By Rob Frohne, KL7NA

A High-Performance, Single-Signal, Direct-Conversion Receiver with DSP Filtering

By adding a DSP demodulator to a well-established and popular receiver design—KK7B's R2—you can have a bit of the latest filter technology at your fingertips!

W ith so many new radios incorporating digital signal processing (DSP) for demodulation and filtering, I got the itch to play with one myself. By "play with one," I don't mean merely *operate* a DSP-equipped receiver, I wanted to be able to put *my own* software into the receiver and put to work some of those nifty signal-processing ideas I teach others how to use!

Why DSP?

You may wonder "What's the big deal?" Why are so many of the new radios supplied with DSP demodulators and filters? It's because digital techniques have several advantages over their analog counterparts. Probably the most exciting aspect of this is that everything is done in *software*. This means that when I design a new demodulator, *you* can download it to *your* radio and *you* then have my new demodulator! No soldering; no hunting for parts; no ugly modifications—it's *so* easy! Another benefit is that the results are predictable. Component tolerances are *not a factor* as they are with analog signal processing. Simulations are easier to do, and their results more closely approximate what happens in a real system. Another advantage results from the speed of today's digital signal processors. Nowadays, it is quite

[1]Notes appear on page 45.

realistic to have a digital filter with the equivalent of *hundreds* of poles. (In the analog domain, it takes one inductor or capacitor to make one pole.) Few analog designers use more than *ten* poles in a filter, because it's very difficult to make an analog filter with more than about *ten* poles work properly.

Some Background

This project started in 1994. Ralph Stirling, KC3F, and I did exactly what Rick Campbell, KK7B, suggested in his R2 receiver article.[1] (Get your copy of that article and have it handy!) We used Rick's R2 receiver and a Texas Instruments C5X DSK DSP starter kit to construct a DSP-based direct conversion (D-C) receiver. We modified the DSP kit with our custom daughterboard containing another TLC32040 codec. The DSP board replaced the analog phase-shift network with its own digital counterpart and a sharp digital filter. With this setup, we were able to obtain an opposite-sideband rejection of almost 50 dB. However, the receiver wasn't quite ready to take home and tune to 20 meters—the gain distribution needed adjustment.

About this time, Motorola unveiled the DSP56002EVM evaluation board.[2] It made obsolete our work with the TI board. The Crystal Semiconductor 16 bit stereo codec on the Motorola board is much better than the two TI 12 bit TLC32040 codecs. The

Motorola 56002 processor uses 24 bits of precision; the TI TMS320C5X uses only 16. It was obvious that we needed to start over.

That start didn't come until I taught a communication systems course during the winter quarter of 1996. As a homework assignment, I had my students try the Motorola EVM (evaluation module) in place of the TI DSK. It proved simple to modify a filter program supplied by Motorola to do the basic filtering necessary for SSB demodulation, and it worked much better than the TI DSK-based receiver. I spent many hours during the summer improving on that design. This article describes that improved system.

Implementation

We chose KK7B's R2 receiver not only because of its high dynamic range and ready availability, but for the image-reject downconverter it uses. Its in-phase (I) and quadrature (Q) channels provide the signals needed to demodulate *any* form of modulation resulting in a band-pass signal.[3] These are called the *in-phase modulation*, x(t) and the *quadrature modulation*, y(t) signals. (See Figure 1.) The demodulators described here are for AM, DSB, CW, SSB and ISB, but because accommodating other modes needs only new software, we can expect more modes to be available in the future.
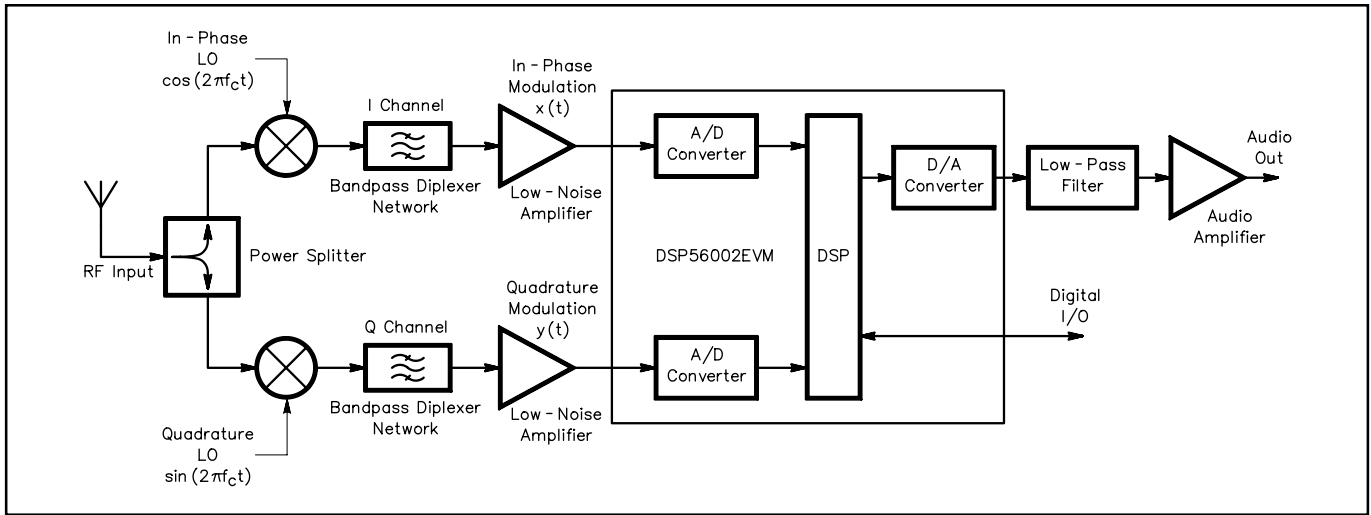
Figure 1—A block diagram of the receiver showing how the DSP56002EVM fits into the R2 receiver signal path. The signals x(t) and y(t) are all that are necessary for demodulating *any* signal in the passband. The DSP software presently demodulates SSB, CW, AM and ISB, but all that is necessary to accommodate any other mode is new software. There are only three hardware connection points needed. It's *so* easy, you've got to try it!

