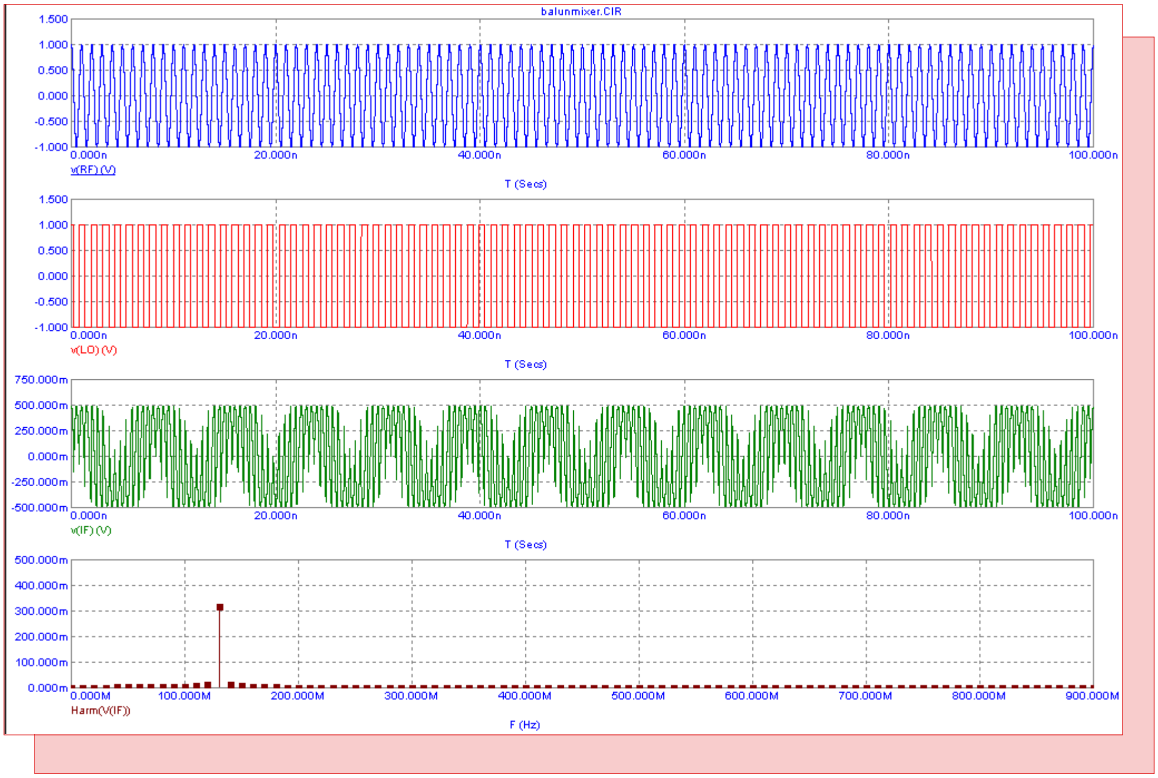
spectrum news

Applications for Micro-Cap™ Users

Fall 2009

News



Balun Macro

Featuring:

- Calculating VSWR, Return Loss, Reflection Coefficient, and Mismatch Loss
- Balun Macro
- Pulse Width Modulator Macro

News In Preview

This newsletter's Q and A section describes how to set parameter values in the schematic for subcircuits that have the PARAMS: keyword defined. The Easily Overlooked Feature section describes the Keep X Scales the Same option which forces all horizontal scales using the same X expression to use the same X scale range.

The first article describes how to calculate VSWR, Return Loss, Reflection Coefficient, and Mismatch Loss in an AC analysis. User functions created through define statements are used for all of these calculations.

The second article describes a balun macro circuit which is a type of transformer that can convert signals between balanced circuitry and unbalanced circuitry.

The third article describes a pulse width modulator (PWM) macro. This modulation method can be used to efficiently provide power for circuitry.

