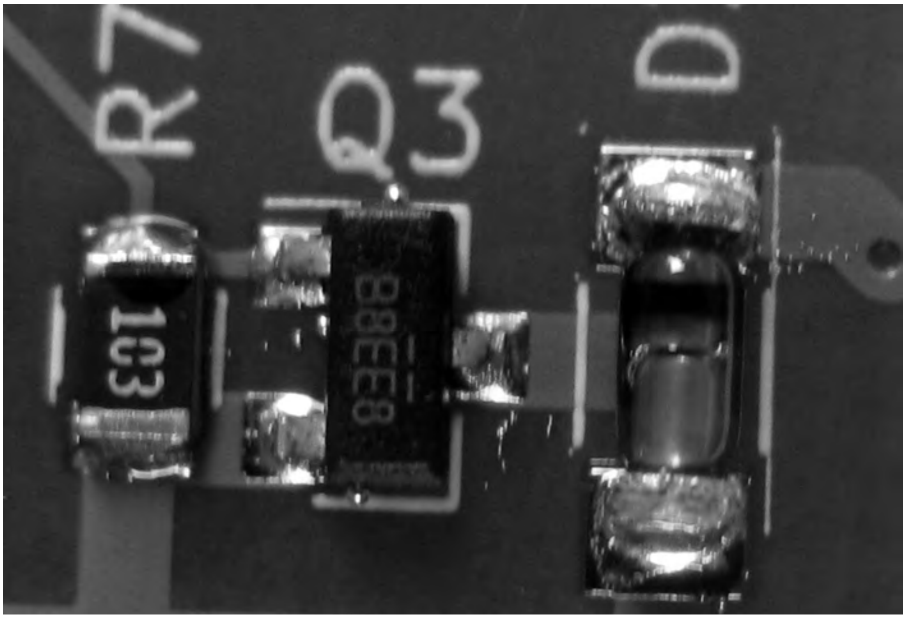


Figure 12 — Portion of a solder re-flowed pc board.

Conclusion

The modified toaster oven does an excellent job of reflow-soldering. Preparing boards for reflow-soldering is relatively easy and it does not matter too much if every pad does not have a precisely correct application of solder paste. Figure 12 shows a portion of the Pi-hat board, used in this project, and with the solder paste applied by hand, as it comes out of the oven. The solder joints are bright and clean. I normally wash a completed board in isopropanol using a tooth-brush to gently brush the components to dissolve and remove any solder flux. Do not use common rubbing alcohol which is isopropanol diluted with water.

Jim Koehler, VE5FP, earned his first ham license at age 15 in 1952. He completed undergraduate university, then was awarded a post-graduate degree in Australia. He returned to Canada in 1966, and became Professor of Physics and Engineering Physics at the University of Saskatchewan, and did research in upper-atmospheric physics. After retiring in 1996, he and his wife moved to Vancouver Island to enjoy Canada's best climate. Jim shares a hobby interest in electronic design with friend Tom Alldread, VA7TA. Both have extensive "junk boxes" and so are able to scrounge parts from one another freely, and cooperate with each other in taking advantage of all the new integrated circuits, which make life so much easier than when they were much younger.

Notes

[1]Jim Koehler, VE5FP, "Reflow Soldering for the Radio Amateur", *QST*, Jan., 2011, pp 32-35.

[2]**https://www.adafruit.com/product/3648**.

[3]The "Pi-hat" bare board is available from Jim Koehler, email **jark@shaw.ca**, for $15 postpaid. The board with all the surface-mount components already assembled on it is available for $25 postpaid.

[4]**https://kicthermal.com/wp-content/uploads/2012/11/Best-Practices-Reflow-Profiling-98675.pdf**.

[5]**www.farnell.com/datasheets/315929.pdf**.

[6]**https://www.raspberrypi.org/downloads/raspbian/**.

[7]**https://learn.adafruit.com/raspberry-pi-zero-creation/overview**.

[8]**https://etcher.io/**.

[9]**angryip.org/**.

[10]**https://en.wikipedia.org/wiki/Secure_Shell**.

[11]**https://www.howtogeek.com/336775/how-to-enable-and-use-windows-10s-built-in-ssh-commands/**.

[12]**https://learn.adafruit.com/adafruit-raspberry-pi-lesson-7-remote-control-with-vnc/overview**.

Jan M. M. Simons, PAØSIM

Prins Mauritsstraat 14, 5923 AZ Venio, The Netherlands; **pa0sim@veron.nl**

Finding Signals in the Noise Using Two Antennas

Use noise cancelling and noise reduction techniques to extract signals from noise.

Man-made noise, and hence, the noise floor on the HF bands in urban areas has significantly increased over the years[1]. Especially in the evening, local man-made noise (QRM) levels can be very high. When living in such an urban area and having limited space for antennas, noise cancelling can be used to attenuate a single identifiable man-made noise source. The remaining noise floor is the sum of multiple man-made noise sources and natural noise. Noise reduction can be used to mitigate this noise floor. The presented approach uses the signals of two small orthogonal antennas to find the signals in the noise. It incorporates one mouse-click cancelling and measuring propagation polarization behavior over frequency. It is a new approach to the one published by Simons[2].

Two Orthogonal Antennas

Two orthogonal antennas, like two identical small active receiving loops (Figure 1), provide two different but co-located antenna signals. Noise arrives from all directions with varying power and polarization. The resulting signal is random in time. The phase difference and amplitude ratio between the antenna signals are also random in time. Even two man-made noise sources with about equal signal strength can result in random behavior. This makes the noise floor random and renders a noise canceller less effective. Of course a single strong man-made noise source can be present. Then noise cancelling is the first line of attack.

Received sky-wave signals like SSB and CW however will only drift slowly in time in direction of arrival and in polarization. The signals of both antennas will be highly correlated and the phase difference and amplitude ratios will be almost constant over a period of time. When receiving circular polarized signals the phase difference is ±90° and the amplitudes are equal all over time.

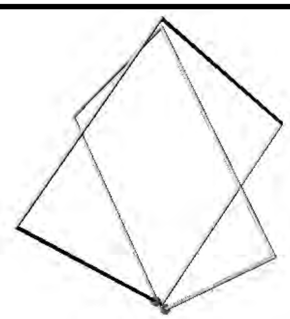


Figure 1 — Two orthogonal small active receiving loop antennas, each 1.25 m by 1.25 m.

