PSpice Applications in the Teaching of Communications Electronics

**Andrew Rusek, Barbara Oakley**
Department of Electrical and Systems Engineering
Oakland University, Rochester, Michigan 48309

Abstract

Many parameters of circuits and devices commonly used in communication electronics can be profitably simulated using the free educational version of PSpice. We have created a broad variety of PSpice macromodules for use in classroom and laboratory teaching, including macromodules that simulate pulse width modulators and demodulators, delta encoders and decoders, generic class C amplifiers, frequency synthesizers, noise generators, bandpass filters, AM modulator/demodulators with noise and interference, analog correlators, oscillators, and phase locked loops. Some of the most interesting schematics and simulation results are demonstrated, and the use of these simulations to stimulate classroom and laboratory learning is discussed. We have found that the triangle of lectures, simulation, and experimental work based on initial simulation forms the optimal method of instruction in our communication electronics course.

I. Introduction

Oakland University's communication electronics course (EE437) is designed to serve as a link between the high level communication system, the subsystem functional block level, and the electrical parts level (Figure 1). The high level communication system might contain, for example, a generic delta modulator as a "black box" subsystem that performs an essential function in the overall system. The subsystem functional block level PSpice implementation of that delta modulator fleshes out the workings of the device with generic blocks, using a integrator, for example, rather than an op amp "wired" as an integrator (Figure 2). The electrical component level realization of a delta demodulator is then composed of the actual electrical parts that are used in the circuit. This is simulated by using the "real" PSpice electrical components, rather than the functional blocks (Figure 3). Of course, this electrical part level is what is finally constructed from electronic components such as resistors,
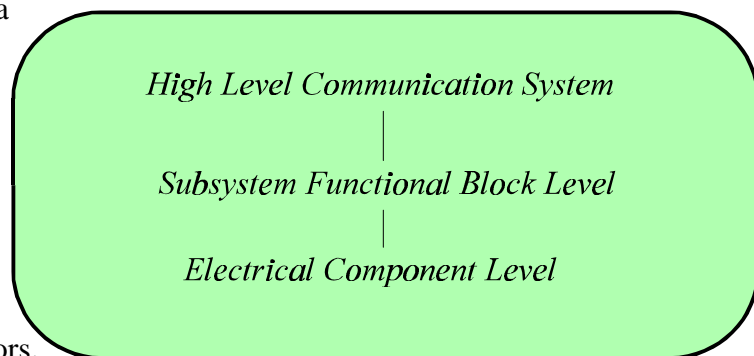
Figure 1: The three levels linking a typical communications system.

capacitors, and operational amplifiers. Using PSpice in this fashion forms a complement, and an enhancement, to the usual methods of working with PSpice in the class-room and the laboratory.[1,2,3,4]   When communication electronics is presented to students using this three-level approach, we've found that students "get it" more quickly. If, for example, we present the general communication system into which a delta modulator fits, and then present the functional blocks within the delta modulator that help it to perform its work, students see the big picture and the main ideas immediately. Translating the functional blocks into a PSpice electrical part implementation, and ultimately into a real circuit, is then straightforward. The schematic shown in Figure 3 was, in fact, composed by students and then used by them to assist in constructing a delta modulator from actual electrical components in the laboratory. By the time the students built the circuit, they had a good idea not only of the workings of their subsystem, but also of its limitations. In Figure 4, for example, the signal to be encoded by the delta modulator is shown in the top portion of the figure. A slow signal, which is well within the sampling limitations of the modulator, is indicated by the green waveform. The red signal beside the green is a much faster oscillation which was applied deliberately to see whether this signal exceeded the modulator's sampling limitations.

The corresponding signals, as they have been encoded by the modulator, are shown in the middle of the figure. The decoded signals are shown at the bottom. Yellow and dark blue show the original signals which were to be encoded. The red and green signals reveal the feedback. The slower (green) signal conforms nicely to the original. The red signal, however, shows a slew that clearly demonstrates the original signal is changing too quickly to be followed. Through playing via simulation with these types of examples, students can quickly and easily test the limitations of their circuit design before they actually construct it.

Of course, the three separate levels, as they have been outlined above and illustrated in Figure 1, are not quite so distinct in practice. For example, the subsystem functional block level shown in Figure 2 includes not only the generic PSpice integrator and amplifier blocks, but also a 7474 IC, resistors, and voltage sources. These types of "real" components often operate faster in PSpice simulations than functional block components.

We use the communication-related macromodules we've developed in several ways. First, while in class, we discuss the overall system into which a delta modulator fits. Next, while still in class, we discuss the workings of a delta modulator by displaying and discussing the functional block level PSpice schematic on a large screen attached to our computer. We then run the PSpice probe post-processor to generate plots that demonstrate the effects of the subsystem (as in Figure 4). We often change a few parameters (such as the frequency of the input signal) and then rerun the simulation. In this way, students begin to see how simulations can be used to gain an intuitive feel for how the subsystem works and what it does.

Finally, we place copies of these subsystem function block level PSpice macromodules on the computers in the communication electronics laboratory. Students perform eight experiments through the course of the class—these macromodules serve as a foundation to allow students to

begin formulating their own working electronic part level designs. These subsequent designs, as well as the original macromodules, then form an integral part of the hands-on experimental process by allowing students to quickly and easily test system limits and various portions of their experimental designs. By using a wide variety of sophisticated PSpice simulations such as these during their laboratories, students become much more attuned to the broad scope of capabilities inherent in PSpice, and to the value of using simulation to assist in their experimental work. A list of the macromodules we have developed and discussed in this paper is shown in Table 1. These macromodules were largely developed using the professional version of MicroSim's PSpice 8.0, but are easily translatable into the student version. The general topics covered in the communication electronics course are shown in Table 2.[5] Another set of related examples used in our graduate-level high frequency electronics course is covered in a companion paper: "PSpice Applications in the Teaching of Wireless and High Frequency Electronics."

Table 1: PSpice macromodules developed for the communication electronics course
a. Delta modulator/demodulator
b. Generic class C amplifier
c. Phase locked loop and voltage controlled oscillator
d. Frequency synthesizer
e. Bandpass filter (with noise)
f. Noise generator
g. AM modulation/demodulation (with noise and interference)
h. Pulse width modulator/demodulator
i. Analog correlator
j. Colpitts oscillator

Table 2: Topics Covered in the Communication Electronics Course
1. RF amplifiers and oscillators
2. Signals and signal spectra, noise, signal filtering
3. Classical AM modulation, transmitters, and receivers
4. Other analog modulators, DSB, SSB
5. FM, PM Modulation, modulators, transmitters, receivers
6. Phase locked loops
7. Introduction to pulse and digital modulation, systems, and circuit components

Working with PSpice macromodules such as these also allows students to easily do a section-by-section breakdown to analyze a circuit they intend to build. This helps students to more easily pinpoint design problems. For example, a phase-locked loop (PLL) can first be constructed in PSpice using inherent PSpice functional blocks, such as integrators, rather than simulated op amps. Once the whole PLL system is working, with such parameters as gain and offset appropriately adjusted, students can begin finalizing the design of individual portions of the system. For example, the voltage controlled oscillator (VCO) is finalized by replacing the functional blocks within the VCO with "actual" electrical components. Since only changes in this one section affect any changes seen in the output waveform, it is easy to debug any resulting problems.

II. Description of the PSpice Modules

a. Delta Modulator/Demodulator
This macromodule was discussed above, and is illustrated in Figures 2, 3, and 4.

b. Generic Class C Amplifier
Figure 5 shows the PSpice implementation of a generic class C amplifier. An RLC filter is emulated by a PSpice generic filter block. Figure 6 reveals the associated waveforms. The small input signal activates a limiting block which simulates the transfer characteristics of a transistor. The "transistor" is on for only a small fraction of the period, thus distorting the consequent drive waveform. The filter is used, in this case, to filter out the fundamental frequency for the resulting output wave. Alternatively, the filter could be used to filter out harmonics, thus allowing the macromodule to serve as a frequency multiplier.

c. Phase-Locked Loop and Voltage Controlled Oscillator
Since sections of this macromodule are also used in other macromodules, it is worth discussing with some care. The functional block illustration of a PLL with a VCO used as input is shown in Figure 7, while the affiliated waveforms are shown in Figure 8. The PLL is represented by the shaded portion in the upper part of Figure 7, and the VCO used to provide an input signal for to the PLL is surrounded by yellow in the lower, non-shaded portion of figure. V(V17:+) represents the sinusoidal input to this VCO–the input could as easily have been a square wave (as shown in Figure 9) or sawtooth (Figure 10). The 1.000 input block to the VCO controls the free run frequency of the device.

The VCO operates via the usual mechanism–the input signal is multiplied against the corresponding output signal, and the resultant is fed into an integrator, which is in turn characterized by its own gain and initial condition (as shown in Figure 7). The gradually increasing or decreasing output signal from the integrator is sent to a comparator. The output signal of this comparator, multiplied by a variable gain (0.3 in this case), is fed back into both the negative terminal of the comparator and the integrator. This configuration results in the hop-scotch up and down signal characteristic of a VCO, with the variation in V17 providing for the variation in the VCO's output pulse width. The gains of the amplifier and integrator, as well as resistor and capacitor values, are all "tweakable" by students, allowing for optimization of the circuit design. Standardization of the amplitude of the pulses supplied to the phase detector portion of the PLL can be obtained in two ways, both of which are shown. In the first method, diodes and voltages provide for clipping (these are at the bottom right of the figure), while in the second method, a PSpice functional block is used to do the clipping (this limiter functional block immediately precedes the blue phase detector block of the PLL).

The PLL consists of the usual phase detector (shown in blue), a filter (implemented by means of a PSpice functional block), and a VCO (this latter VCO is shown in orange in Figure 7). The voltage immediately after the -1 inverting amplifier, V(R9:2), represents the phase inverted demodulated output from the lowpass filter. Students can separately test various units–phase

detector, filter, VCO–and change them as necessary to match simulated parameters to real parameters.

Using this type of PSpice macromodule, students can clearly and easily see the effects of changing system parameters. For example, changing variables related to the filter can change the overshoot of the response of the system to pulses (as shown in Figure 9). The voltage to frequency conversion factor of the VCO can be altered by a simple change of amplification. All of these adjustments are typical of the process of PLL optimization, and they mimic the kinds of design processes and decisions students may be called upon to make later on, as engineers.