Contents

Book Recommendations

General SPICE

- *Computer-Aided Circuit Analysis Using SPICE*, Walter Banzhaf, Prentice Hall 1989.
 ISBN# 0-13-162579-9

- *Macromodeling with SPICE*, Connelly and Choi, Prentice Hall 1992. ISBN# 0-13-544941-3

- *Inside SPICE-Overcoming the Obstacles of Circuit Simulation*, Ron Kielkowski, McGraw-Hill, 1993. ISBN# 0-07-911525-X

- *The SPICE Book,* Andrei Vladimirescu, John Wiley & Sons, Inc., 1994. ISBN# 0-471-60926-9

MOSFET Modeling

- *MOSFET Models for SPICE Simulation, William Liu, Including BSIM3v3 and BSIM4*, Wiley-Interscience, ISBN# 0-471-39697-4

VLSI Design

- *Introduction to VLSI Circuits and Systems,* John P. Uyemura, John Wiley & Sons Inc, First Edition,
 2002 ISBN# 0-471-12704-3

Micro-Cap - Czech

- *Resime Elektronicke Obvody,* Dalibor Biolek, BEN, First Edition, 2004. ISBN# 80-7300-125-X

Micro-Cap - German

- *Schaltungen erfolgreich simulieren mit Micro-Cap V,* Walter Gunther, Franzis', First Edition, 1997. ISBN# 3-7723-4662-6

Micro-Cap - Finnish

- *Elektroniikkasimulaattori,* Timo Haiko, Werner Soderstrom Osakeyhtio, 2002. ISBN# 951-0-25672-2

Design

- *High Performance Audio Power Amplifiers,* Ben Duncan, Newnes, 1996. ISBN# 0-7506-2629-1

- *Microelectronic Circuits,* Adel Sedra, Kenneth Smith, Fourth Edition, Oxford, 1998

High Power Electronics

- *Power Electronics,* Mohan, Undeland, Robbins, Second Edition, 1995. ISBN# 0-471-58408-8

- *Modern Power Electronics,* Trzynadlowski, 1998. ISBN# 0-471-15303-6

Switched-Mode Power Supply Simulation

- *SMPS Simulation with SPICE 3,* Steven M. Sandler, McGraw Hill, 1997. ISBN# 0-07-913227-8

- *Switch-Mode Power Supplies Spice Simulations and Practical Designs*, Christophe Basso, McGraw-Hill 2008. This book describes many of the SMPS models supplied with Micro-Cap.

Micro-Cap Questions and Answers

Question: I have two of the CoreSat components in my schematic. I would like to set the Bsat and N parameters for each of these components individually. The problem is that when I double click on one of them to open the Attribute dialog box and proceed to edit the parameter values in the text field where the subcircuit model is displayed, it changes those parameters for both CoreSat components. How do I set the parameters for just a single instance of the component?

Answer: In the Attribute dialog box, when a subcircuit model is displayed in the text field, that is the model that all instances of the component will use in that schematic. Any edits to the model in that field will change the model for all instances in the schematic. For a subcircuit, any parameters that can be passed through to the model are specified in the Params: section of the .Subckt header. For the CoreSat model, the subcircuit header is as follows:

```
.subckt coresat L1 L2 110 100 params: Feddy=25k IVSEC=0 Ae=0.000067 lm=0.037
+lg=0 Bsat=350m ur=6000 N=15 Hc=50
```

The Bsat and N parameters are two of the parameters that can be passed through to the subcircuit. In order to modify these for an individual component, the PARAMS: attribute needs to be used. Double click on the component while in Select model. When the Attribute dialog box comes up, select the PARAMS: attribute. For the value of the attribute, enter a string such as the following:

```
Bsat=500m N=10
```

This string will set the Bsat parameter to 500m and the N parameter to 10 for that specific component. The values for the parameters in the subcircuit header are the default values. When the parameter is specified in the PARAMS: attribute, this value will then have the priority for that instance of the component.

Easily Overlooked Features

This section is designed to highlight one or two features per issue that may be overlooked among all the capabilities of Micro-Cap.

Keep X Scales the Same option

The Keep X Scales the Same option forces all horizontal scales using the same X expression to use the same X scale range. This option is available under the Scope menu when in an analysis.

When this option is enabled, drag scaling or panning in one plot group will change the horizontal scale in all other plot groups that have the same X expression. This option is convenient when you want separate plot groups for the waveforms but want the X scale ranges to track each other. This option provides a simple operation to examine the details of multiple waveforms over a specific time range rather then having to scale each plot group separately.

In the plot below, the initial simulation was run for 1ms. In order to closely examine the waveforms, the top plot group was scaled with the mouse to show a narrow range of the waveform from approximately 162us to 299us. Once the top plot rescales, all of the other plot groups also rescale to this new X range.

While the scaling operation in the plot can also affect the Y range for that plot group, the other plot groups will maintain the Y ranges that were previously defined for them.