Summary of the New Noise Reduction Approach

The processing is performed in the frequency domain at audio frequencies and is based on a three-dimensional space for estimating whether it is a signal or noise. The amplitude ratio and phase difference can be calculated for each frequency in the audio spectrum of the two antenna signals *A* and *B* that arrive from the two orthogonal antennas, and used as a pointer to a location in a two-dimensional (2D) space. The 2D spaces of all frequencies together make a single three-dimensional (3D) space. The amplitude ratio and phase difference, representing the polarization, and frequency are the coordinates of this 3D space.

When just noise is present the locations are uncorrelated and randomly spread over time. The locations of a signal with noise however will be concentrated in that space

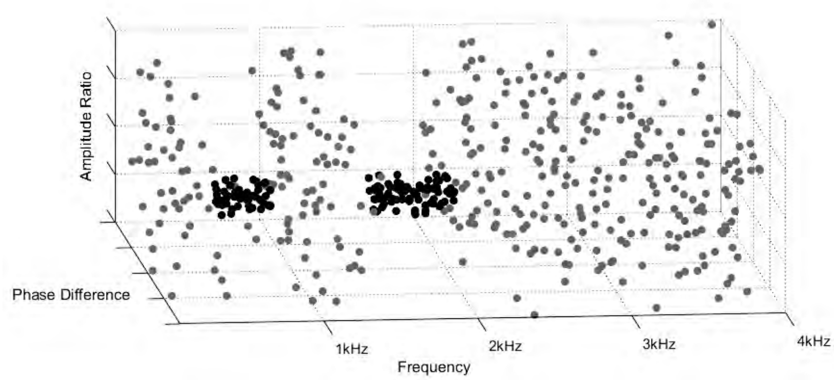


Figure 2 — Possible locations of a SSB signal power with noise over a short period of time in the 3D space.

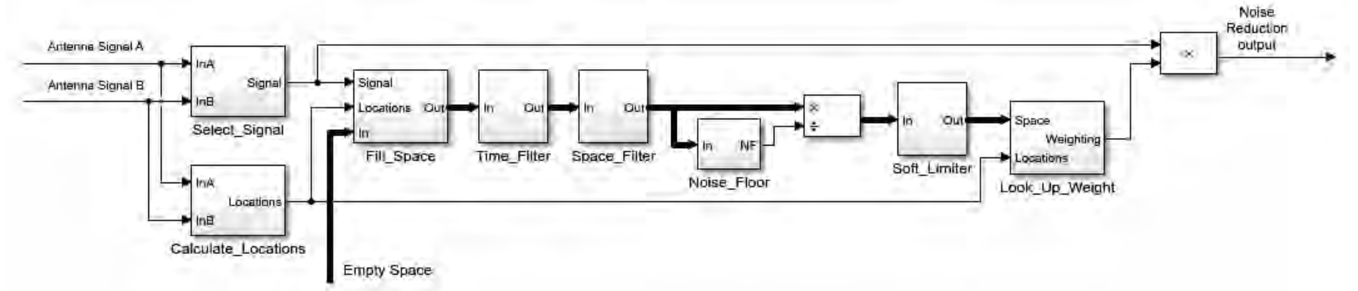


Figure 3 — Noise reduction processing using a 3D space.

over time. They not only will be concentrated in the 2D space of a single frequency, but also over a certain frequency span. Because of modulation the frequency components in a signal cover a bandwidth. In Figure 2 signal components of an SSB signal are present around 600 Hz and 1.7 kHz. They are concentrated on each frequency and over a limited frequency span. The higher power level of the signal components is indicated by a darker dot.

The thick lines in Figure 3 indicate the 3D spaces in the processing. An empty 3D space is filled on the corresponding locations with the power of the frequency components of the selected signal. The noise reduction is based on the assumption that the concentration of power in the 3D space over time represents an estimate of the location of the signal in that space. The concentration of power over time is calculated by performing in sequence a low-pass filtering in time and a 3D low pass filtering in space. The result is a low pass filtered space.

The noise floor is a sample of the low-pass filtered space when only noise is present. The power level in the noise floor is a measure for the expected signal levels. It is needed to normalize the low-pass filtered space. By scaling it to the maximum power in the noise floor the space can be weighted using a *sigmoid* function. The sigmoid function soft limits the space. The soft-limited concentration of power in the 3D space over time now not only represents an estimate of the location of the signal in space, but also whether it is a signal or noise.

The selected antenna signal is then weighted (filtered) by using the actual location as a pointer to the weighting factor in this soft-limited low-pass filtered 3D space.

Processing in Frequency Domain

The audio output signals from two phase coherent receivers are processed with 8 kHz sample frequency. The processing is done in the frequency domain using a 512 sample long FFT/IFFT and 8 times overlap with a 64 ms long Hann window. Each frequency bin is 15.625 Hz. With the Hann window this results in a 22.5 Hz bandwidth (-3 dB). Equivalent noise bandwidth (ENBW) is about 23.5 Hz. In order to have enough measurements in time of the spectra (125 per second) at least 8 times overlap is needed. The Hann window is applied also after the IFFT, because rather extreme filtering is possible[3].

Both antenna signals *A* and *B* are real signals in time. The FFT provides for each antenna 512 frequency bins with complex numbers. The output of the FFT however is conjugate symmetric and only 256 of the 512 bins need to be processed. The bin containing the dc component is skipped.

Spatial Information

Each pair of signal *A* and signal *B* bins can be plotted in a 2D space. The location in that space is set by the phase difference *P* (*x*-axis) and the amplitude ratio *R* (*y*-axis),

$$P = \text{angle}(A) - \text{angle}(B);\ [0° < P < 360°]$$

It is not practical to use the amplitude ratio directly. If the magnitude of antenna *A* is $|A|$ and of antenna *B* is $|B|$ the ratio is set by,

$$R = \frac{|A| - |B|}{|A| + |B|};\ [-1 < R < 1]$$

This allows one of the signals to be zero. Noise will have random locations in that space. In Figure 4A the locations in a single bin over 4 seconds are plotted. Each dot is one of 4×125 measurements. The distribution in space depends on the statistics of the noise in a bin. The *randn* function of Matlab[4] is used to generate normally-distributed pseudorandom numbers.

If a circularly polarized carrier wave signal is added with a 3 dB signal-to-noise ratio (SNR) on the antenna signals, the locations in space (Figure 4B) will become concentrated at the signal location. The relative phase of the circular polarized signal components is -90° and their amplitude ratio is 1. With an SNR of 3 dB in a single bin the concentration in space is already significant.