Perhaps most importantly, these types of macromodules allow students to quickly and actively understand the workings of complicated circuitry with minimal frustration. Figure 10, for example, shows a PLL system very strongly "driven" with a large deviation in input frequency. The top frame shows the input signal that modulates the frequency of the VCO–the function of this input signal is to test the limits of the PLL. As seen in the middle frame, the PLL VCO also shows a very broad range of frequencies. Due to the limited bandwidth of the PLL filter, the reconstructed signal is very distorted. Following the rising phase of the lowest frame, one can see that the demodulated signal follows the shape of the input, almost reaching 1 volt, then the signal abruptly falls down, showing that the PLL has gotten out of control. When the input ramp decreases, the output signal is once again able to "catch up," so that the input ramp is once again reconstructed. As students clearly begin to see, the point or threshold when the output signal starts "catching up" again is the capture range boundary. The distance between such boundaries is the capture range, while the distance between the peaks of rising and falling ramps is called lock-in range. Students can see the workings of the PLL even more clearly by pulling up a plot such as that of Figure 11, which shows V(Voutfilt) vs. V(Vinramp). When the PLL is working well, the input signal should be translated into frequency of the PLL input signal. Two signals are shown–one output that is able to follow the input signal, and one that is not.

d. Frequency Synthesizer
The frequency synthesizer shown in Figures 12 and 13 illustrates how the VCO frequency becomes a multiple of the input signal frequency. In addition, the figure demonstrates tuning processes by means of a ramp. Students can see how far the frequency synthesizer can be tuned because they can easily change the amplitude of the ramp. Note that the system makes use of a PLL driven by a VCO, much as described in section c above. However, the PLL in this case uses an RC circuit rather than a PSpice functional block to simulate the filter.

e. Bandpass Filter (with noise)
The functional block bandpass filter is demonstrated in Figure 14. When the bandpass filter has a low quality factor (20 dB), the output response of the filter is shown in Figure 15–it responds more quickly to changes in the input, which in this case consists of a sine burst at 465 KHz. When the bandpass filter's quality factor is changed to 60 dB, the output waveform changes so as to react more slowly and ring far longer (Figure 16).

f. Noise Generator

A PSPice noise generator macromodule is shown in Figure 17. "Noise" is added to a voltage from a file of random numbers, producing a resultant noise-ridden output as shown in Figure 18. This macromodule is then used as a functional block within other systems, such as the AM Modulator/Demodulator and the Analog Correlator.

g. AM Modulation/Demodulation (with noise and interference)

The AM Modulator/Demodulator is shown in Figure 19. An interfering signal is supplied by the portion of the circuit highlighted in yellow. Simulated noise is input into the system by the circuit highlighted in orange (see the noise generator in section f above). The antenna circuit is simulated by a filter, as indicated in light blue. The output waveforms (Figure 20) reveal a great deal about how badly the system can be corrupted by interference from a neighboring channel, and by random noise. The input signal is shown as V(intelligence) at the top of the figure. Underneath is the input signal plus modulation. Below that is the input signal with both random noise and simulated interference from a neighboring channel. The output of the IF amplifier is next. Lastly, the output of the detector is shown–a moderately clean signal, considering the corruption. Students can easily increase either noise or interference to see the resulting effects on the output waveform.

h. Pulse Width Modulator/Demodulator

The functional block schematic for a pulse width modulator (PWM) and demodulator is shown in Figure 21, with corresponding waveforms shown in Figure 22. The varying signal (V2) is added to the input signal (V1) and both signals are applied to a block that includes threshold and limiting functions (shaded portion of Figure 21). The waveform is then amplified and put through a limiter to result in the PWM output. The lower, non-shaded portion of the circuitry shows how the PWM wave is demodulated using integrators and a sample and hold unit.

i. Analog Correlator

The functional block representation of an analog correlator is shown in Figure 23, with corresponding waveforms shown in Figure 24.   The operation of this correlator is best understood by looking at the waveforms. An input pulse (V(input), third waveform from the top) is corrupted with "noise" (fourth waveform from top) from a noise source (the noise macromodule discussed above). The resulting sum of input and noise voltage is shown in the second waveform from the bottom. As can be seen, the presence of a signal buried in the noise is virtually undetectable.   The correlator works by sending a "sieve" waveform, similar to the original pulse, which is swept through time with a parametrically imposed time delay (V(move)–the second waveform from the top). The output of the correlator is shown as a function of time (bottom), with the peak output of the correlator shown at the top of the figure. As the swept pulse begins to overlap with the original pulse buried in the noise, the output of the correlator begins to rise. One would expect that the peak of the correlator would appear at the time corresponding to the exact center of the input pulse. However, due to inaccuracies related to the convergence of the PSpice program, the peak of the correlator output arrives slightly ahead of the center of the input pulse.

j. Colpitts Oscillator

The functional block schematic of a Colpitts oscillator is shown in Figure 25, with corresponding waveforms in Figures 26 and 27. The output of the oscillator (indicated by the voltage probe V) is fed back through the inductor to the input of the limiter. The inductor and the 100 pF capacitor each provide for a 90E phase shift, so the input to the limiter is shifted in phase by 180E. G1 is an amplifier can be thought of as adding a small energy boost to make up for energy lost in the resistor. In a very real sense, the sigmoidal limiter, combined with G1, acts like a transistor with soft limitations. They also provide for an additional 180E phase shift, for a total of 360E, to achieve positive feedback. The frequency of oscillation of the oscillator is roughly $1/2\delta\sqrt{LC_{eq}}$, where $C_{eq} = (C_1C_2)/(C_1 + C_2)$. The output voltage across C_1 is conventionally thought of as being cut down by the voltage divider formed by L_1 and C_2.

Students can easily change the 20K Ù load to see harmonics and distortions in the operating frequency using Fourier analysis embedded in PSpice probe (Figure 28). They can also see that too strong a feedback, i.e., too much amplitude on the feedback signal caused by an improper voltage divider, can produce clipping which results in a square wave instead of a sine wave. Likewise, they soon find that if the feedback is too weak, oscillations may not be created–that is, the output voltage, once reduced by the voltage divider, and then multiplied by the gain, may not prove to be sustainable when the resistive load takes its own share of energy.

III. Conclusions

The types of circuits and devices that can be simulated using PSpice can be breathtaking in their variety. But unless students have a broad exposure to many of the different types of tricks that can be played to emulate the performance of a circuit, they have little understanding of the power of the simulation software at their fingertips. Introduction of PSpice macromodules into a senior level electronics-oriented class, such as a communication electronics course, is an excellent way to stimulate students' imagination. This paper has provided a guide to some of the ways that simulation can be developed and profitably used in such a course.
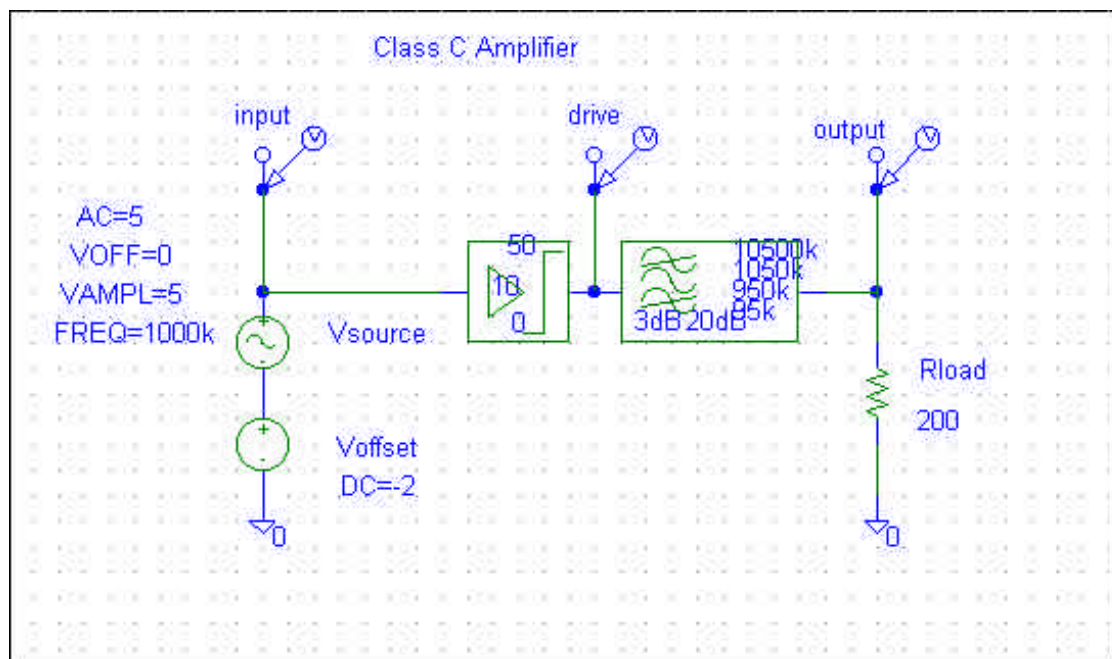
PSpice lends itself nicely to a hierarchical presentation structure. Presenting subsystems initially in terms of functional blocks allows students to quickly grasp the fundamental functions of a circuit. Refinement to actual electronic components is then straightforward. Perhaps most importantly, allowing students to "test drive" fundamental communication circuits and play on the fly with important design parameters in circuits they are building makes labs, projects, and classes much more interesting and educational.

Figure 2: Schematic of major components of the delta modulator/demodulator subsystem. Notice that although some actual components are present, such as the 7474, the resistors, and the voltage sources, many of the more complicated functions are implemented by generic PSpice blocks, such as the integrator and the amplifier.

Figure 3: PSpice component level implementation of a delta modulator.

Figure 4: Waveforms affiliated with the delta modulator circuit.