The R2 receiver also provides a very good match for the Motorola DSP EVM. To duplicate this project, all you need to do is connect the analog I/O connectors of the DSP board to three places on the R2 receiver board. It is a delight to be able to put this project together without any ugly dead-bug wiring or extra analog circuit design! More on that under Hardware.

Project cost is modest. The R2 kit is available for about $109 from Kanga; the circuit board alone is available for $20.[4] With group orders, I've seen the Motorola DSP board price as low as $85; standard price is about $150.[5] The addition of an NVROM (discussed later under Boot ROM,) adds a few dollars to the total cost.

The steep skirts of the digital filters are advantageous on SSB and CW. Simulated and measured filter responses for those modes are shown in Figures 2A and B. The simulation does not include quantization effects of the analog-to-digital converter (ADC) or the processor. The filter shape factors are all less than 1.3:1.

Opposite-sideband rejection is at least as good with the DSP demodulator (we measured it at almost 50 dB) as it is with the original analog demodulator. Figure 3 shows a simulation of the opposite-sideband rejection for SSB and CW. The simulation assumes that all the analog electronics are perfect and that there is no quantization error. With these assumptions, the sideband rejection is much greater than the 50 dB we measured. It is very important that the I and Q channel components be matched carefully and that the local oscillator (LO) provides equal amplitude signals, 90° out of phase.
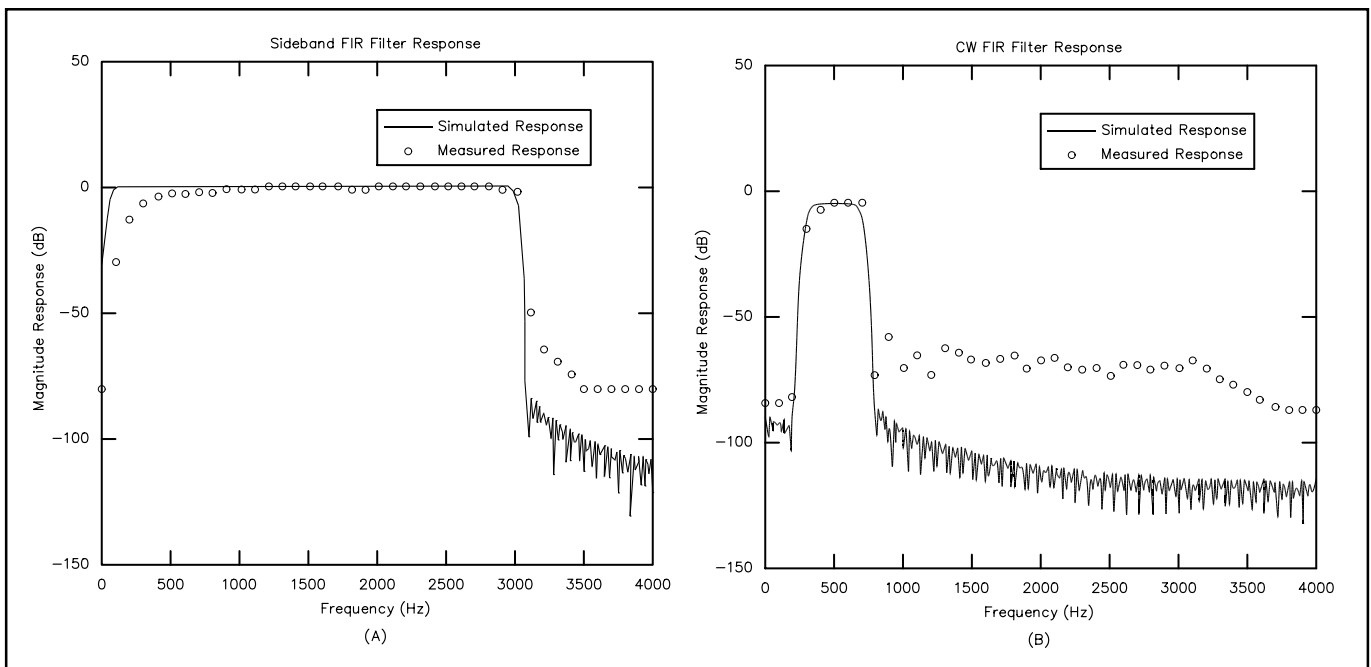