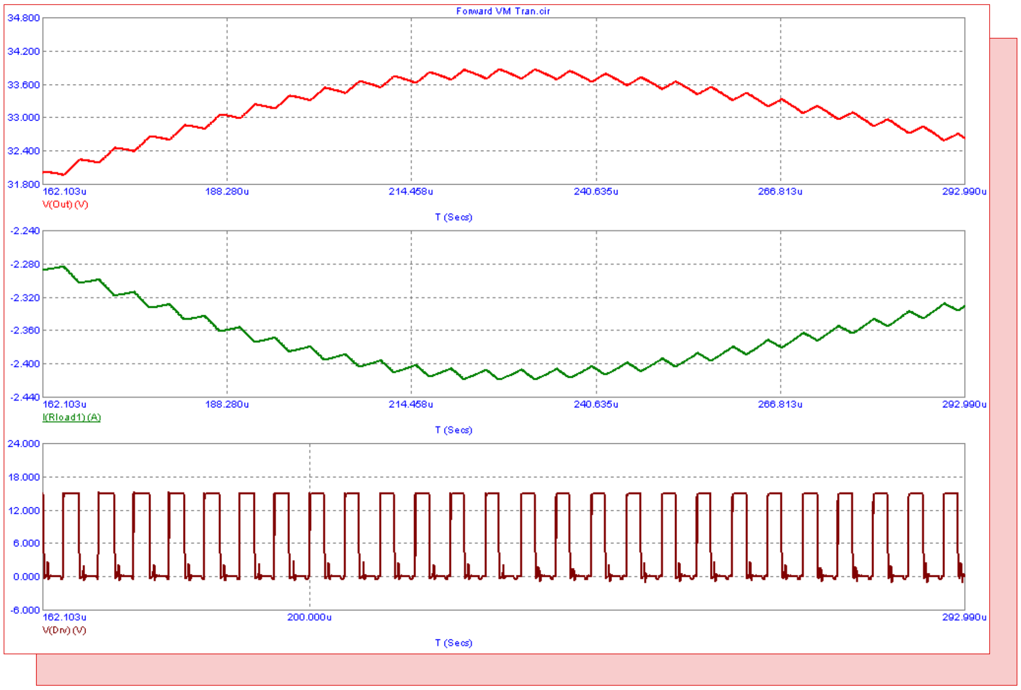


Fig. 1 - Keep X Scales the Same option

Calculating VSWR, Return Loss, Reflection Coefficient, and Mismatch Loss

There are a number of calculations that are useful when simulating the transmission of a wave through a line. These calculations can be quite important in calculating the energy that arrives at the load versus how much energy the transmitter is producing. Ideally, the load impedance should match the characteristic impedance of the transmission line so that all of the transmitted energy is available at the load. When the load impedance does not match the characteristic impedance of the transmission line, part of the voltage will be reflected back down the line reducing the available energy at the load.

One measurement is the reflection coefficient (Γ). The reflection coefficient measures the amplitude of the reflected wave versus the amplitude of the incident wave. The expression for calculating the reflection coefficient is as follows:

$\Gamma = (Z_L - Z_S)/(Z_L + Z_S)$

where Z_L is the load impedance and Z_S is the source impedance. Since the impedances may not be explicitly known, the reflection coefficient can be measured in a similar manner to an S11 measurement by using the wave amplitudes at the source and at the node following the source impedance. The following define statement user function can be used to measure the reflection coefficient.

.define RefCo(In,Src) Mag(2*V(In)-V(Src))

where In is the node name of the node following the source impedance and Src is the part name of the source component.

The VSWR (Voltage Standing Wave Ratio) measurement describes the voltage standing wave pattern that is present in the transmission line due to the phase addition and subtraction of the incident and reflected waves. The ratio is defined by the maximum standing wave amplitude versus the minimum standing wave amplitude. The VSWR can be calculated from the reflection coefficient with the equation:

$\mathrm{VSWR} = (1 + \Gamma)/(1 - \Gamma)$

The following define statement user function can be used to measure the VSWR.

.define VSWR(In,Src) (1+RefCo(In,Src))/(1-RefCo(In,Src))

The return loss measurement describes the ratio of the power in the reflected wave to the power in the incident wave in units of decibels. The standard output for the return loss is a positive value, so a large return loss value actually means that the power in the reflected wave is small compared to the power in the incident wave and indicates a better impedance match. The return loss can be calculated from the reflection coefficient with the equation:

$\text{Return Loss} = -20*\mathrm{Log}(\Gamma)$

The following define statement user function can be used to measure the return loss.