Concentration in space is the number of measurements per unit of area over time. However not only the location in space, but also the power is available in the measurements. Noise will have random locations and random power (Figure 5A). A carrier wave signal with a 3 dB SNR (Figure 5B) will express itself by a higher concentration and by a higher power. The sum of the noise and the carrier wave signal is concentrated around the location of the carrier wave signal. A higher signal to noise ratio results in a greater concentration of power in space.

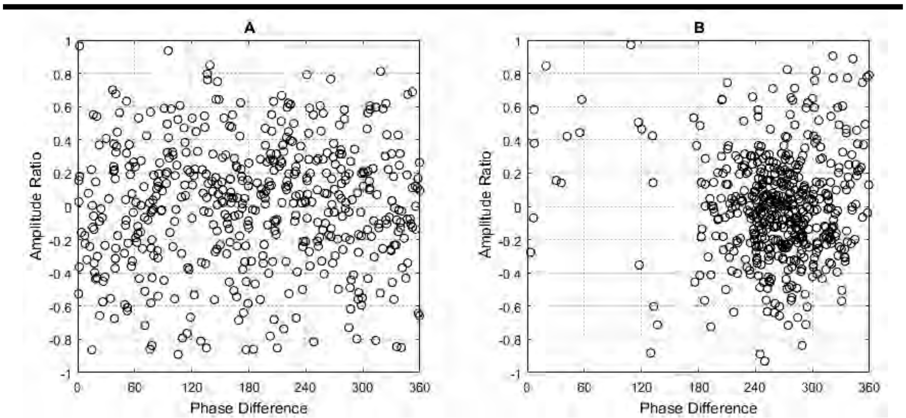


Figure 4 - The concentration in space in a single bin (23 Hz bandwidth) of just noise on both antennas (A), and the same noise with a circular polarized signal with 3 dB SNR (B). Each dot is one of 4×125 measurements over 4 seconds.

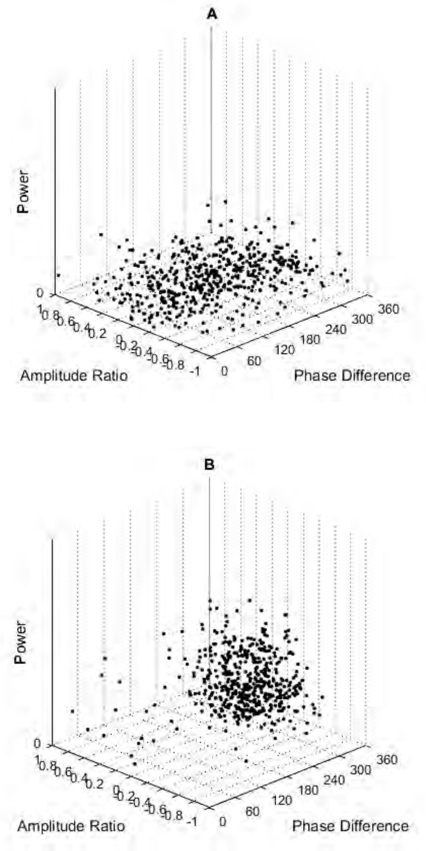


Figure 5 — The concentration of power in a single bin (23 Hz bandwidth) of only noise (A) on both antennas and the same noise with a circularly polarized signal with 3 dB SNR (B). Each dot is one of 4×125 measurements over 4 seconds.

Extracting the Spatial Information

The spatial information we need is in the concentration of power in space over time. It is derived by calculating per unit area the sum of the power over time. Calculating over time is a low-pass filter operation of the power in space over the measurements. Calculating per unit area the sum of the power in space is a 2D low-pass filtering operation. Two filters are needed: low-pass filtering in time and low- pass filtering in space.

The Spaces

First the locations in space of the measurements have to be quantized to reduce the size of the space for processing. The processing is performed for all 256 frequency bins. The space of each frequency bin is called a subspace. The size of the subspace is set to 16 by 16 forming 256 locations. A further reduction is implemented for SSB by combining the measurements of 4 consecutive subspaces into a single subspace. This reduces the number of subspaces from 256 to 64. In this way 64 (16×16 size) subspaces have to be processed. CW needs only about a quarter of the 4 kHz spectrum and a single frequency bin is used in a subspace.

Figure 6 shows the 64 (16×16 size) subspaces. Four subspaces are sliced out of the full 3D space. The white square indicates a possible location of a signals power in space on a single frequency in subspace 45. Instead of processing it as a 3D space, all subspaces are combined in a single 2D space. The single 2D space consists of 64 subspaces in a row. Total size is 16×1024. Thus for SSB the row consists of 64 frequency ranges of 4 bins and for CW of 1 bin.

The 16×1024 space is clearly not a convenient display presentation. The total space is reorganized for displaying as a 2D image consisting of 8×8 subspaces (Figure 7).

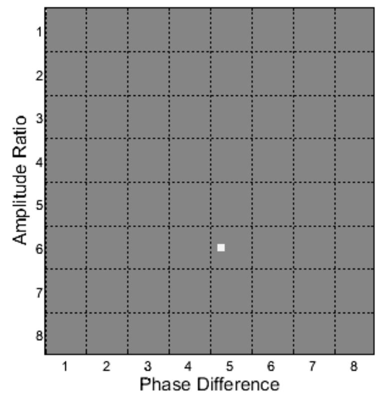


Figure 7 — All 64 subspaces are combined for displaying in a 128×128 image. The white square indicates a possible location of a signal power in subspace 45. Top-left lowest frequency, bottom-right highest frequency.

Low Pass Filtering in Time

The filtering in time of the 16×1024 2D space is implemented as a *RC*-like filter with an attack and a decay time constant. Each location in that space is filtered over time. In Figure 8 the default time constants are 25 ms for attack and 38 ms for decay. Each dot is a measurement in 8 ms steps.