Figure 2—At A, simulated and measured frequency response for SSB. Note the steep skirts. The filter shape factor is 1.1:1. It would be even better, but the dominant low-frequency response is from the analog diplexer. The FIR design cutoff frequencies for this filter are 60 Hz and 3 kHz. The CW filter response curves (B) show a shape factor of less than 1.3:1. This is much better than the 2.1:1 shape factor of the analog demodulator. The FIR design cutoff frequencies are 300 Hz and 700 Hz.
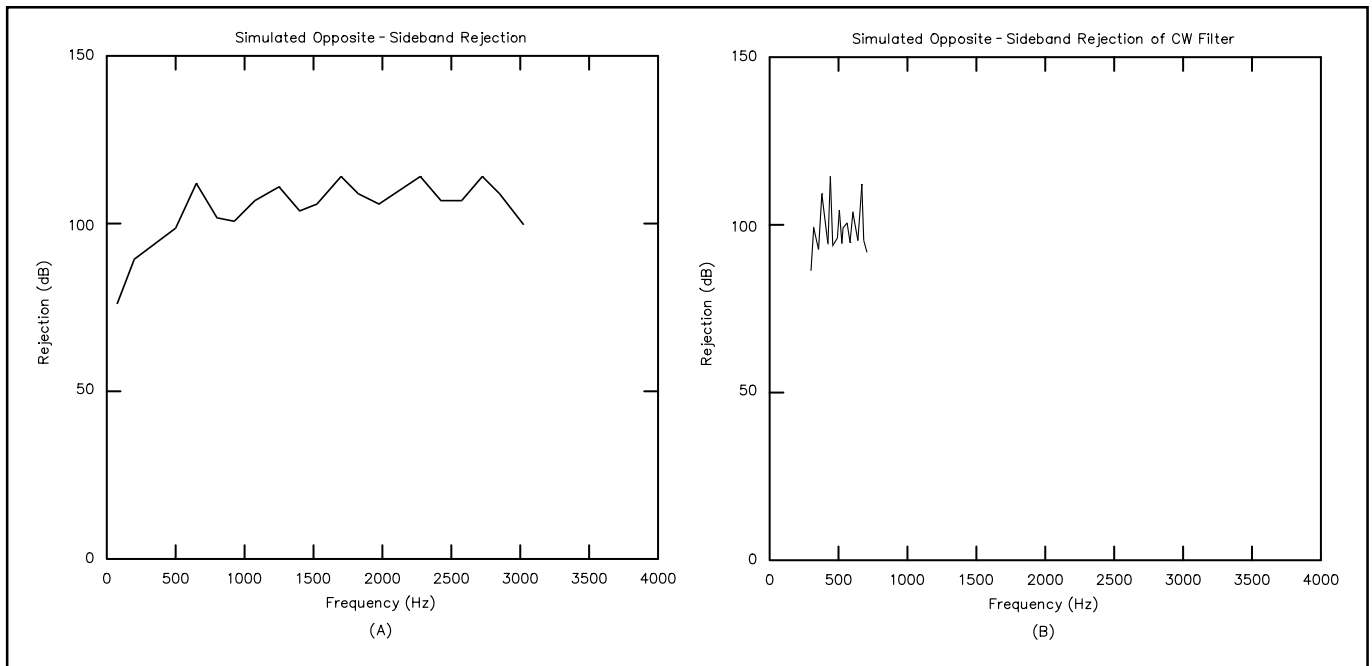
Figure 3—SSB opposite-sideband rejection (A) is limited only by the FIR filter's finite length. This filter's response is good enough to blame the opposite-sideband leakage on the analog components preceding the FIR filter. At B, simulated opposite-sideband rejection of the CW filter, assuming everything is perfect except the FIR filter. Quantization effects are also neglected. The FIR filter is not the cause of much opposite-sideband leakage. This, however, is not the case for all FIR filters tried.

There is a possible 0.5 dB gain mismatch between channels of the codec. I made no effort to measure and eliminate this, but it would be easy to do. The advantage of the DSP demodulator is that there are fewer components that must be matched. It might be worth the time and effort to match those more carefully.