.define RetLoss(In,Src) -20*Log(RefCo(In,Src))

The mismatch loss measurement describes the amount of power that will not be available at the load due to the reflected wave in units of decibels. It indicates the amount of power lost in the system due to the mismatched impedances. The mismatch loss can also be calculated from the reflection coefficient with the following equation:

Mismatch Loss = -10*Log(1 - Γ^2)

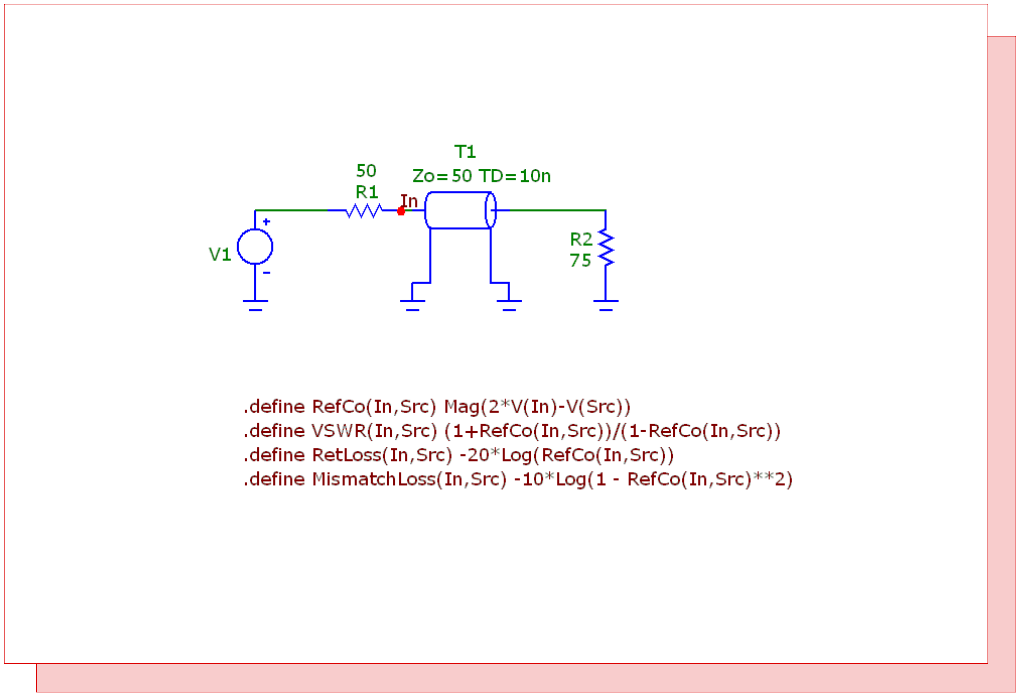
The following define statement user function can be used to measure the mismatch loss.

.define MismatchLoss(In,Src) -10*Log(1 - RefCo(In,Src)**2)

For the VSWR, return loss, and mismatch calculations, the In and Src parameters are defined in the same manner as they are for the reflection coefficient define statement.

If only the VSWR, return loss, or mismatch loss measurement is to be performed in the analysis, the reflection coefficient define statement must also be present to perform the calculation since it is referenced in all three of these calculations.

A simple circuit is displayed in the figure below to demonstrate the use of these define statement user functions. The circuit consists of a voltage source, two resistors, and an ideal, lossless transmission line. The load resistance has been set to 75 ohms to create a mismatch with the 50 ohm characteristic impedance of the transmission line. The four define statement user functions have been entered in the schematic as grid text. Each statement must be entered as a separate grid text. They may also be entered in the Text page of the schematic to reduce the clutter in the schematic.



```
.define RefCo(In,Src) Mag(2*V(In)-V(Src))
.define VSWR(In,Src) (1+RefCo(In,Src))/(1-RefCo(In,Src))
.define RetLoss(In,Src) -20*Log(RefCo(In,Src))
.define MismatchLoss(In,Src) -10*Log(1 - RefCo(In,Src)**2)
```

Fig. 2 - Mismatched transmission line example

An AC analysis simulation is then run on the circuit. The four Y expressions plotted for the simulation are:

VSWR(In,V1)
RetLoss(In,V1)
RefCo(In,V1)
MismatchLoss(In,V1)

The AC simulation results are displayed below. Since this example circuit is entirely resistive, the AC analysis response will be constant across the entire frequency range. Note that the node name used as the parameter within these functions does not have to be named In. It can be any name that the user chooses or even the node number of the node. V1 is the part name for the voltage source in the schematic.