Figure 5: Functional block PSpice representation of a Class C amplifier with generic band-pass filter.
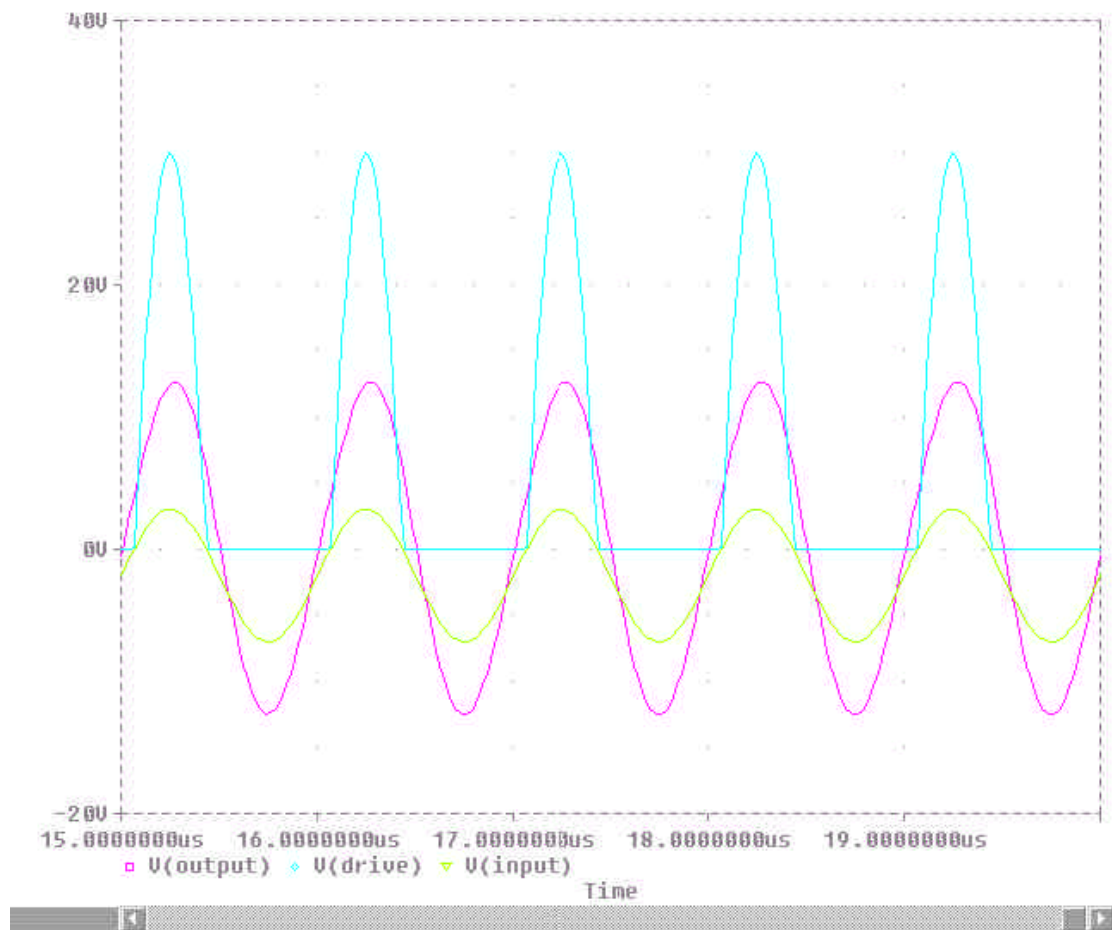
Figure 6: Input, drive, and output signals of the functional block PSpice Class C amplifier with generic band-pass filter.
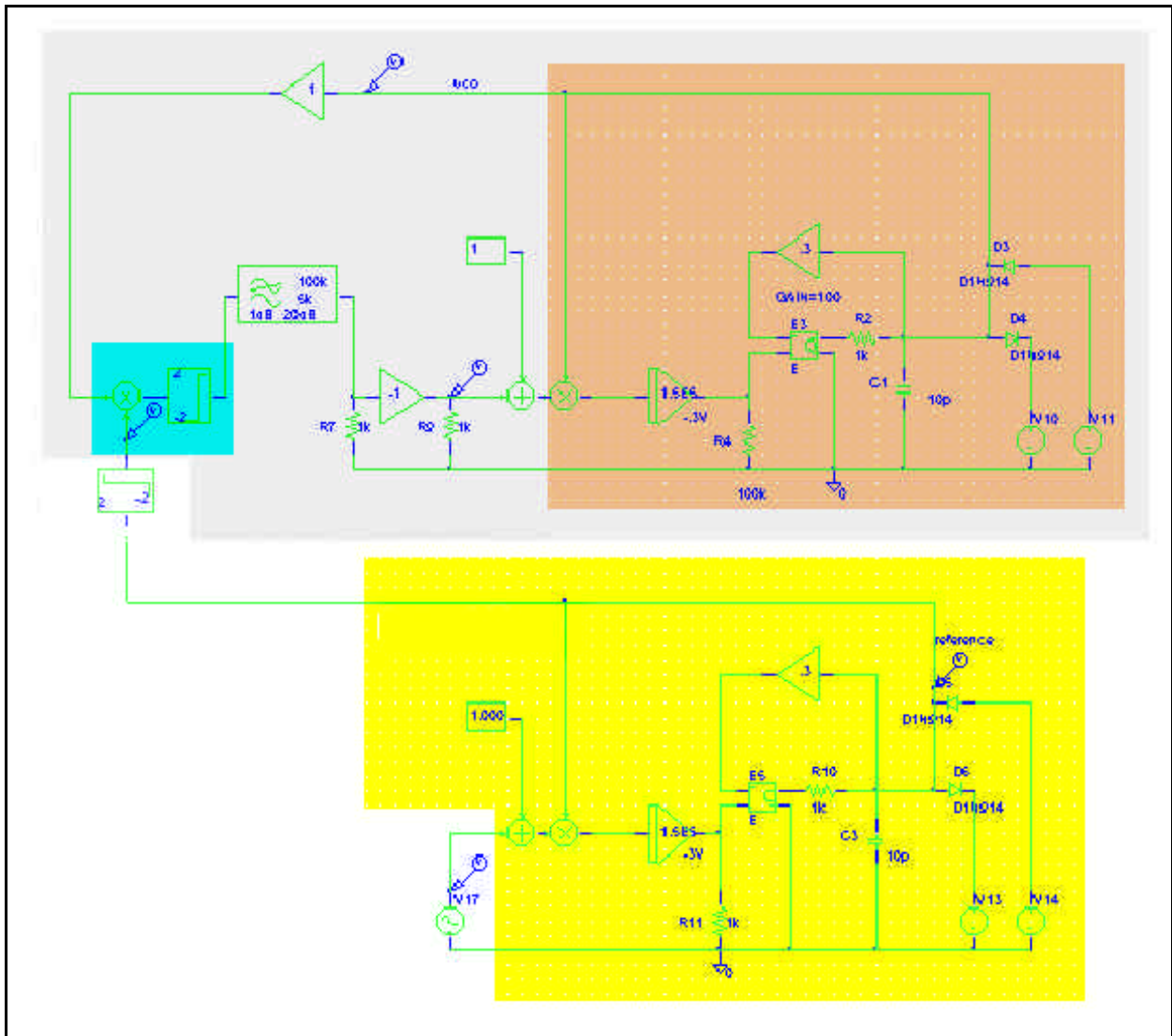
Figure 7: Functional block diagram of phase-locked loop system. The section in yellow serves to modulate the input waveform, (which is a sinusoid, a square wave (pulse), or a triangle input from the voltage source at V17). The orange represents a voltage controlled oscillator
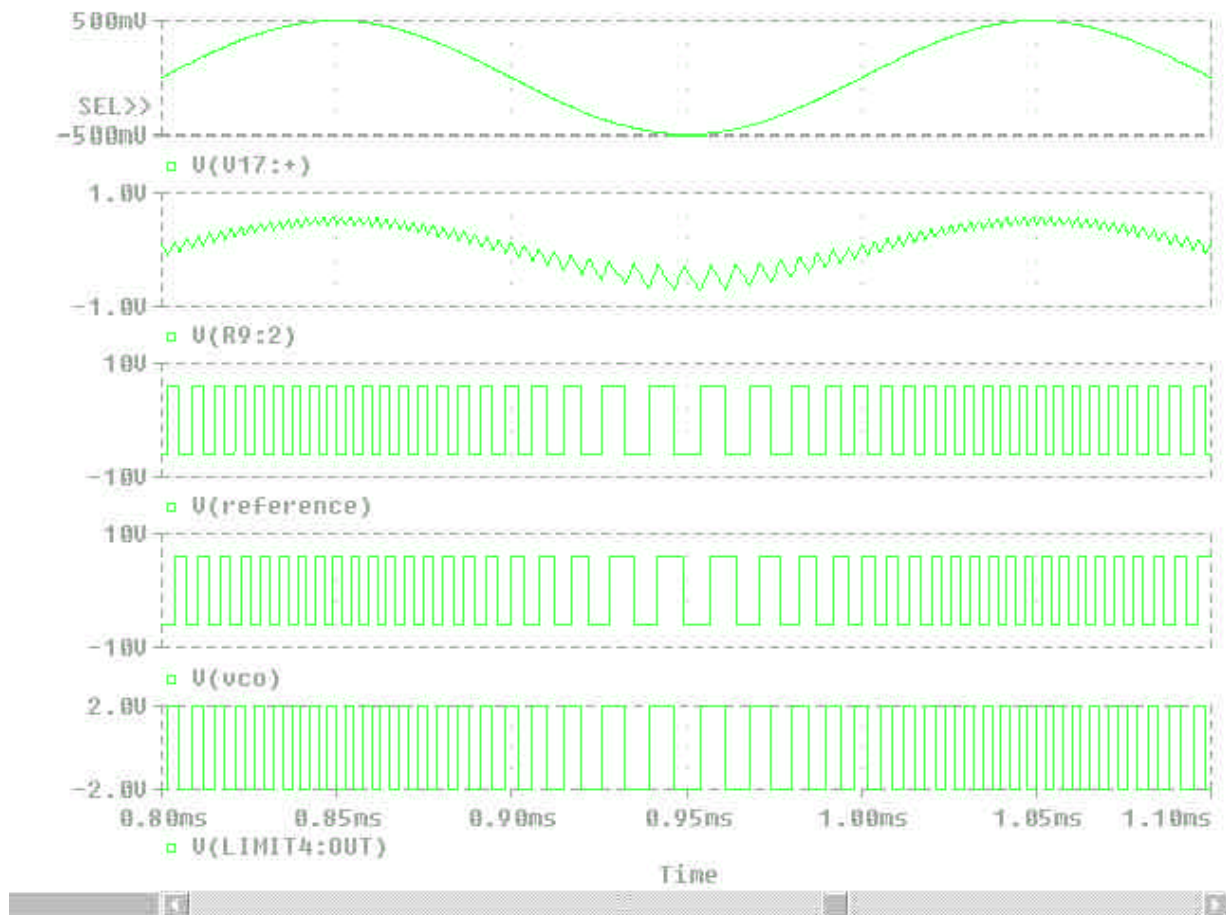
Figure 8: Waveforms associated with the phase-locked loop system. The input waveform is shown at the top (V17:+).