In Figure 9 the low-pass filter is applied, otherwise there would be only 4 measurements in the plot. Noise is spread over the whole subspace (Figure 9A). The signal concentrates the measurements in subspace (Figure 9B). Because 4 bins are combined in the subspace for SSB, the noise from the other 3 bins is also present. The locations in these bins are also affected by the carrier wave signal as a result of

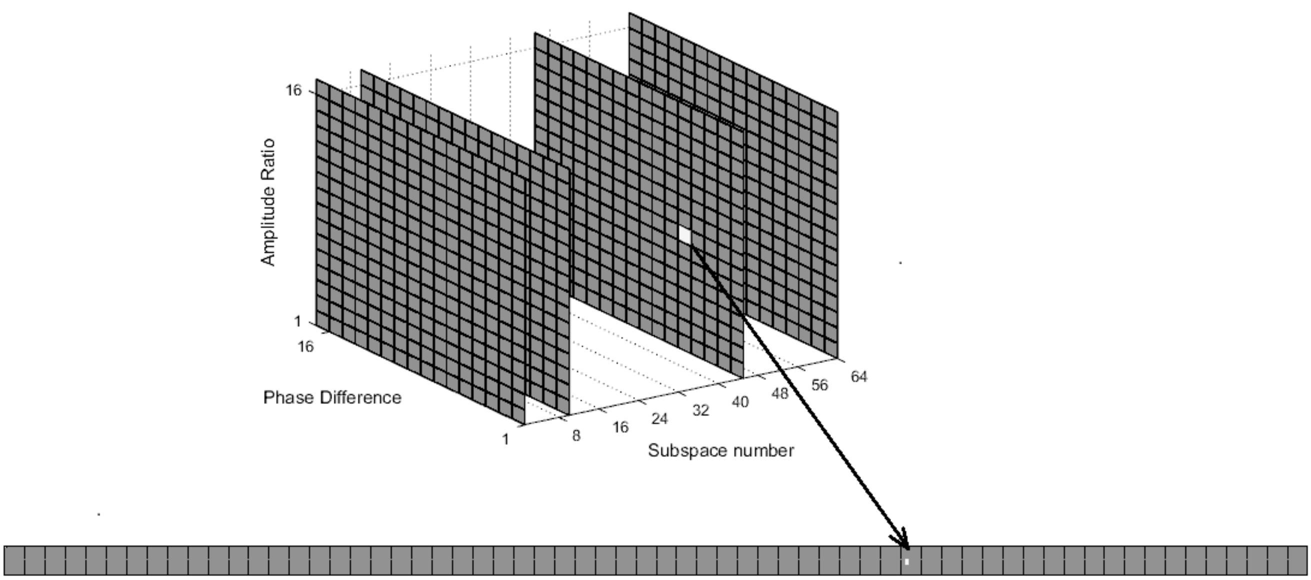


Figure 6 — The 64 (16×16 size) subspaces are combined in a single 16×1024 2D space for processing.

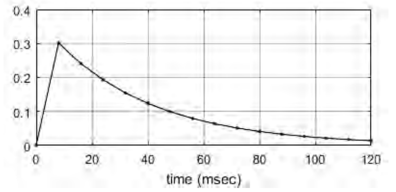


Figure 8 — The *RC*-like low-pass time filter default response over 120 ms on a single measurements with amplitude of 1.

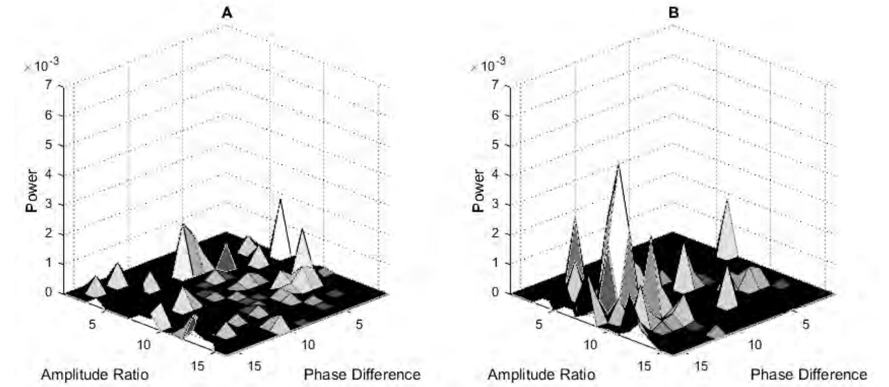


Figure 9 — The effect of the low-pass filtering in time on a SSB subspace with (A) only the noise, and (B) the same noise with a circularly polarized signal at 3 dB SNR.

spectral leakage. For CW the default decay is scaled four times longer. The number of measurements in a subspace is four times lower and a longer decay increases the number of effective measurements.

Low Pass Filtering in Subspace

The low-pass filtering in space is performed by filtering the spectrum of the 2D space. A 2D FFT calculates the spectrum of the 2D space. The spectrum is multiplied with a 2D low-pass filter (LPF). The result is transformed back with a 2D IFFT (Figure 10).

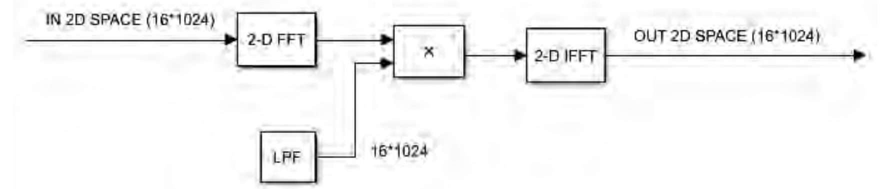


Figure 10 — Block diagram of the low-pass filtering in space.

The coefficients for the LPF can be calculated from the required impulse response in a subspace. If the required response is known, the coefficients are obtained by the inverse-IFFT (or FFT) on that response. The FFT of the response is the product of the spectrum of the input space with a single location filled with an amplitude of 1 and of the LPF. So the LPF coefficients can be calculated by dividing the FFT of the response and the FFT of the impulse in the input space.

As a response a Tukey (tapered cosine) window is used in the phase and in the ratio direction in space, because it is flexible in setting window shapes. The total 2D low-pass filter is the product of the LPF for the phase direction and for the ratio direction.

A signal will concentrate the power at the signal's location (Figure 11). This increases the level at that location at the cost of the other locations in the subspace. This is a positive effect, because it enhances the signal's location.