Because of the analog filtering before the DSP stage, I expected the IMD dynamic range to be the same with digital demodulation as it is with analog demodulation. Measurement bears this out. On the other hand, the blocking dynamic range turns out to be about 13 dB poorer with the digital demodulator than with its analog counterpart. It is still quite respectable, however, at 113 dB. Perhaps it is reasonable to keep both demodulators, switching between them as necessary, especially because the analog demodulator is already on the R2 board.

Hardware

Connecting the DSP demodulator to the R2 receiver is simple. Other than the R2 board, the Motorola DSP EVM, some shielded wire, a few connectors, and, optionally, a switch or two and enclosures, no other hardware is needed. It is *very easy*!

EVM and R2 Interconnections

The Motorola EVM codec has a wide range of adjustability for input and output signal levels. This is quite important. It makes it possible to connect to the R2 receiver in just the places that Rick suggested

without adding any gain or attenuation. The EVM uses miniature stereo connectors for audio input and output. The current software uses one output channel for LSB and the other for USB. This allows us to demodulate independent sideband (ISB). Beyond this stage, only *one* sideband is amplified.

Basically, the R2's audio phase-shift networks are replaced by the DSP board. Refer to the R2 information I told you to have on hand for the following steps: On the R2 board, disconnect the wire from C10 to R20 and C21, and reconnect it to the left audio-input channel of the DSP board. Similarly, disconnect C20 from R35 and C26, and reconnect it to the right audio-input channel of the DSP board. These two connections send the in-phase modulation, x(t), and the quadrature modulation, y(t), to the two analog input channels of the DSP EVM. (See Figure 1.)

The analog output must be connected to the R2's audio amplifier. To do this, disconnect the wire from the output of U5A to C25. Depending on which sideband you want to demodulate, connect the wire leading to C25 to the right or left audio output of the DSP board. All other connections remain as they were for the R2 with the original analog demodulator. I suggest placing the DSP EVM and the R2 board in separate enclosures and using shielded cable for all interconnections.

Boot ROM

One step I took to make the receiver run

stand-alone (without having to use a PC to load the software every time you turn it on) was to insert an ATMEL AT29C256 non-volatile, read-only memory (NVROM)[6] into a socket soldered in the area marked **U10** on the evaluation board. (It was thoughtful of Motorola to provide this luxury for us.) This NVROM can be programmed on-board using free software available on the Internet or a preprogrammed IC can be purchased from Kanga.[7, 8] The NVROM is an accessory well worth the small additional investment.

Software Description

The software demodulates AM, DSB, CW, SSB and ISB. I picked SSB and CW because they are my favorite HF modes. I like to have an AM demodulator to use when listening to WWV and shortwave broadcast stations. DSB can be demodulated with the AM detector, so it is a freebie. I implemented ISB because it was so easy to do, and having it made changing sidebands easier. ISB doesn't seem to be used much on the ham bands, but it could be used for simultaneously transmitting an SSTV picture on LSB while explaining it on USB voice. These modes don't exploit all the advantages of a DSP demodulator, but they certainly illustrate it. This receiver is just *crying* for some software for the digital modes! Can't you hear it?

Sampling Rate

I tried several different sampling rates between 8 kHz and 48 kHz. Sampling

theory advises us to use a rate at least twice as fast as the highest frequency component in the signal we are processing. Faster sampling rates actually cut the quantization noise a small amount, but they leave less time for processing the signal between samples. I finally settled on an 8 kHz rate because the filters I used to get good opposite-sideband rejection required more processing time. With the Motorola EVM, you can set the processor clock speed to 80 MHz via software. (Motorola claims some of their EVMs will operate at that speed.) I decided I would rather have the software work with *all* the EVMs, so I didn't try 80 MHz. Yet another possibility is to use the new Motorola DSP56303EVM. Its processor is twice as fast as the 56002, and it uses the same codec.
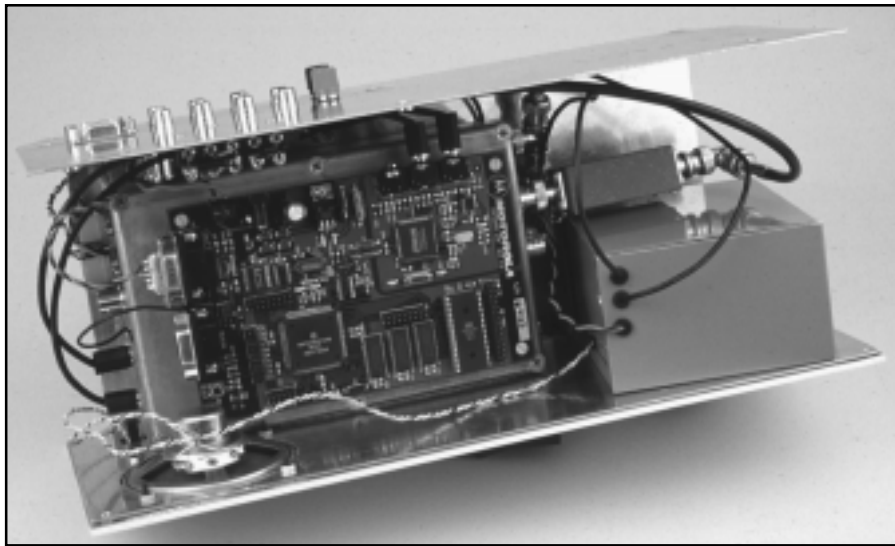
AGC