Figure 9: Waveforms associated with the phase-locked loop system. In this case, the input waveform, V(V18:+), is a square wave.
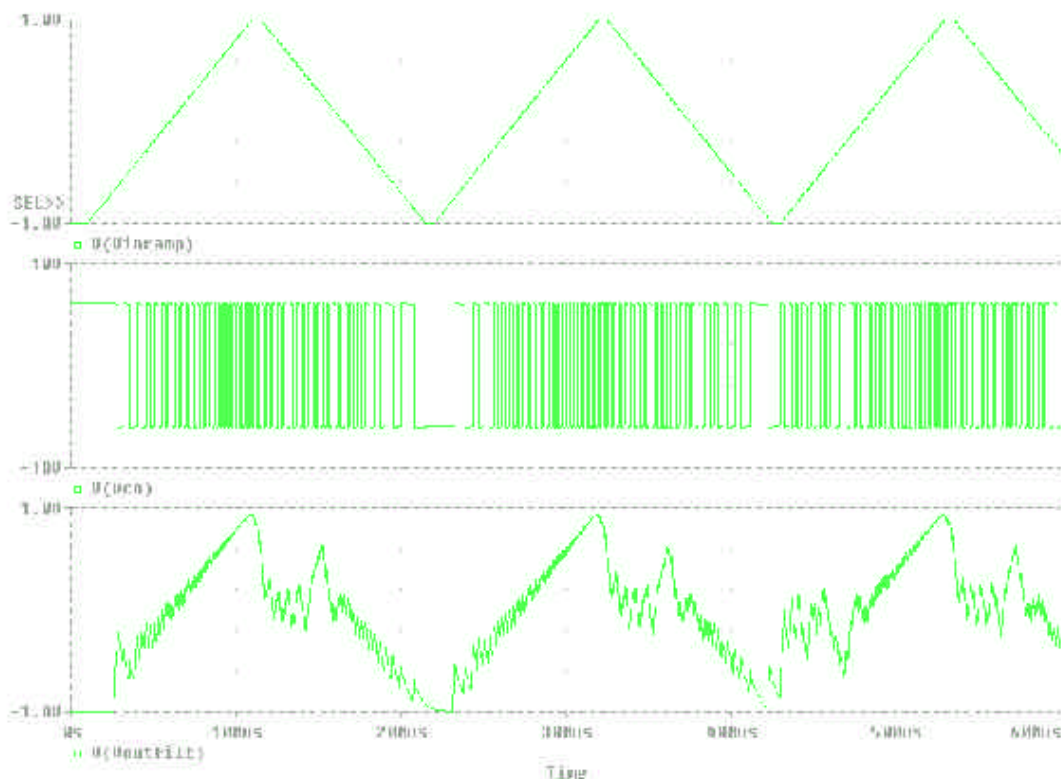
Figure 10: Waveforms associated with the phase-locked loop system. In this case, the input waveform, V(V18:+), is a sawtooth wave. The bandwidth of the filter in this example is set too low.
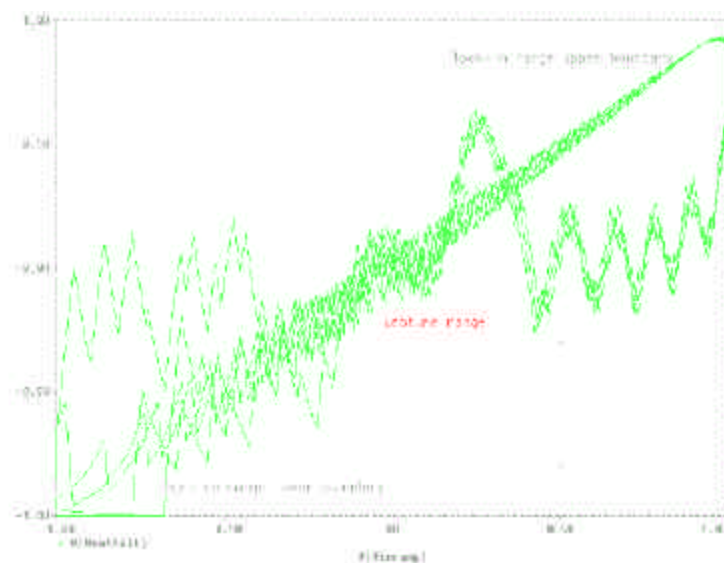
Figure 11: V(Voutfilt) vs. V(Vinramp). The input signal should be translated into frequency of the PLL input signal. Two signals are shown–one output that is able to follow the input signal, and one that is not.
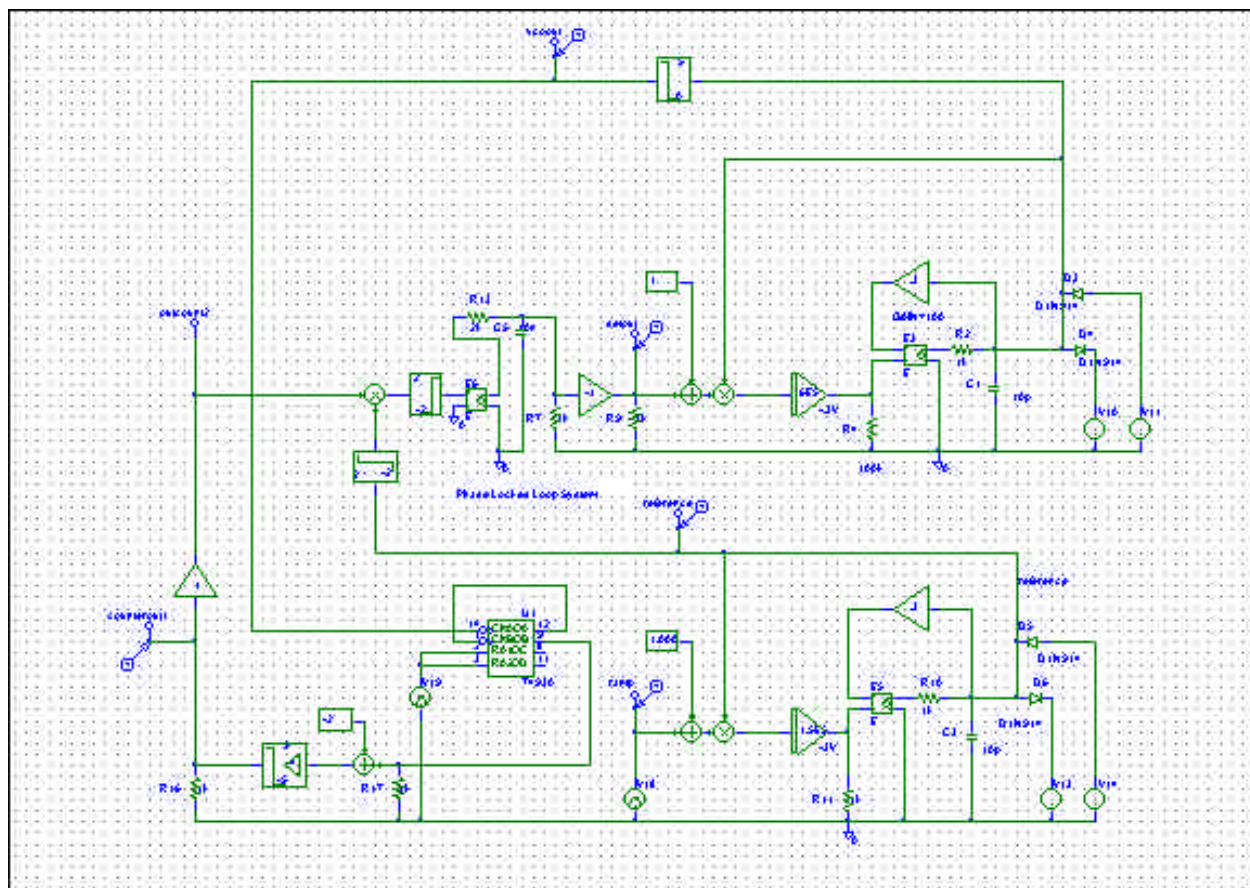
Figure 12: Functional block schematic of the frequency synthesizer.
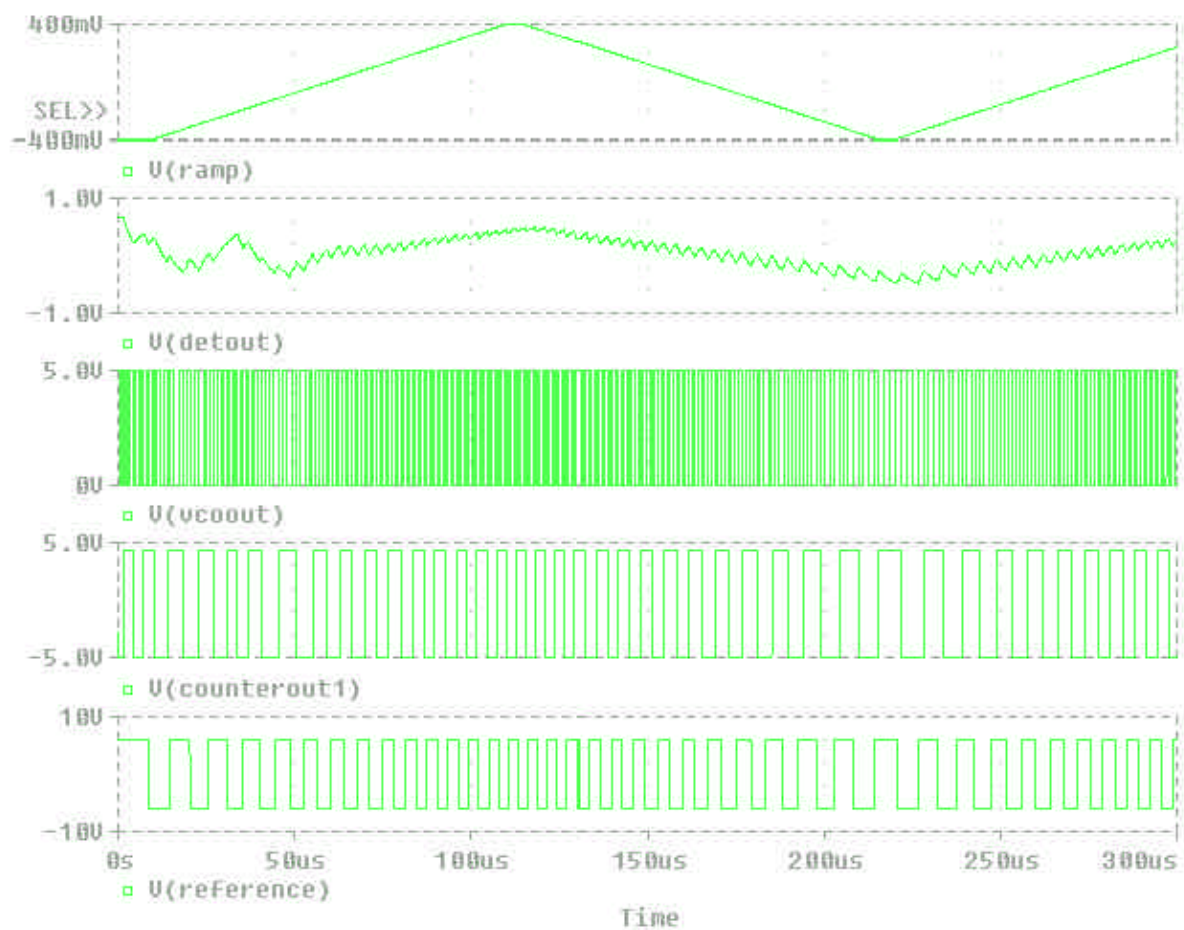
Figure 13: Waveforms associated with the frequency synthesizer.