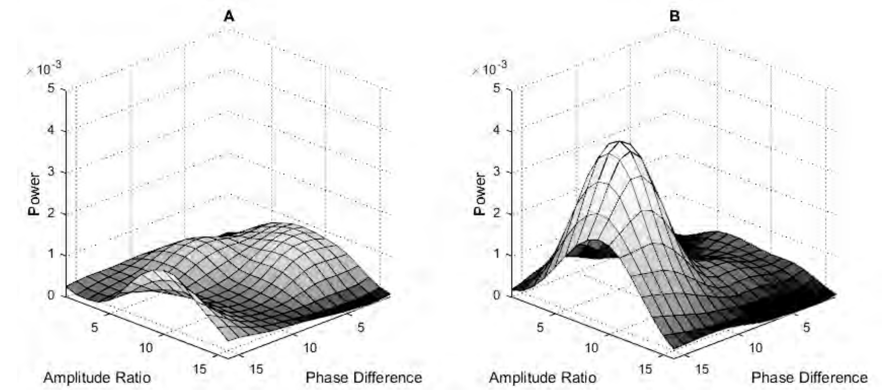


Figure 11 — The effect of combining a time filter and subspace low-pass filter in a SSB subspace with (A) only noise and, (B) with a circular polarized signal at 3 dB SNR.

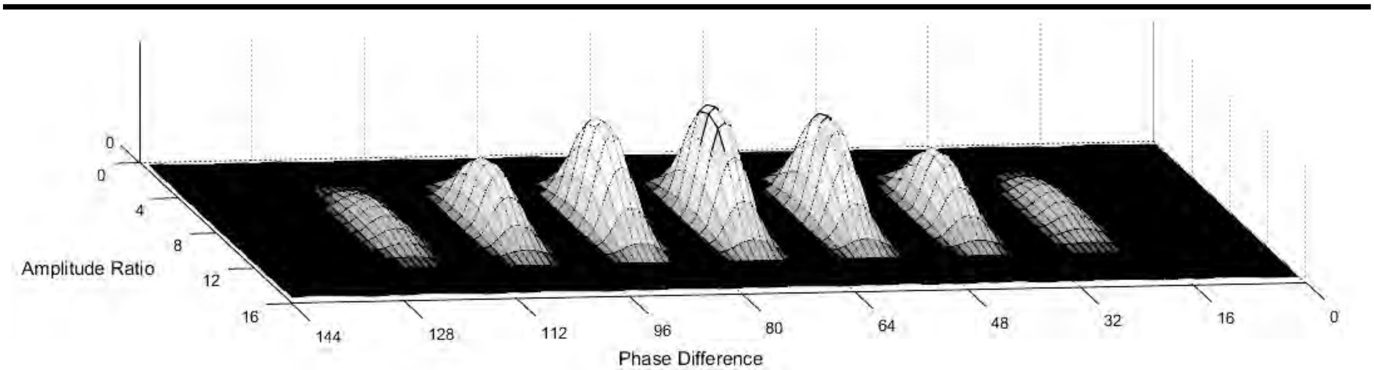


Figure 12 — The response of the 3D filtering in subspace and over frequency displayed in the 2D space. Nine subspaces are sliced to show the response of the total LPF calculating the concentration of power in a volume in 3D space.

Low Pass Filtering over Frequency

Until now the filtering was limited to each single subspace. It is expected that the signal components will overlap in consecutive subspaces. So there is also a concentration of power in subspaces over frequency. The LPF can be extended in the same way to perform a low-pass filtering of locations over subspaces. Figure 12 shows the response using a Tukey window over 9 subspaces.

The combination of 4 bins in a single subspace for SSB acts in-principle as a low-pass filter over frequency. Just like the quantization of the locations in subspace to 16×16 acts as a low-pass filter in the subspace. The span of the 4 bins is about 4.5×15.625 Hz = 70 Hz.

Spatial Filtering

A signal is spatial filtered by weighting the signal components in the frequency bins according the location in a weighted space. The weighted space is based on an estimate of the signal locations — an estimate of the noise as a reference and a weighting function. The locations in the weighted space contain a measure of whether it is a signal or noise. The locations of the signals in the bins are used as pointers to the weighted space.

Estimating the Location of the Signal

The current time-filtered and space-filtered subspaces contain the estimate of the location in space of the present signal. It is a result of calculating the concentration of power in the 3D space over time.

Estimating the Noise Floor

The sum of the power on each location over all subspaces is called bulk space. When it is sampled, if only noise is present, it represents the noise floor of the receiving system and the received noise. It is expected that the level and the statistics of the noise is stationary over time. The noise floor is a simple representation or estimate of this statistics.

The noise floor is shaped by the statistics of the locations in space and the corresponding power: (*concentration* × *power*). The power depends on the selected antenna signal. The locations in space are set by the noise (amplitude) statistics of both antennas. The result is that at the borders of the ratio *R* on the *y*-axis the measurements are less frequent. This frequency dominates in the noise floor level, because it sets the concentration over time. At the borders of the *y*-axis the noise floor level will be lower.

In the example (Figure 13) instead of using the sum of the antenna *A* and *B* signals, the strongest signal according location in space is selected for the power. The noise floor has to be scaled to the actual receivers (audio) bandwidth to get the noise floor in a subspace. Only subspaces within the receiver's bandwidth contain noise. If the bandwidth is smaller the sum of subspaces is lower.

In a practical situation the SSB/CW signal can be present making it difficult to sample the noise floor at the right moment. The sum over all locations in bulk space represents the total power. It is expected that this total power is minimal when only the noise is present. The noise floor is detected by monitoring this power over a period of time and sampling the bulk space when the power has its minimum. Because there will be variations on the noise floor, an averaged minimum is needed. A moving average over 8 noise floor estimations is calculated and sampled.

Weighting by Soft Limiting

Weighting uses information of the noise floor and the current time filtered and space filtered subspaces (2D space) containing the location estimation of the signal. The maximum power in the noise floor is an indication of the expected signal levels. The 2D space is scaled (normalized) to this maximum. This allows an alternative sigmoid function (Figure 14) to be used as a soft limiter. It calculates the weighting gain in space based on this scaled space. The gain is called soft-limited gain (Figure 15).

The soft-limited 2D space locations contain a measure for whether it is a noise (Figure 16A) or signal (Figure 16B). The locations of the signals in the frequency bins are used as pointers to the soft-limited 2D space. The resulting soft-limited gain is used for weighting the signal.

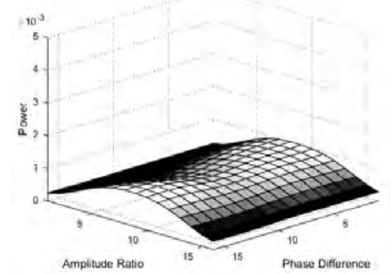


Figure 13 — The scaled sampled noise floor as a result of combining time filtering and space filtering of only noise.