In all modes, the software incorporates 22.5 dB of AGC. The AGC is different than that of most receivers. You will *not* find it useful in keeping your eardrums in one piece. Its purpose is to keep the signals to the ADC in range as long as possible. The dynamic range of the ADC is greater than 80 dB.[9] Because of the narrow filters preceding the ADC, I noticed no AGC pumping from a signal 20 kHz off frequency, until it was −10 dBm (more than 60 dB over S9). At that level, an oscillation at about 1 Hz ensues.

Designing and debugging the AGC turned out to be one of the most challenging parts of this project. The codec has provisions to let the DSP know when an overload has occurred. It also provides a way of adjusting the gain (on the fly) by up to 22.5 dB in 1.5 dB increments. The gain is changed only when the signal passes through zero. This helps eliminate "zipper noise" created by sudden gain changes when the signal is at a high level. The AGC software checks for an overload condition. When it finds one, it lowers the gain. Then it doesn't check again until the codec has had time to clear the overload condition and change the gain.[10] The AGC has a fast attack of about 0.75 dB/ms, and a slow decay of about 6 dB/s. There is a noticeable—but unobjectionable—cycling effect when AGC is active because of the discrete gain steps of 1.5 dB. This effect occurs, of course, only on the very loudest signals (above about −50 dBm, or more than 20 dB over S9) and is noticeable only on constant-level carriers.

Demodulation

The generalized demodulation scheme follows that shown in Figure 1. I chose this scheme because it retains all the information present in the modulation in the signals x(t) and y(t). With only software changes, we can demodulate any form of modulation. Actually, limitations in the speed of—or memory available to—the DSP, or the large bandwidth of the modulated signal may prevent this, but I don't foresee that happening often.

SSB, ISB, AM, DSB and CW can all be demodulated in DSP using only linear operations. Two channels of finite impulse response (FIR) filtering are employed for this. These are the I and the Q channel filters (see Figure 4 and 5). FIR filters were chosen because of their simplicity. Any band-limited linear operation can be approximated by selecting the proper FIR fil-

---

Designing Custom Frequency Responses for FIR Filters

Designing your custom frequency response into a FIR filter is really quite easy. All you need to modify are the FIR filter coefficients. Presented here are only the formulas to tailor the cut-off frequencies to your liking. If you are familiar with calculus, investigate the references.[* † ‡]

Three steps are involved in calculating the filter coefficients. The first is to find the unwindowed I and Q filter coefficients:

$$h_I(m) = \frac{1}{\pi m}\left[\sin\left(\frac{2\pi m f_H}{f_s}\right) - \sin\left(\frac{2\pi m f_L}{f_s}\right)\right] \qquad \text{(Eq 1)}$$

and

$$h_Q(m) = \frac{1}{\pi m}\left[\cos\left(\frac{2\pi m f_H}{f_S}\right) - \cos\left(\frac{2\pi m f_L}{f_S}\right)\right] \qquad \text{(Eq 2)}$$

$$m\varepsilon\left\{-\frac{M}{2},...,\frac{M}{2}-1\right\} \qquad \text{(Eq 3)}$$

The desired lower cutoff frequency is f_L, and the upper one is f_H; f_S is the sampling rate (8 kHz in this case). M is the number of filter coefficients you use. The software presently uses 256 coefficients for each channel. This is the reason M goes from −128 to 127. Note the similarity between the I and Q coefficients. The only change between them is the sine terms becoming cosine terms. This provides the 90 degrees of phase delay necessary. If you are modifying the response curve for AM or DSB, set $h_Q(m) = 0$.

The next step is to apply the windowing function. The purpose of the windowing function is to reduce the sidelobes (bumps in the filter response caused by Gibbs' phenomenon). The Kaiser window I used is a semioptimum windowing function that allows trading sidelobe levels for sharpness of the transition region, by varying the parameter, β. I used a β=8.0. Other windowing functions are possible.[**] To apply the windowing function, multiply each coefficient by its window value, given by:

$$w(m) = \frac{I_0\left[\beta\sqrt{1-\left(\frac{2m}{M}\right)^2}\right]}{I_0(\beta)} \qquad \text{(Eq 4)}$$

where $I_0(.)$ is the zeroth-order modified Bessel Function of the first kind.