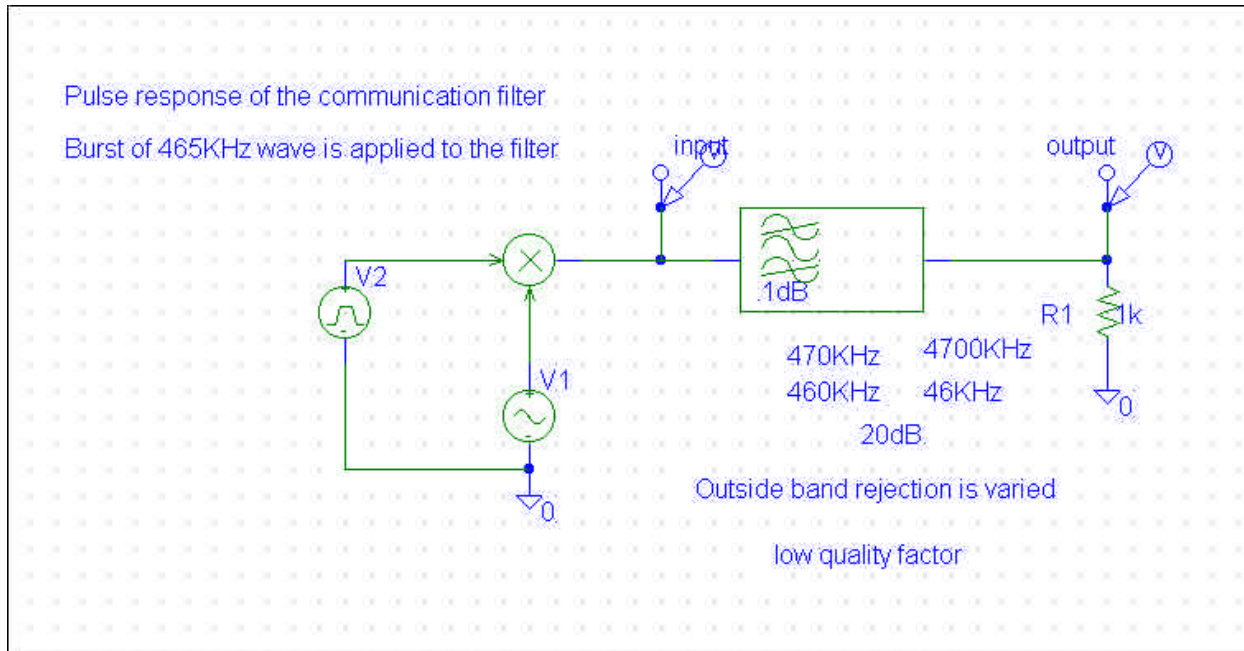
Figure 14: Functional block diagram of a bandpass filter. 20 dB produces a low quality factor. An increase to 60 dB produces a very high quality factor.
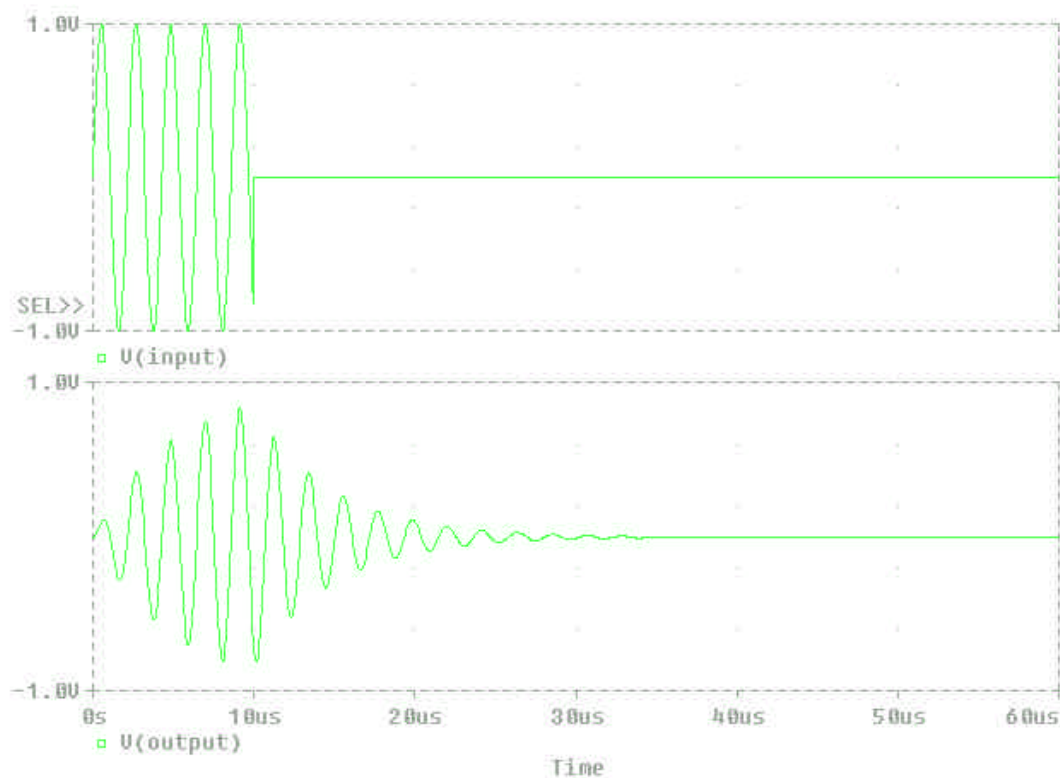
Figure 15: Output versus input waveform with a low quality factor (20 dB) bandpass filter.
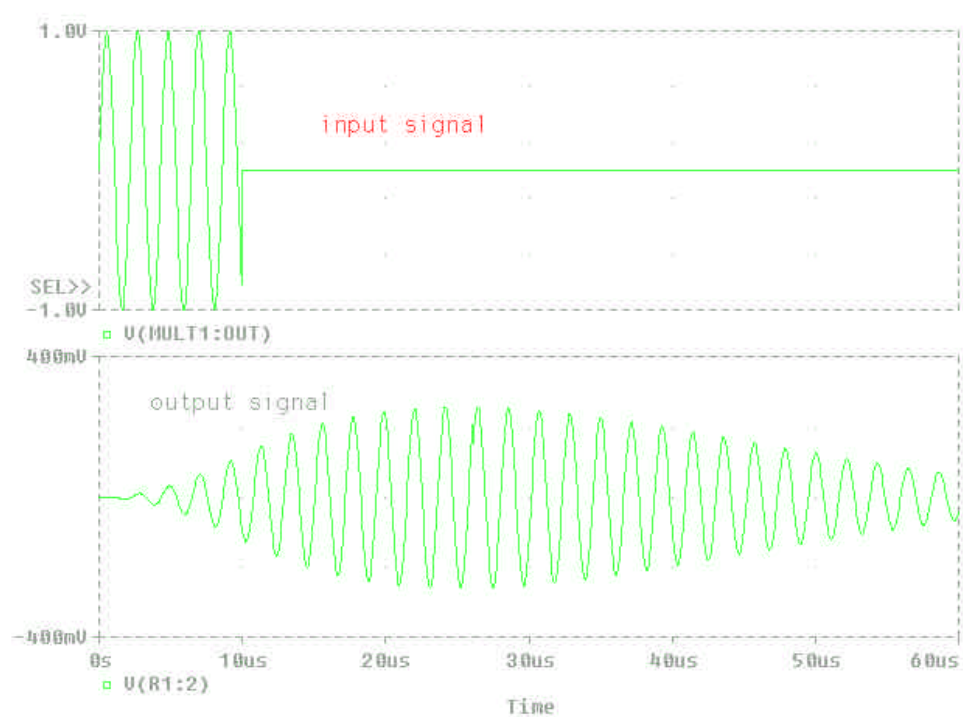
Figure 16: Output versus input waveform with a high quality factor (60 dB) bandpass filter. Notice the delayed response and extended ring time compared with the high quality factor filter.
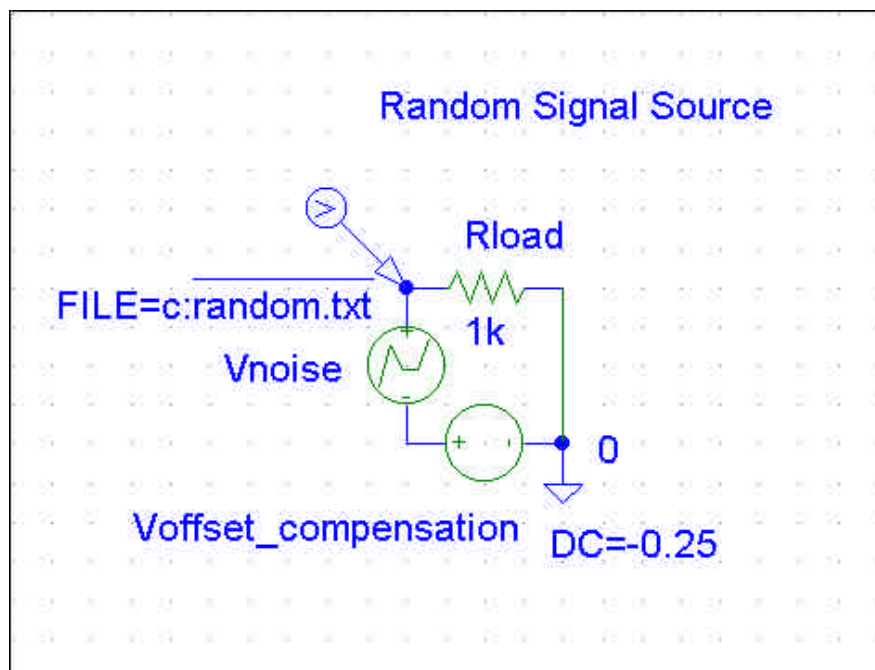
Figure 17: Functional block simulation of a noise generator.
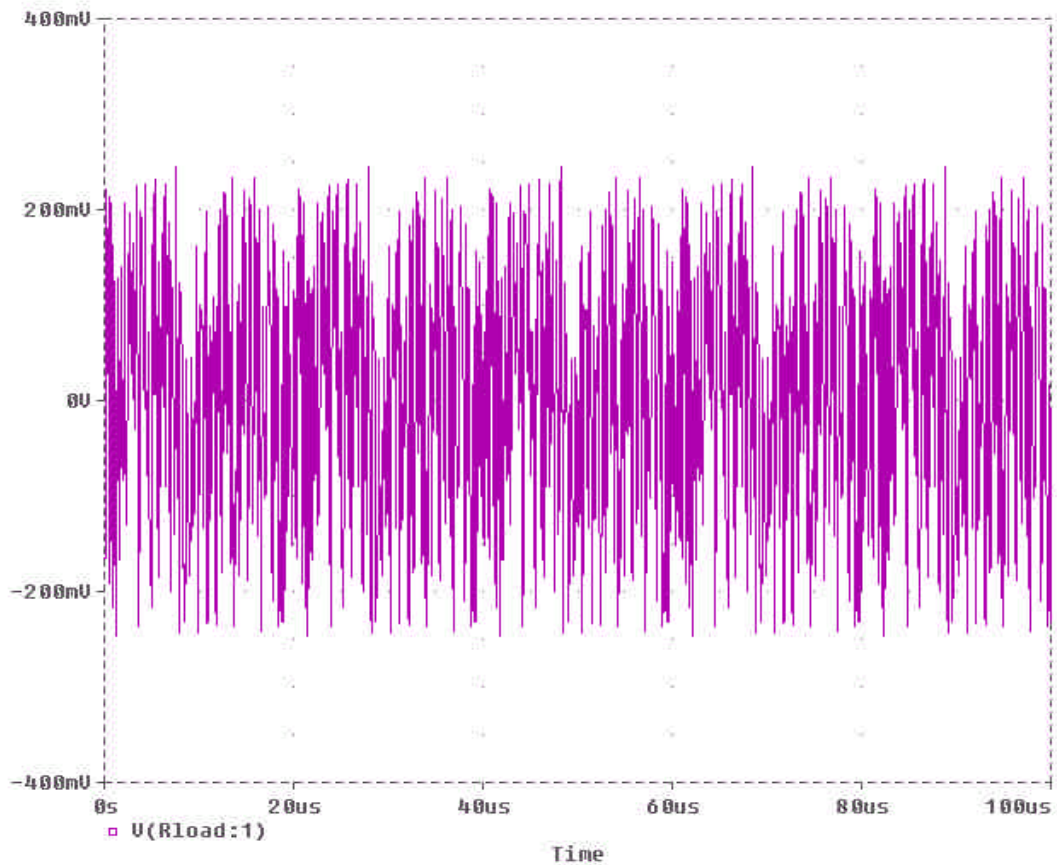
Figure 18: Output waveform of noise generator.
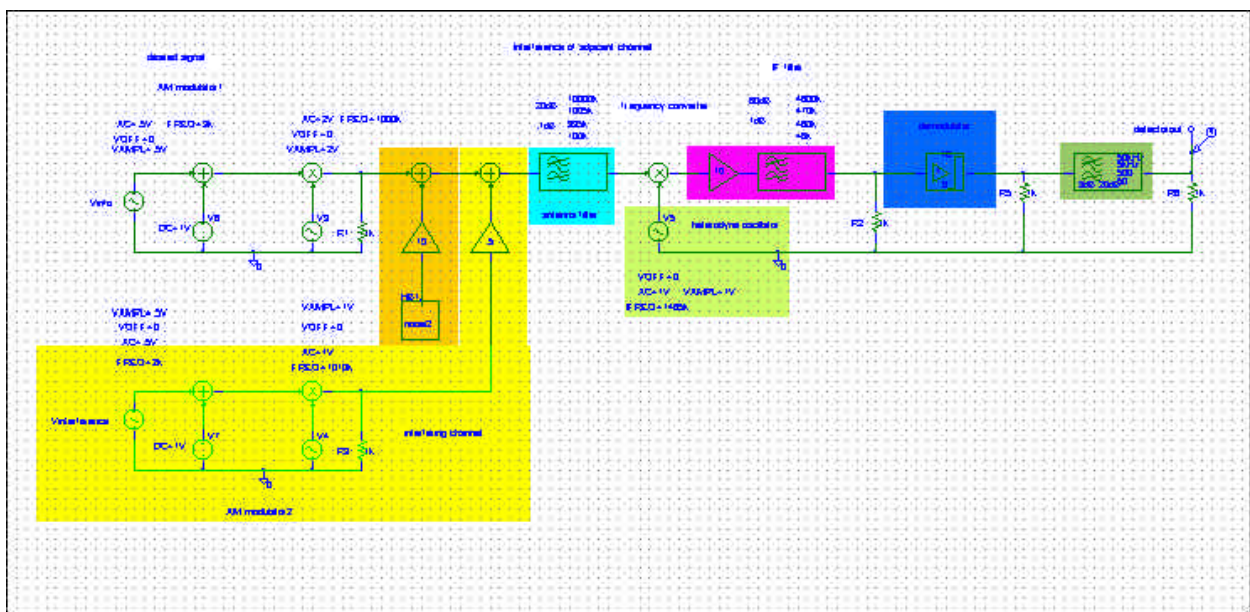