Selecting Antenna Space and Signals

Antenna Space

The location in subspace is the result of the selected antenna signal space. Up until now the antenna signals provided by the linear polarized antennas *A* and *B* are used. Any linear polarized signal can be seen as the sum of two circular polarized signals. The Left and Right hand circular polarized antenna signals *L* and *R* can be calculated by summing both signals with ±90° phase shift. In the frequency domain the phase shift can be calculated much easier by multiplying the complex numbers of the spectrum of one antenna with +/-1i. In this way the linear polarized antennas *A* and *B* provide also the signals of a left hand and of a right hand circular polarized antenna.

Signals

The location in subspace is always set by the amplitude of the two signals of the selected antenna space. The power used in the subspaces however is set by the selected signal. For the best estimate of the location of the signal in space, the power of the selected signal needs to have the best SNR. Even if the location of the signal in space is found, the sum of the signal and the noise will be passed by the noise reduction. It is only filtered by the noise reduction; the noise is

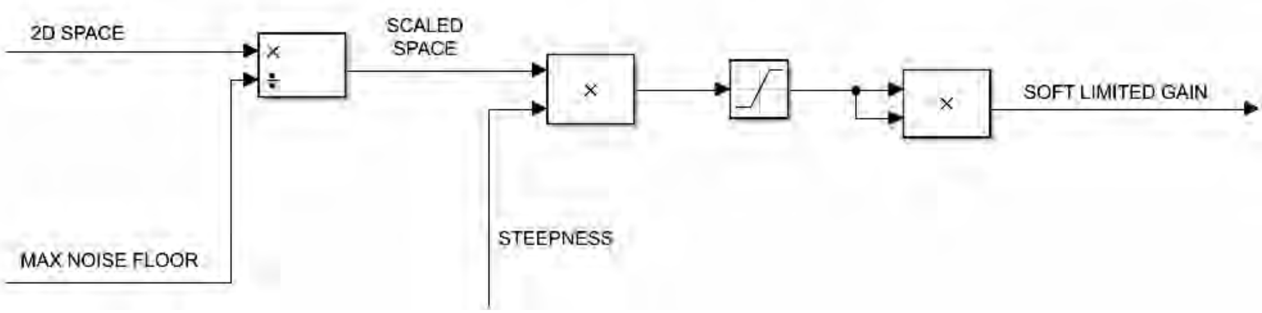


Figure 14 — Block diagram of the alternative *sigmoid* function used as a soft limiter. The scaled space is multiplied by the steepness limited between 0 and 1, and squared.

not subtracted. A few dB or more in SNR can be gained by selecting the signal with the best SNR, especially when using noise cancelling. In that way we gain from having two antennas.

Five signals can be selected:

- Noise cancelled
- Diversity (Mono)
- Diversity (Stereo)
- *A*/*L* (antenna *A* or *L*)
- *B*/*R* (antenna *B* or *R*).

Any combination of the two antenna signals results in a location in space that is noise cancelled.

The polarization diversity signal is a selection between *A* and *B* (or *L* and *R*) signals based on the location in subspace. The strongest antenna signal is selected and used for estimating the location. With Mono the noise reduction (soft-limiter gain) acts on this signal and with Stereo on both antenna signals (enabling Stereo Diversity Noise Reduction). In principle, selection of one of the antenna signals is not needed, but it skips the need for setting the noise cancelling.

Implementation and GUI

The processing, including the GUI of Figure 17, was developed and prototyped in Matlab/Simulink. At the right, the spaces are displayed as an image. The subspaces of a SSB signal are displayed showing frequency selective locations in the subspaces (white dots).

All noise reduction settings are located on the bottom row. The AGC is at the end of the processing. The required space and signal can be selected, the detection of the noise floor can be enabled, the noise floor itself and a dc level can be used as an optional threshold. The detection Noise Floor switch enables finding and updating the noise floor. While it is switched ON, a new noise floor is tracked until switched OFF. The settings of the Time Filter, Space Filtering and the Soft Limiter are at the bottom right.

One Mouse-Click Noise Cancelling

In the GUI bulk space, the sum of all subspaces, can be selected for displaying. In Figure 18 bulk space shows the location of an identifiable man-made noise source in space. It is resized to 128×128 using bilinear interpolation for displaying and for a higher location resolution. The maximum in the display (white cloud) indicates the location. The location can be selected by clicking on it in the displayed image. The acquired coordinates in the image can then be recalculated to the settings needed for the noise canceller. The antenna with the strongest man-made noise is known from the location and it's signal is scaled down before subtracting. So only a single mouse-click is needed to set the noise canceller. The location is valid only if it is not already noise-cancelled by accident. Clicking on the opposite (or another) location in space will reveal the correct location and will show an increase in noise level on the S-meters.

If the man-made noise is present only on one of the antennas *A* or *B*, the location will be spread over the *x*-axis, because the phase difference is in the noise. The maximum is then spread also and is less visible. Switching to the *L*/*R* space will reveal the location of the maximum at ratio $R = 0$ (amplitude *L* equals amplitude *R*) and at a phase difference of 0° or 180°, because it is a linearly polarized signal with a fixed orientation. Instead of noise cancelling in the *A*/*B* space, the *B* or *A* signal can then be selected for processing.

Local man-made noise can also be circular polarized, especially in the near field. The signal at antenna *L* or *R* can be zero. In that case the man-made noise will become

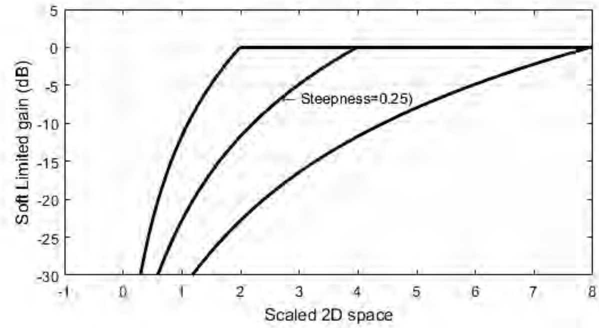


Figure 15 — The alternative *sigmoid* function in decibels. The soft-limited gain as function of the scaled 2D space with steepness: 0.5, 0.25 and 0.125.