The final step is to plot the response curve actually obtained (neglecting, of course, quantization and finite wordlength effects). If you like the response, great. If not, go back to step one. Remember: The dynamic range of the codec limits the stopband attenuation to about 80 dB. The equation for computing this frequency response is:

$$|H(f)| \approx \sum_{m=1}^{\frac{M}{2}} 2\,h(m)\sin\left(\frac{2\pi m f}{f_S}\right) \qquad \text{(Eq 5)}$$

It is exact for an odd number of coefficients, M, and very close for an even number of M. (All these steps are automated if you have access to *MATLAB*. The *MATLAB* code uses an exact relation for any number of M, even or odd.) *MATLAB* will also plot the theoretical opposite sideband rejection if everything except the FIR filter is perfect. The software that does this is available at my Web site: **http://www.wwc.edu/~frohro/R2-DSP.html**.—*Rob Frohne, KL7NA*

*John Ash, KB7ONG, Fred Christensen, KA6PNW, and Rob Frohne, KL7NA, "DSP Voice Frequency Compandor for use in RF Communications," *QEX*, Jul 1994, pp 5-10.
†Rob Frohne, KL7NA, "A Practical Note on Designing FIR Filters," Feedback, *QEX*, Oct 1994, p 32.
‡John Wiseman, KE3QG, "A Complete DSP Design Example Using FIR Filters," *QEX*, Jul 1996, pp 3-15.
**Alan V. Oppenheim and Ronald W. Schafer, Discrete-Time Signal Processing (Englewood Cliffs, Prentice-Hall, 1989).

An inside view of the receiver. The Motorola DSP EVM sits atop an aluminum enclosure that contains the R2 components. The gray box to the right houses the LO. The small box nearby is the 40-meter band-pass filter that sits between the antenna input and the input to the R2.
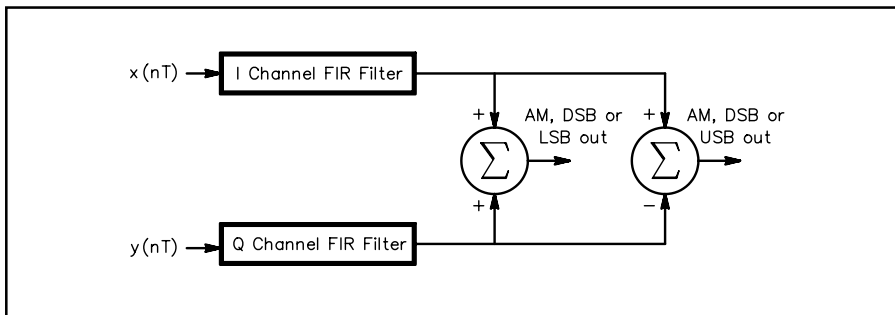

Figure 4—Block diagram of the program used for all modes: SSB, CW, AM, DSB and ISB. The signals x(nT) and y(nT) represent the sampled in-phase and quadrature modulations, respectively. With the SSB, CW, ISB demodulator, the I and Q channels have identical gain characteristics, but the Q channel's phase lags the I channel's by 90°. The outputs for USB and LSB (or one of the outputs for AM and DSB) are sent to the two output channels of the stereo codec.
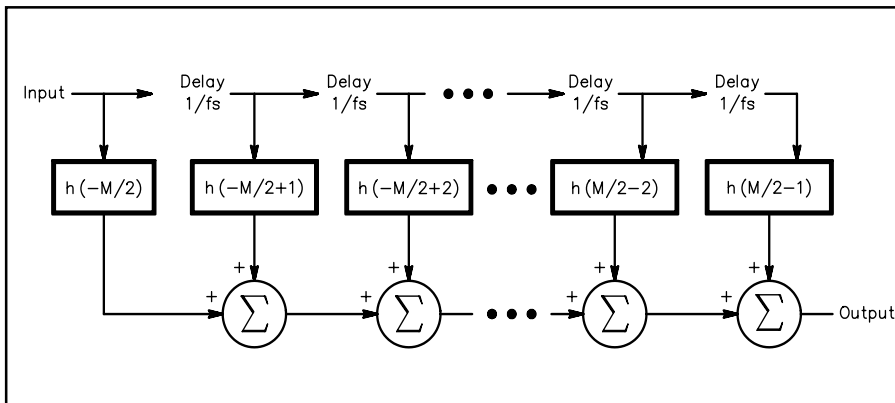

Figure 5—FIR filter represented as a tapped delay line. There are M filter coefficients, $h(\bullet)$, and the delays are each $1/f_s$ second, which is the time between samples.

ter coefficients, h(M). The more coefficients used, the better the approximation. A FIR filter is really just a tapped delay line. The coefficients are the weights of each of the delayed signals that are summed at the output. This makes programming particularly easy; there is only one program, but there are eight sets of coefficients, two for each demodulation type. To design your own custom frequency response, you need not understand the details of the program—you need only understand the basics of FIR filter design. (If this interests you, see the sidebar "Designing Custom Frequency Responses for FIR Filters" and John [KE3QG] Wiseman's article[11] for designing your own custom filter responses.) After x(t) and y(t) are filtered, their sum yields the one sideband, and their difference yields the other sideband. Each sideband is sent to a channel of D/A on the stereo codec.