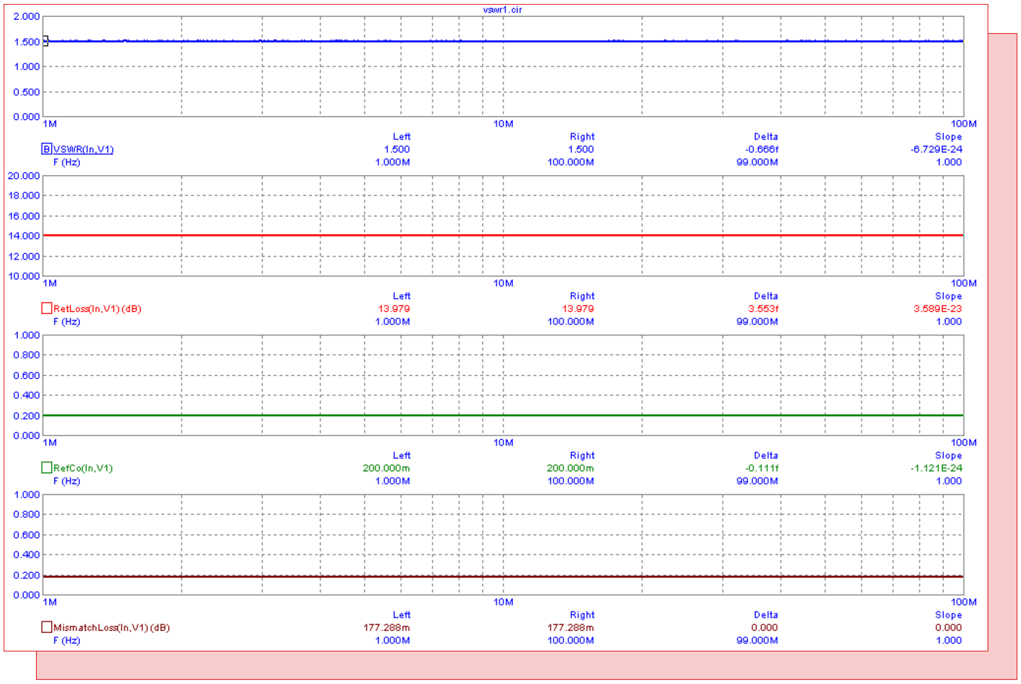


Fig. 3 - Mismatched transmission line AC analysis simulation

The AC simulation returns the following results for this circuit:

VSWR = 1.5
Reflection Coefficient = .2
Return Loss = 13.979dB
Mismatch Loss = .177dB

The define statements can also be placed in the MCAP.INC file which can be accessed through the User Definitions under the Options menu. Placing them in this file makes the functions globally available for all circuits.

Reference:

1) VOLTAGE STANDING WAVE RATIO (VSWR) / REFLECTION COEFFICIENT RETURN LOSS / MISMATCH LOSS, Granite Island Group, http://www.tscm.com/vswr.pdf

Balun Macro

The balun device is a type of transformer that can convert signals between balanced circuitry and unbalanced circuitry. It is a bidirectional device so the signal conversion can go in either direction. The balun is often used in communications and RF to provide compatibility between systems. A balun can be created in Micro-Cap by combining two ideal transformers. The balun macro circuit below was created based on a Ken Kundert model described in Reference 1.