Figure 19: Functional block diagram of AM Modulator/Demodulator.
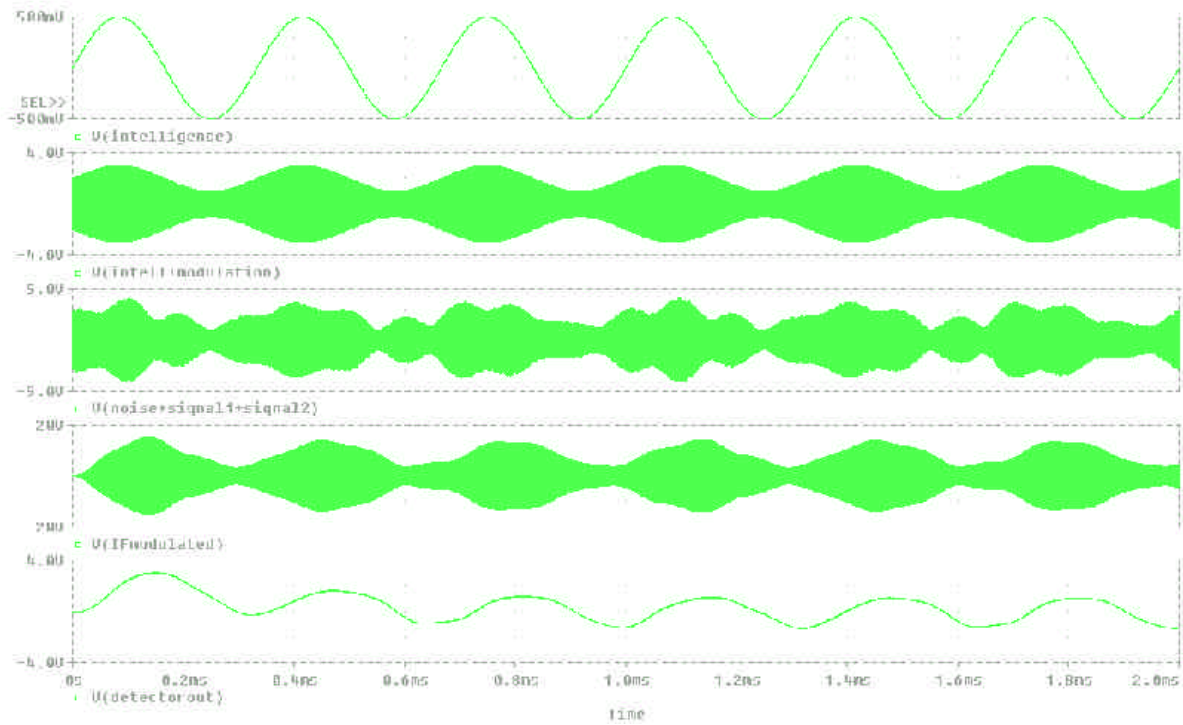
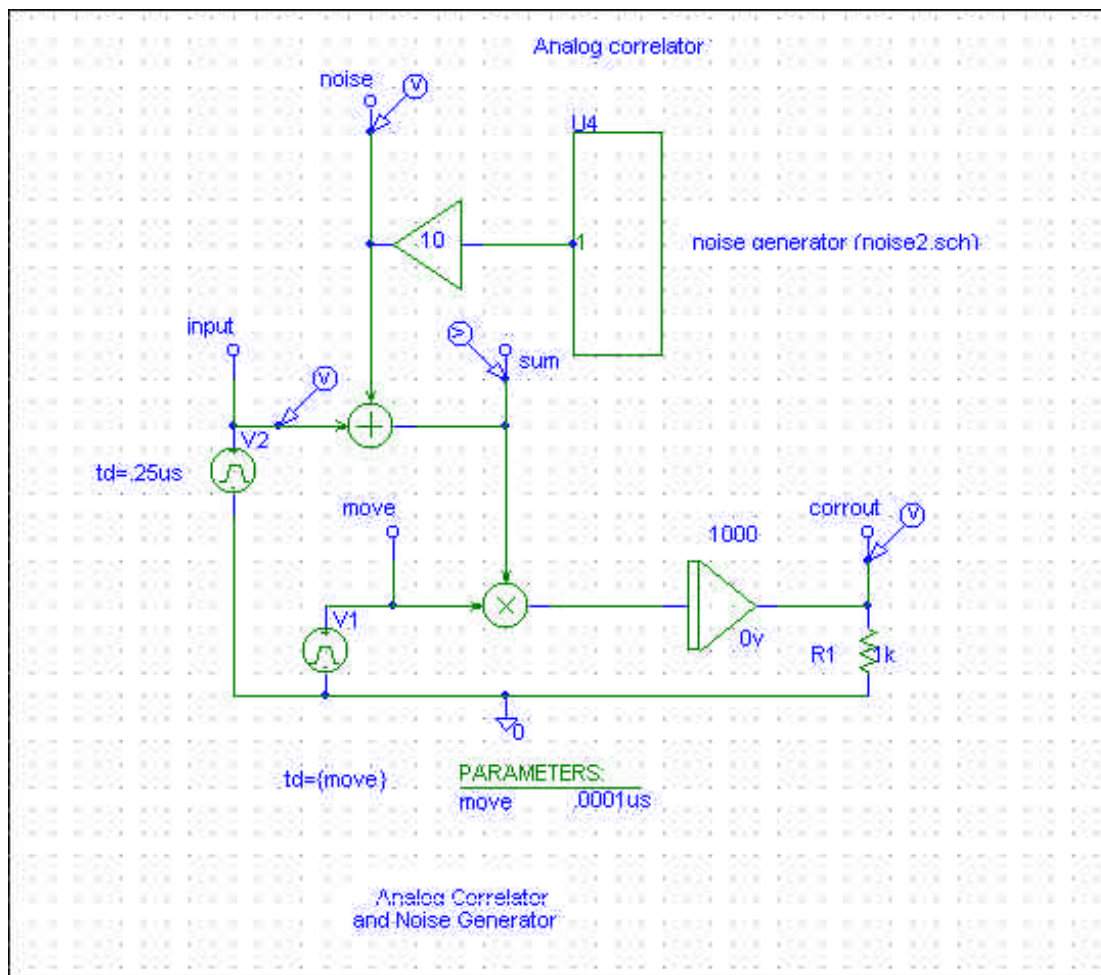**Figure 20.** Waveforms affiliated with the AM Modulator/Demodulator.
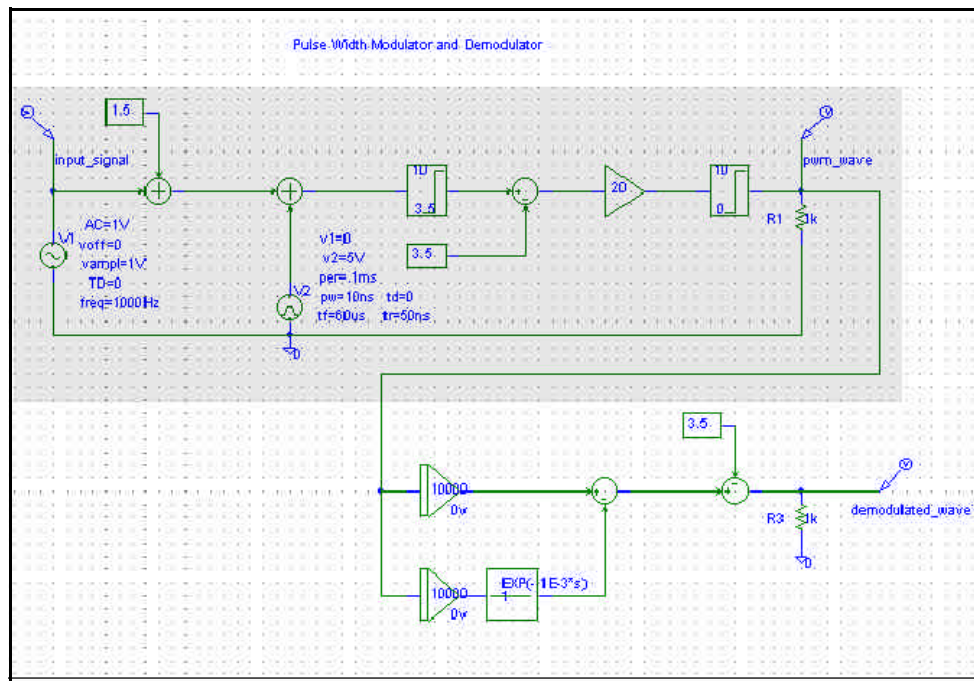
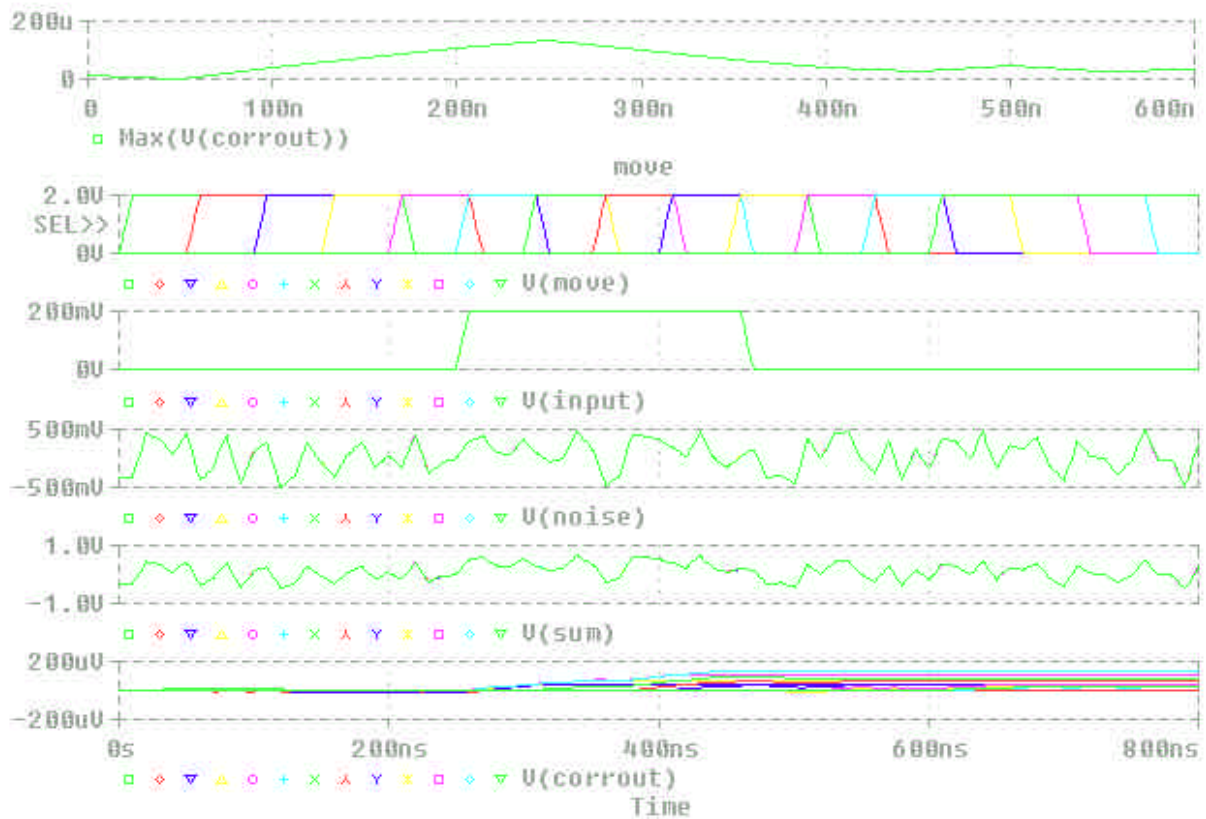**Figure 21:** Functional block diagram of a pulse width modulator (shaded portion) and demodulator.
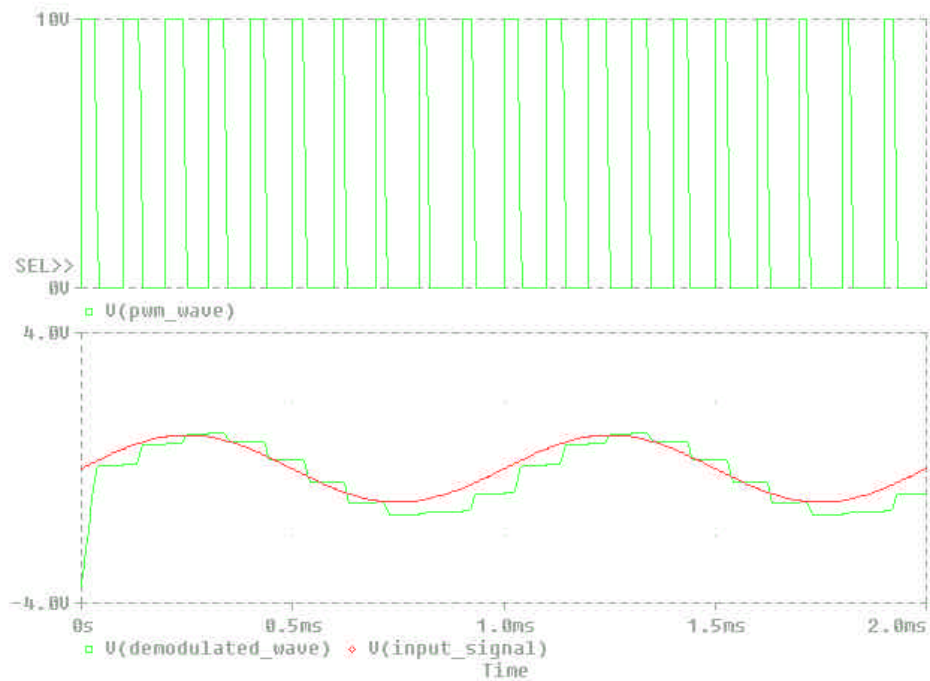
Figure 22: Waveforms affiliated with the functional block diagram of the pulse width modulator and demodulator.
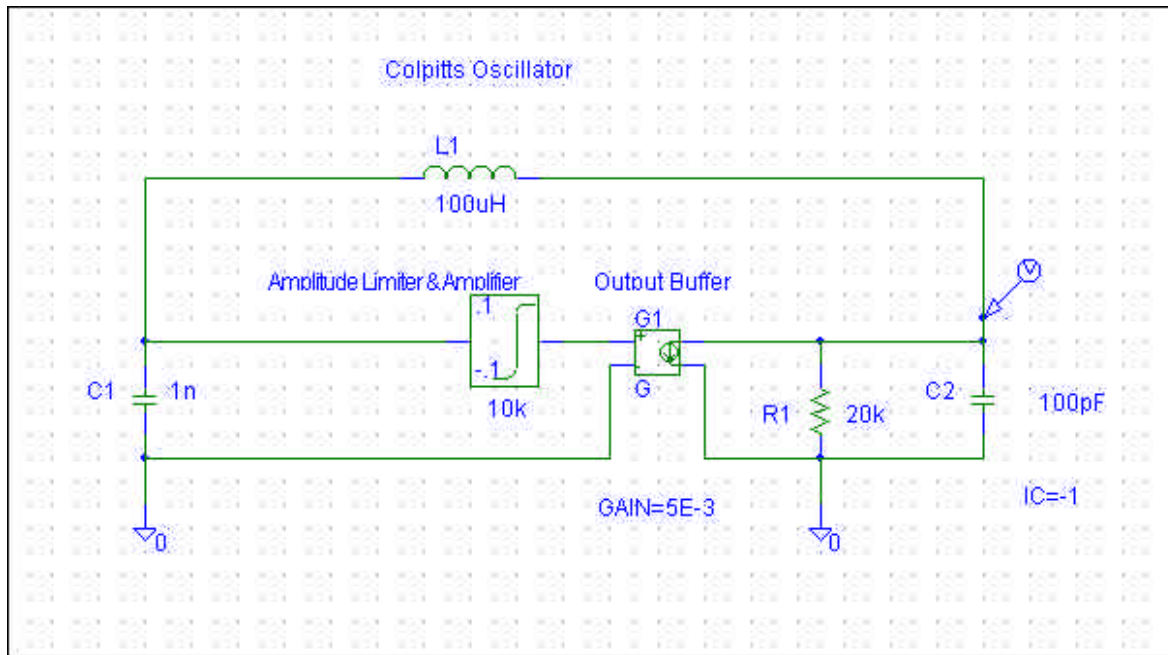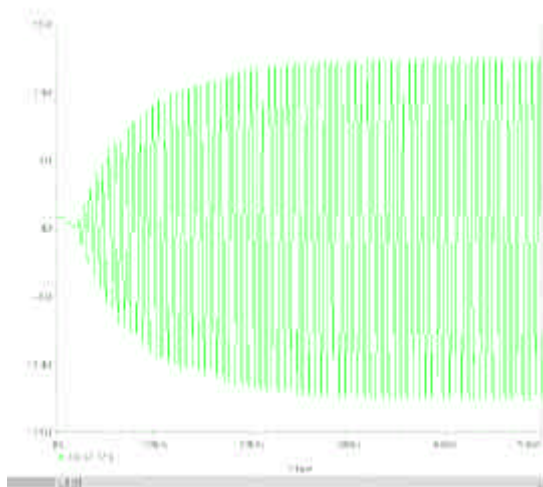
Figure 23: Functional block diagram of analog correlator with noise generator.
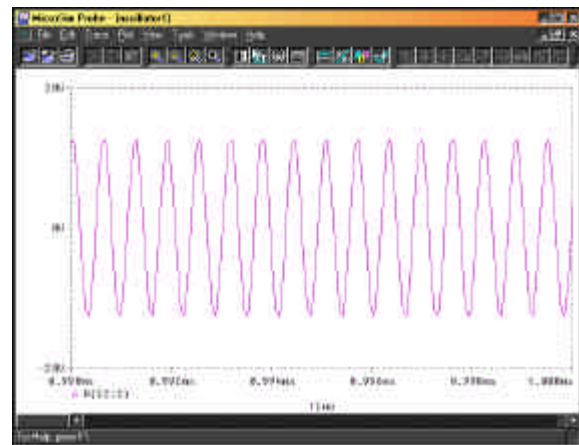
Figure 24: Waveforms affiliated with the analog correlator.