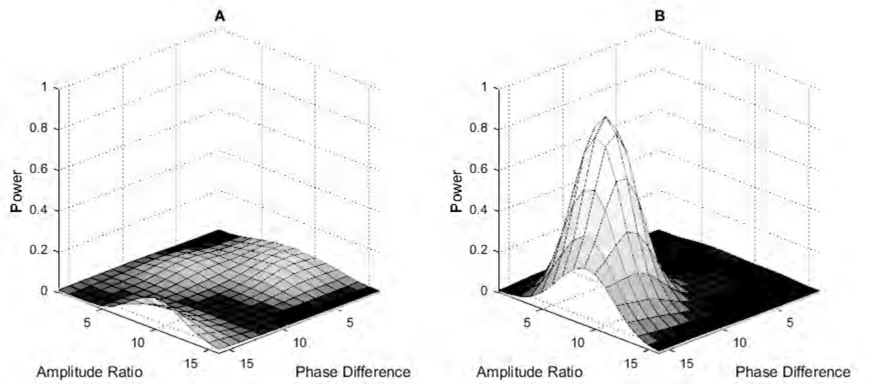


Figure 16 — The soft-limited (weighted) subspace with (A) only noise and, with (B) a signal at 3 dB SNR. Vertical scale is the soft-limited gain.

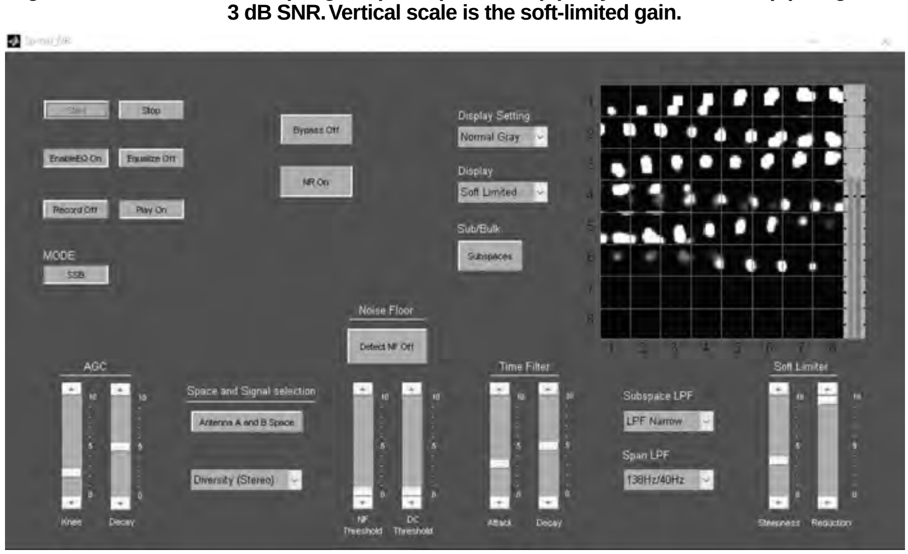


Figure 17 — The GUI with subspaces displaying a frequency selective fading SSB signal.

best visible in the *A*/*B* space at ratio $R = 0$ with a ±90° phase difference. Instead of noise cancelling in the *L*/*R* space, the *R* or *L* signal can then be selected if polarization of the signal allows.

The noise canceller setting result in one space can be converted to the setting needed in the other space. So noise cancelling can be done in the space showing the man-made noise location best and so giving the best noise cancelling setting. This makes it also possible to switch between spaces without needing to update the noise cancelling setting.

In practice, the noise on both antennas will not be perfectly uncorrelated and the noise can show a preference in location. In an urban area this can be caused by a dominant man-made noise source, by poor balancing of the loop antennas or by coupling with surrounding objects or other antennas. This location will be visible in bulk space and the noise canceller can be set to this location for best noise cancelling.

Noise cancelling in effect maximizes the polarization mismatch and/or the radiation pattern for maximum attenuation of the noise source.

Propagation

Knowledge of propagation is needed to understand the signal location[5,6] in space. All shortwave signals travelling through ionized media [in the presence of Earth's magnetic field, — *Ed*.] propagate by two modes, the ordinary (*O*) and extraordinary (*X*) mode[7,8]. The signals of both modes are opposite-elliptically polarized. NVIS signals on 80 m in the Netherlands are nearly circularly polarized during the day, because the ordinary wave predominates. Selecting the left-hand circular signal increases the SNR up to 3 dB. When selecting the right-hand circular signal the SNR can decrease more than 20 dB. Also, on the other bands like 40 m, the signals can be elliptical polarized. When both modes are present at about equal amplitude the resulting polarization becomes linear and the orientation will rotate (drift) over time. Predominantly linearly polarized signals are rarely stable in linearity and orientation. The propagation attenuation and delay of the *X* and *O* mode signals drift independently in time. If the signal on one antenna drifts below the noise the phase difference will become noisy. The location of a drifting elliptically polarized signal traces a nice circle in subspace over time.

Multipath reception can cause frequency selective fading (FSF). It is not only expressed in the frequency selective amplitude, but also in the frequency selective polarization. As a result, the locations in space will be frequency selective. Because the subspaces are filtered independently, the noise reduction can very well cope with that. It limits however the use of the frequency span filtering. A lower

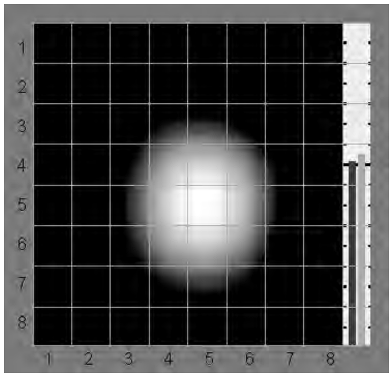


Figure 18 — Bulk space showing the white cloud representing a single 10 dB man-made noise over noise source (scaled to maximum).

span bandwidth can be selected when FSF is present. If noise cancelling has to be used the signal can fade on the noise cancelled location. Diversity is the preferred signal selection when frequency selective fading is present and there is no dominant man-made noise source.

When the polarization of the signal is mainly circular or elliptical, the *A*/*B* space is preferred. The locations are mainly on the *x*-axis and the phase difference is a well defined ±90°.

If the signals are linearly polarized and rotating, the location can be at the edges of *A*/*B* space and the phase difference can become random over the entire 360°. The *L*/*R* space is then preferred, because the linear signals are on the *x*-axis and the phase difference is better defined.