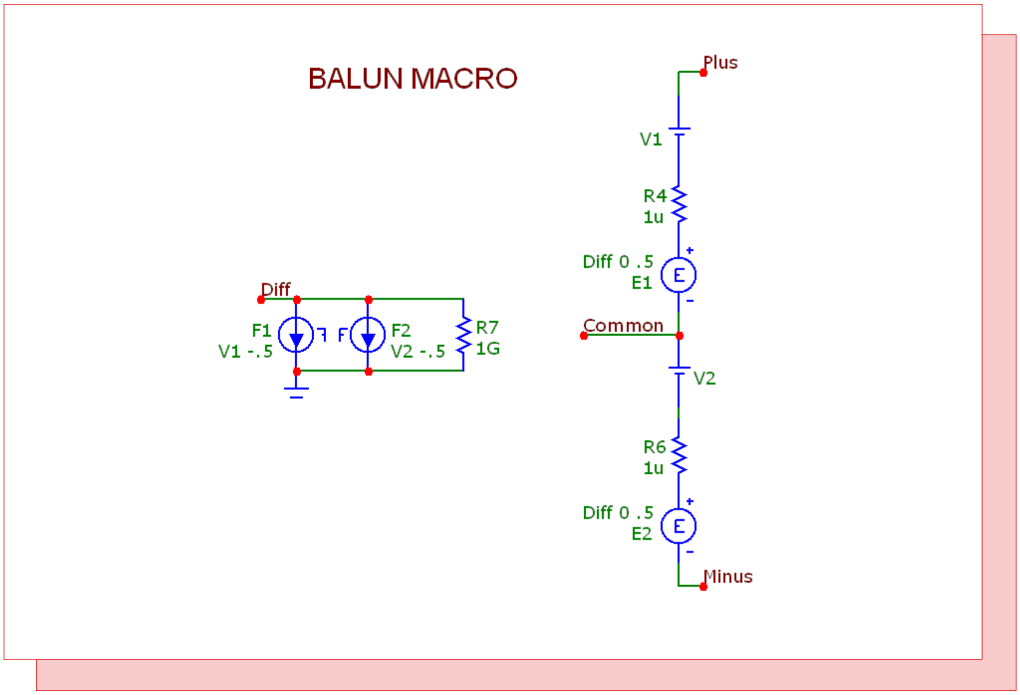


Fig. 4 - Balun macro circuit

The macro circuit has four pins. The Diff pin is for the differential mode unbalanced signal. The Common pin is for the common mode unbalanced signal. The Plus and Minus pins are for the balanced positive and negative signals.

The macro circuit combines two ideal transformers. The first transformer consists of the F1 FIofI dependent source, E1 EVofV dependent source, and the V1 battery. The V1 battery is used to measure the current into the Plus pin of the balun. The F1 component senses the current through V1 and outputs a current of -.5*I(V1). The E1 component measures the voltage at the Diff pin and outputs a voltage of .5*V(Diff). This transformer is setup to provide a voltage gain of .5 from the Diff pin to the Plus pin. The second transformer consists of the F2, E2, and V2 components and operates in the same manner as the first transformer. However, the polarity on the balanced side of this transformer is reversed in order to create the negative signal at the Minus pin. Since the macro is created from ideal components, the balun will work across all frequencies. The voltage relationships for the balun are as follows:

V(Diff) = V(Plus) - V(Minus)
V(Common) = .5*(V(Plus) + V(Minus))
V(Plus) = V(Common) + .5*V(Diff)
V(Minus) = V(Common) - .5*V(Diff)

The corresponding current relationships are:

I(Diff) = .5*(I(Plus) - I(Minus))
I(Common) = I(Plus) + I(Minus)
I(Plus) = .5*I(Common) + I(Diff)
I(Minus) = .5*I(Common) - I(Diff)

The 1G ohm resistor at the Diff pin provides a DC path to ground at that node. The two 1u ohm resistors on the balanced side of the balun prevent any possible voltage source/inductor loops that may occur when the macro is used in a schematic.

One application that baluns can be used for is in a double balanced mixer circuit. The double balanced mixer is often used in microwave systems to down convert the frequency of an RF signal in order to process it. The RF signal is combined with a local oscillator signal in order to output an intermediate frequency signal. An example of a double balanced mixer is shown below.