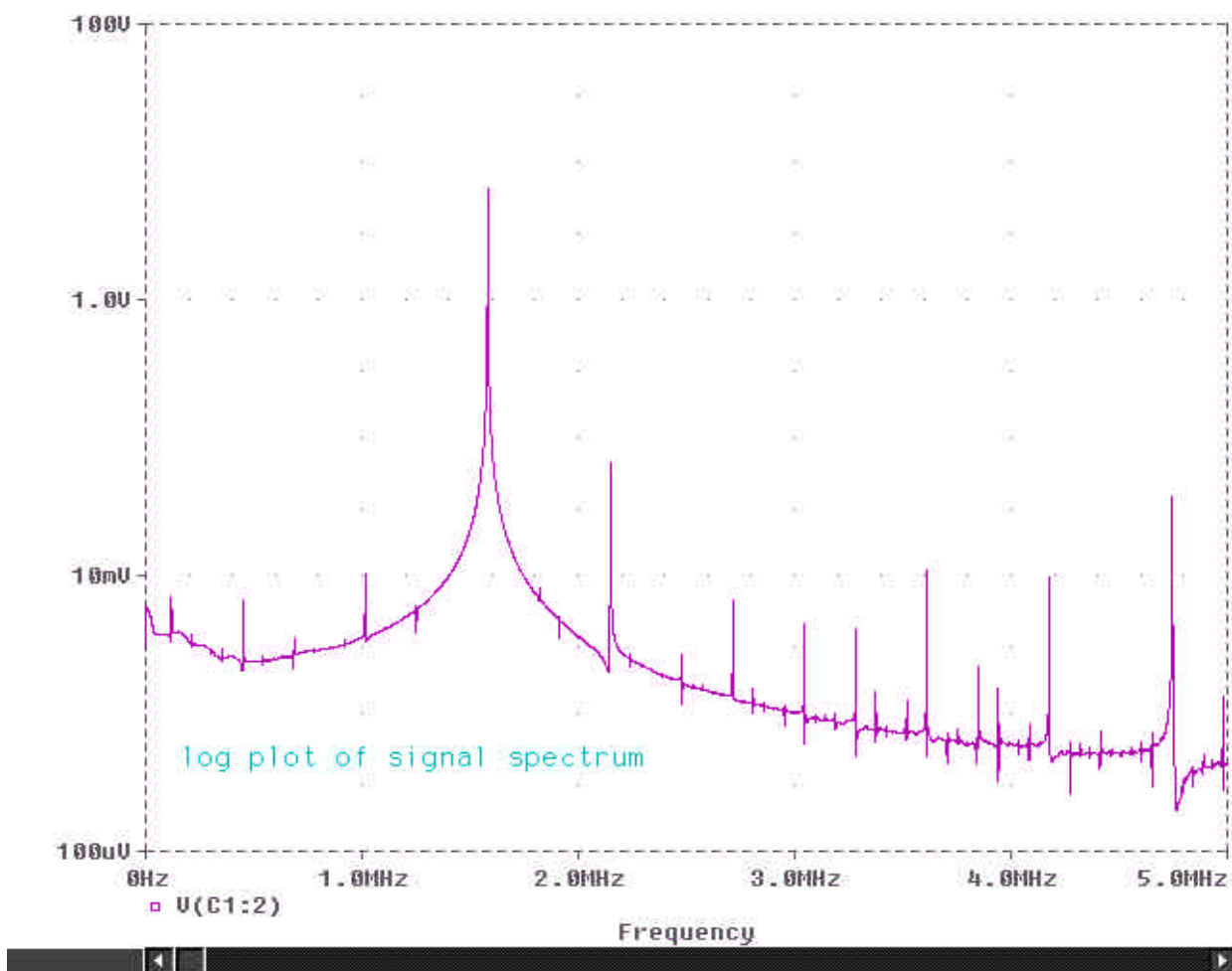**Figure 25.** Functional block diagram of Colpitts oscillator.