MATLAB[12] is a very useful tool for computing filter coefficients, plotting the corresponding frequency-response curves and computing the opposite-sideband rejection of the whole receiver, assuming everything is perfect except the FIR filters. The only things I didn't simulate in *MATLAB* were the effects of quantization of the ADCs and DACs, the finite word length of the DSP and the imperfections in the analog circuitry. I was grateful that most of the design could be so easily simulated. It shortened the testing and debugging phase of the project by many, many hours.

AM and DSB Demodulation

The same product detector is used for AM and DSB demodulation. A product detector ac-coupled to the output demodulates either AM or DSB modulation. Using a product detector has the advantage of having a heterodyne for zero beat. I find this particularly handy to tune in the signal exactly.

The I channel is simply passed straight through. The Q channel is unused. Actually, its gain is set at zero so it won't affect anything when the outputs of the two channels are summed. See Figure 4. To accomplish this, the I filter coefficients are set to {1,0,0,0,0,...}, and those of the Q filter to {0,0,0,0,0,...}. These coefficients let the R2's analog filters and the DSP's antialias filter do all the filtering for AM and DSB demodulation. The narrow, analog, elliptical, filter bandwidth of my R2 board is fine for listening to shortwave broadcast stations. In the future, however, I intend to try a Butterworth design having a wider bandwidth. With that, a faster sampling rate and the right coefficients, the DSP could tailor the frequency response

CW, ISB and SSB

For CW, ISB and SSB, the I and Q channel filters have identical-magnitude responses; the phase of the Q channel's filter

Table 1
Receiver Performance Specifications, SSB

SSB Receiver	R2 with DSP Demodulator	R2 with Analog Demodulator
Minimum Discernible Signal	–116 dBm	–116 dBm
Blocking Dynamic Range	113 dB	126 dB
IMD Dynamic Range	89 dB	90 dB
Shape Factor	1.1:1	1.6:1
Opposite Sideband Rejection	40 to 50 dB	40 to 50 dB

response is delayed by 90° with respect to the I channel. See Figure 4. The band-pass filter widths are those appropriate for the different modes (3 kHz for SSB and 400 Hz for CW). This scheme yields single-signal reception.[13]

The filter responses are created by defining the passband desired and computing the coefficients from that data. Finally, the coefficients are tapered at the ends by multiplying them by a windowing function. The windowing function sacrifices steep skirts for a more-uniform, high attenuation in the stop-band. If the windowing function is omitted, stop-band sidelobes from the Gibbs phenomenon[14] are relatively large. After trying several windowing functions, I finally settled on a Kaiser window.[15] It has an adjustable parameter, ß, to control the trade-off between a steep transition region and lower sidelobe levels. You can use the formulas given in the FIR filter design sidebar to compute your own filter coefficients and plot the resulting frequency-response curves. All the software, including the *MATLAB* code, is available free on the Internet and from the ARRL BBS.[16]

Test Results

Table 1 provides a summary of the measured parameters compared to those of the original analog demodulator. SSB was used for all the measurements (see Table 1). I used several old HP-606A signal generators for the LO and test signals because they are virtually free of phase noise. I measured the dynamic range—both two-tone IMD and blocking—at 20 kHz spacing. To measure the output, I used a Fluke 8910A true RMS voltmeter. The power splitter used for the dynamic range tests is a Mini-Circuits PSC2-1. The LO phase-shift network I used is a one-pole crossover network.[17]

Table 1 shows some of the trade-offs between the analog and DSP demodulators. The DSP wins on shape factor, but loses on blocking dynamic range. Don't forget, however, to factor in the *versatility* of the DSP. The real advantages of the DSP demodulator won't be seen until someone starts using this setup to design custom demodulators for some of the fancier digital modes.

Summary

I've enjoyed listening to Europe, South America and the Pacific from my office using this receiver with only a UHF discone on the roof.

Download my software! Play with it! I'm looking forward to the day when I'll hear you on 20 meters saying "Hey Rob! Why don't you download *my* latest demodulator!"

Acknowledgments

I want to thank Ralph Stirling, KC3F, who did the hardware portions of the first generation DSP demodulator with the TI DSK, my communication systems students for trying this out for me in their homework assignments and for reading the manuscript, and Jim Forsyth, WA7CLI, for constructing a respectable cabinet for the R2-DSP receiver.