Both the spaces *A*/*B* and *L*/*R* use the phase difference. If one of the antenna signals drops in amplitude because of drifting polarization, the phase difference becomes noisy and even can be set by the noise if that signal becomes zero. This affects the concentration in the subspaces. It is however compensated by the 3 dB higher signal level of the other antenna signal.

Phase Coherent Receivers

Both receivers must be as identical as possible. Most critical is the phase behavior. Common names are phase coherent, phase locked and phase synchronous. However, they don't necessarily indicate the same phase. It is essential that both receivers use the same local oscillator signals for equal frequency and equal phase. It is not practical when the phase difference is not fixed and jumps to other values when tuning over the band. As a result every change in frequency needs measuring the changed phase difference again or resetting the noise cancelling again.

Less critical are the phase/gain frequency responses of both receivers. As long as the phase/gain frequency response over the audio band is fixed it can be equalized in the processing. A non-flat frequency response can be flattened in the processing. The pass band must be flat, because a single noise floor reference is used for all frequency bins. HF filtering affects the phase difference also when tuning over the band. Fixed differences like gain differences and phase offsets can be calibrated out and so are less critical. The AGC must be performed after the noise cancelling and noise reduction.

There are professional analog receivers designed for diversity reception (such as Collins R390A) or direction finding (such as Telefunken Telegon) that can be used. However they are rare or expensive. Some ham transceivers support diversity reception, but the phase difference is not fixed. Another option is to make two analog receivers (such as two Elecraft K2 receivers) phase-fixed by sharing the same local oscillators.

SDR receivers are a modern and practical implementation of potentially identical phase-fixed receivers. During the development of the noise reduction, I switched from using two analog K2 receivers to using the ANAN200D.[9] The ANAN200D receivers however — as practically all SDR receivers for ham radio with multiple spectral capture units — are not phase fixed. The phase difference is not known and can jump to other values when tuning. That is the main reason receiver equalization in the processing is still needed. Equalization uses an external noise source connected to both receiver inputs. *PowerSDR* software runs on a second separate computer so all processing power of the first processing computer stays available for the noise reduction. The analog line-out of the ANAN200D is fed via a sound card to the processing computer. Running it all on one computer might be possible, but the phase difference in the *PowerSDR* audio interfacing is not controlled.

Test Signals and Screen Recordings

Most actual band signals vary too much in SNR for testing purposes. The question is at what minimum SNR level are the signals still readable. Because of fading actual band signals are usually not constantly at that controlled level.

One set of test signals can be made artificially in a separate *Simulink* model by combining actual band man-made noise and actual band signals with a very high SNR. Signal parts with about equal signal strength are selected out. The content of the conversation is not relevant. In that way the overall SNR can be controlled by adding the signal to the noise with a tunable level. This is valid, because actual signals are also the sum of band noise and signal. In the *Simulink* model the rms levels of the noise and of the signal can be measured so the SNR level can be set to known values. A time constant of 125 ms is used in the rms measurements. The peak rms value of the signal represents the peak value of the transmitter's power and so of the signal strength.

A second set of test and demonstration signals are actual band signals without modifications. Artificial noise and CW are also used as a test signal. Audio noise reduction examples and screen recordings are presented[10] in audio samples and screen recordings.

Discussions

A single strong man-made noise source has to be resolved for best noise reduction results. It behaves like a signal and has a preferred location in space. This affects the noise reduction, because the location of a signal is pulled to the location of this noise source depending on the signal's amplitude. For the same reason, diversity is not effective when a strong man-made noise source is present. Diversity is preferred if no single dominant man-made noise source is present. A signal cannot then be cancelled or attenuated by noise cancelling.

Conclusions

With the one mouse-click noise cancelling even the weakest identifiable man-made noise source or any noise preference can be cancelled fast and easily. The presented stereo diversity noise reduction is very effective by using the signals of two orthogonal antennas. It is based on the concentration of signal power over time in a 3D space. Only minimal information of the SSB and CW signal is needed. The noise reduction has one problem. Except for the very weak signals you can't tell anything about the signal strength just by listening. You must switch off the noise reduction for that.

The filtering in the subspaces can be regarded as polarization filtering, not including direction of arrival. In that way the noise reduction uses filtering over time, frequency and polarization. At the same time it can benefit from selecting the best space and best signal for the actual band conditions. It enables stereo diversity noise reduction.

Selecting the *best space* and *best signal* are the most relevant settings. In an urban area the noise cancelled signal is the best selection most of the time, unless the location of the man-made noise source coincides with the signal location in space.

Displaying the polarization behavior of the signals over frequency gives feedback for a better understanding of the propagation trends over time. The presence of elliptical polarization or frequency selective fading becomes visible. It also helps in selecting the best space and *best signal* for the band conditions.

Displaying both the location of the signals and of the man-made noise enables you to see if the location of the man-made noise and the location of the signal coincides.

Jan Simons, PAØSIM, has been a licensed radio amateur since 1975. He works as a mixed analog/digital electronic engineer for Oce/Canon, a printer and copier company in The Netherlands. Developing mixed signal ASICs for piezo inkjet print heads and methods for using the piezo as a sensor for detecting underperforming nozzles is a part of his job. As a radio amateur, besides the technical part of the hobby, he especially enjoys working DX and CW on shortwave.

Notes

[1]Additional information and noise reduction examples of SSB and of CW signals; **www.pa0sim.nl/QEX.htm**.

[2]J. M. M. Simons, PAØSIM, "Effective Directivity for Shortwave Reception by DSP," *QEX*, July/Aug. 2006, pp. 37-45.

[3]**www.katjaas.nl/FFTwindow/FFTwindow&filtering.html**.

[4]**www.mathworks.com**.

[5]B. A. Witvliet, PE5B, "Near vertical incidence skywave: Interaction of antenna and propagation mechanism," PhD dissertation, University of Twente, Enschede, The Netherlands, 2015.

[6]M. C. Walden, "High-Frequency Near Vertical Incidence Skywave Propagation," *QEX*, Jul./Aug. 2017, pp. 21-34.

[7]**www.pa0sim.nl/XOpropagation.htm**.

[8]Eric Nichols, KL7AJ, "Gimme an X, Gimme an O," *QST* Dec. 2010.

[9]**https://apache-labs.com/**.

[10]Audio samples and screen recordings: **www.pa0sim.nl/QEX.htm**.