Figure 26: Start of voltage oscillations across the Collpitts oscillator.

Figure 27: Oscillations in the Collpitts under steady state conditions.

Figure 28: Frequency spectrum of Colpitts oscillator.

References

1. "PSpice and MATLAB in Undergraduate and Graduate Electrical Engineering Courses," Asad Azemi and Edwin Yaz, Proceedings of the Frontiers in Education Conference, pp. 456-459, November 1-6, 1994.

2. PSpice®- A Critical Thread in Vertical and Horizontal Curriculum Integration," Gopal Mohan, J. Michael Jacob, Proceedings of the American Society of Engineering Education, June 20-23, 1999.

3. Electronics WorkBench® and PSPICE® computer-aided design systems as educational tools for second and fourth-year university courses in Electronics," Martin P. Mintchev, Brent J. Maundy, Proceedings of the American Society of Engineering Education, June 20-23, 1999.

4."A Comparison of Electronic Design and Analysis Packages," Elaine Cooney, Carlos Monsanto, Proceedings of the American Society of Engineering Education, June 18-21, 2000.

5Electronic Communication Techniques (4th Edition), Prentice Hall, 1999, by Paul H. Young.

ANDREW RUSEK
Andrew Rusek is a Professor of Engineering at Oakland University in Rochester, Michigan. He received an M.S. in Electrical Engineering from Warsaw Technical University in 1962, and a PhD. in Electrical Engineering from the same university in 1972. His post-doctoral research involved sampling oscillography, and was completed at Aston University in Birmingham, England, in 1973-74. Dr. Rusek is very actively involved in the automotive industry with research in communication systems, high frequency electronics, and electromagnetic compatibility. He is the recipient of the 1995-96 Oakland University Teaching Excellence Award.

BARBARA OAKLEY
Barbara Oakley is an Assistant Professor of Engineering at Oakland University in Rochester, Michigan. She received her B.A. in Slavic Languages and Literature, as well as a B.S. in Electrical Engineering, from the University of Washington in Seattle. Her M.S. in Electrical and Computer Engineering was received from Oakland University in 1995, and her Ph.D. in Systems Engineering, also from Oakland, was received in 1998. Her research involves biomedical applications and electromagnetic compatibility. She is a recipient of the NSF FIE New Faculty Fellow Award, and was also designated an NSF New Century Scholar.