Notes
[1] Rick Campbell, KK7B, "High-Performance, Single-Signal Direct-Conversion Receivers," *QST*, Jan 1993, pp 32-40.
[2] Johan Forrer, KC7WW, "Using the Motorola DSP56002EVM for Amateur Radio DSP Projects," *QEX*, Aug 1995, pp 14-20.
[3] Leon W. Couch II, "Digital and Analog Communication Systems," 4th ed. (New York: Macmillan Publishing Company, 1993). A band-pass signal is one whose spectra is concentrated about a carrier frequency.
[4] Kanga US, Bill Kelsey, 3521 Spring Lake Dr, Findlay, OH 45840, tel 419-423-4604 e-mail **kanga@bright.net** and **http://qrp.cc.nd.edu/kanga/**.
[5] The DSP EVM can be purchased directly from Motorola and from any of several outlets such as Hamilton Hallmark, Newark, Arrow, Wiley, etc. Check your telephone book Yellow Pages or visit **http://www.mot-sps.com/sales/sales_web.html**.
[6] A list of Atmel product distributors can be found at **http://www.atmel.com/atmel/contents.html**.
[7] The free software is *FLASHBOOT.ASM*, by Chris Hanna and Helmut Strickner, and is available from the TAPR site at **ftp://ftp.tapr.org/tapr/SIG/hfsig/upload/flshboot.zip**.
[8] See Note 4.
[9] A data sheet for the Crystal Semiconductor CS4215, 16-bit multimedia audio codec, is available at **http://www.cirrus.com/products/overviews/cs4215.html**. Crystal Semiconductor Corporation is a division of
Cirrus Logic Inc, 3100 West Warren Ave, Fremont, CA 94538, tel 510-623-8300, fax 510-252-6020, **http://www.crystal.com**.
[10] The codec uses a digital filter in its sigma delta ADC that has a pipeline of 16 data points, so we must wait until those have passed before we check for the overload condition.
[11] John Wiseman, KE3QG, "A Complete DSP Design Example Using FIR Filters," *QEX*, Jul 1996, pp 3-13.
[12] *MATLAB*, The MathWorks Inc, Cochituate Pl, 24 Prime Park Way, Natick, MA 01760; **http://www.mathworks.com/products/**.
[13] Gary Breed, K9AY, "A New Breed of Receiver," *QST*, Jan 1988, pp 16-23.
[14] Jerry B. Marion, "Classical Dynamics of Particles and Systems," 2nd ed. (New York: Academic Press, 1970), p 133.
[15] Alan V. Oppenheim and Ronald W. Schafer, "Discrete-Time Signal Processing," (Englewood Cliffs: Prentice-Hall, 1989), pp 452-462.
[16] This software is available at **http://www.wwc.edu/~frohne/R2-DSP.html** and from the ARRL by ftp from **oak.oakland.edu/pub/hamradio/arrl/qst-binaries** and from the ARRL BBS 860-594-0306..
[17] Rick Campbell, KK7B, "High-Performance, Single-Signal, Direct-Conversion Receivers," *QST*, Jan 1993, p 37, Figure 6A, and Gary Breed, K9AY, "A New Breed of Receiver," *QST*, Jan 1988, pp 16-23.

Rob Frohne, KL7NA, is a professor of electrical engineering at Walla Walla College. He holds a BSE from Walla Walla College and an MSEE and PhD from Purdue University. Rob was first licensed in the mid-'70s as a teenager and has previously held the call signs WL7IIP, KL7IIP and V63NA. He enjoys working DX, especially long-haul on 20 meters.

You can contact Rob at 214 W. Whitman Dr, College Pl, WA 99324, e-mail **frohro@wwc.edu.**

New Products

MINI BOOM MIKE FOR HAND-HELDS
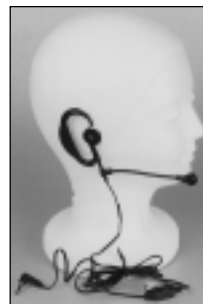
◊ SPM-400 series mini boom mikes, manufactured by Premier Communications, work with all popular Amateur Radio and commercial hand-held radios. The new SPM-400 features a comfortable, adjustable rubber ear hook and a clip-on PTT switch for hands-free operation. Specs: Mike impedance, 1000 ohms; speaker impedance, 8 ohms.

Four models are available: SPM-400 (ADI, ICOM, Standard and old-style Yaesu); SPM-401 (Kenwood); SPM-402 (Yaesu FT-50 and VX-1R); and the SPM-403 (for Motorola Radius-compatible radios).

Prices: $25.95 (400); $27.95 (401); $20.95 (402); $29.95 (403). For more information, contact Premier Communications at 909-869-5711; **http://www.adi-radio.com**.