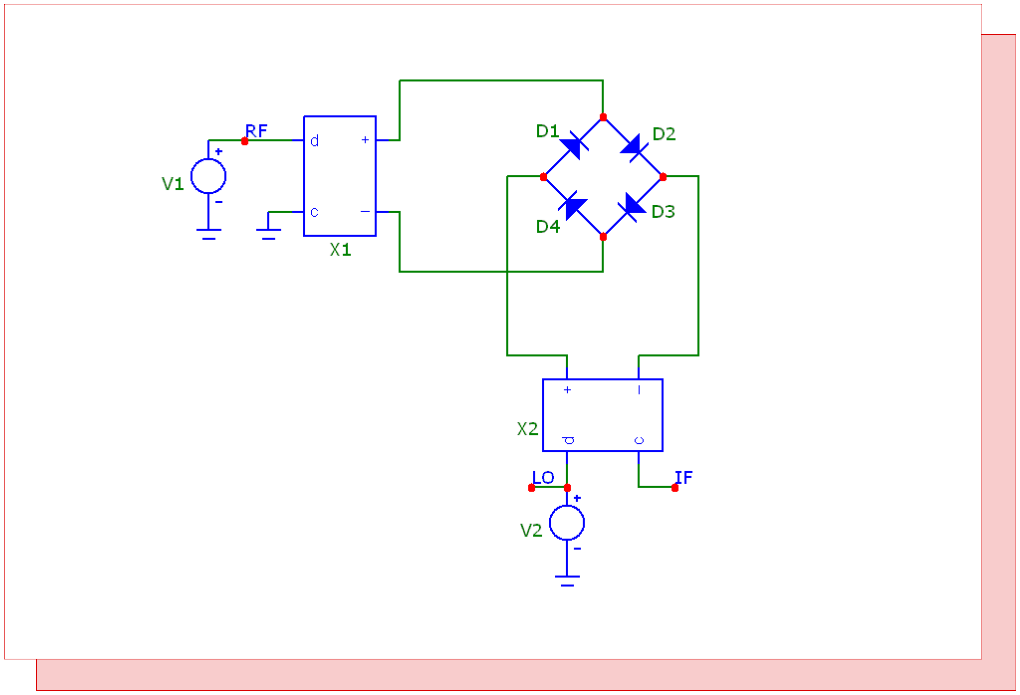


Fig. 5 - Double balanced mixer circuit

The X1 balun has a 1GHz sine wave at its unbalanced pin to represent the RF signal. The balanced output is connected across two nodes of a diode bridge. The X2 balun has an 870 MHz square wave as its unbalanced signal which represents the local oscillator signal. The balanced outputs of this balun connect across the other two nodes of the diode bridge. The Common pin of the X2 balun produces the intermediate frequency signal. The combination of the RF and oscillator signals should produce an intermediate frequency signal of 130MHz.

The mixer circuit is run in transient analysis with a simulation time of 100ns. The resulting simulation is shown in Figure 6.

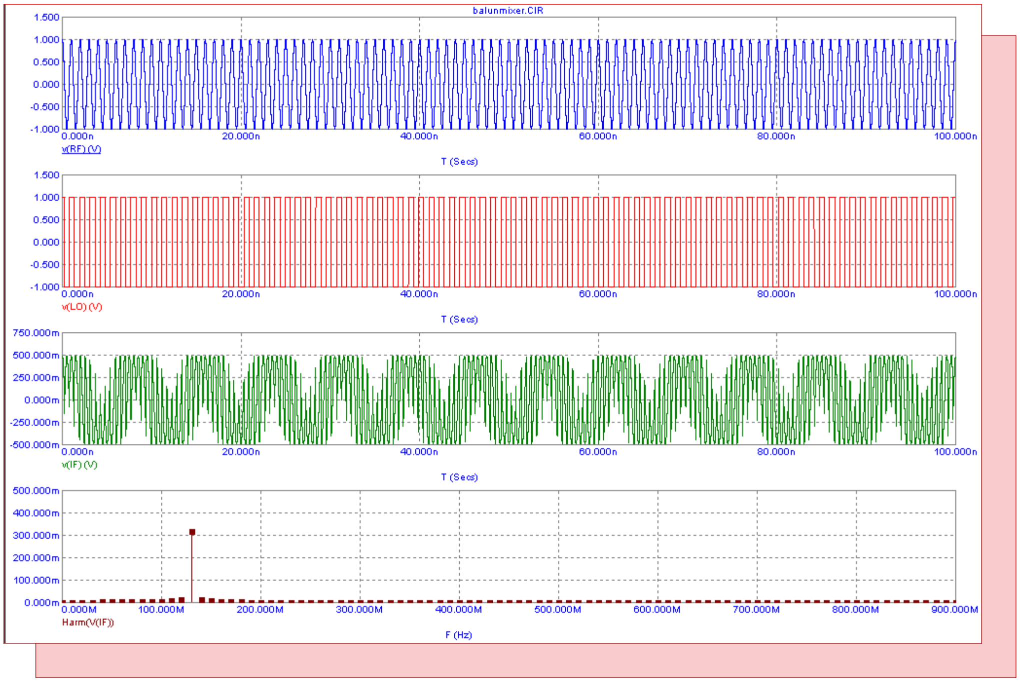


Fig. 6 - Double balanced mixer simulation

The top plot displays the 1GHz sine wave at the RF node. The second plot displays the 870MHz square wave at the LO node. The third plot shows the intermediate frequency waveform that is created from the other signals at the IF node. A low frequency sinusoidal waveform can be seen in the signal. The bottom plot displays the harmonics of the intermediate frequency waveform by plotting the expression:

Harm(V(IF))

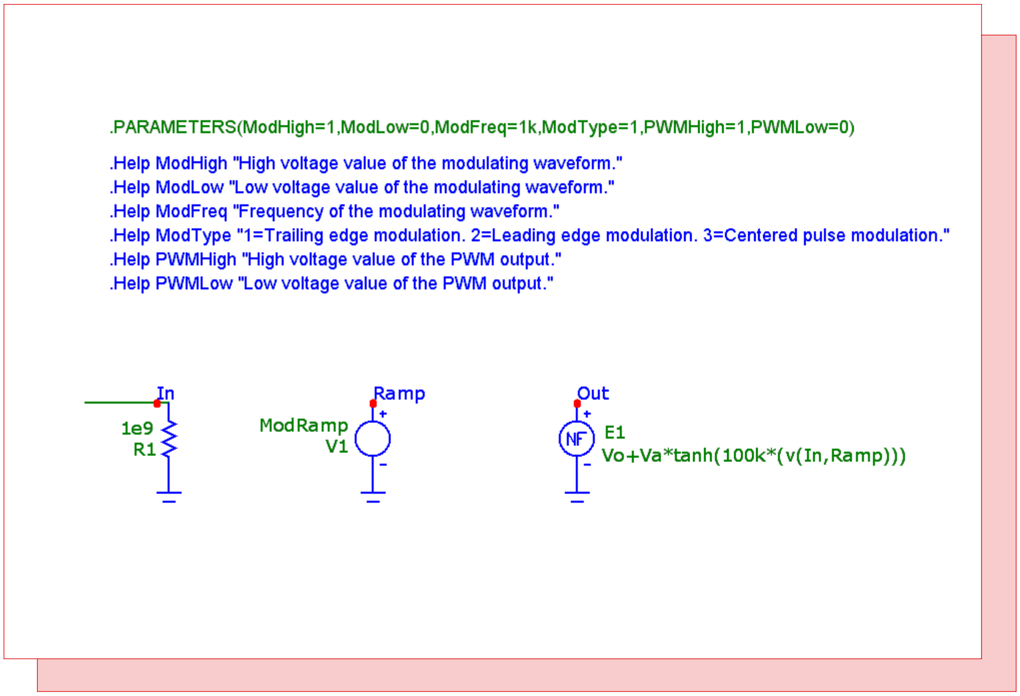
As expected, the harmonic plot shows that the fundamental frequency of the intermediate frequency waveform comes out at 130MHz.

References:

1) A Test Bench for Differential Circuits, Ken Kundert, Designer's Guide Consulting, Inc., http://www.designers-guide.org/Analysis/diff.pdf
2) Double-balanced mixers, Microwaves101.com, http://www.microwaves101.com/encyclopedia/mixersdoublebalanced.cfm

Pulse Width Modulator Macro

One method to efficiently provide power for circuitry is through pulse width modulation (PWM). PWM switches the power on and off in order to reduce the amount of power that is lost in the load. This modulation technique produces a series of pulses whose duty cycle varies depending upon a reference signal. A low duty cycle corresponds to when the power is off most of the time, while a high duty cycle corresponds to when the power is on most of the time. The modulation signal used is typically a sawtooth or a triangle waveform. One method to producing a PWM signal is the intersective method. When the reference signal is greater than the modulation signal, the output is high. When the reference signal is less than the modulation signal, the output is low. A PWM macro circuit using the intersective method appears in the figure below.



```
.PARAMETERS(ModHigh=1,ModLow=0,ModFreq=1k,ModType=1,PWMHigh=1,PWMLow=0)

.Help ModHigh "High voltage value of the modulating waveform."
.Help ModLow "Low voltage value of the modulating waveform."
.Help ModFreq "Frequency of the modulating waveform."
.Help ModType "1=Trailing edge modulation. 2=Leading edge modulation. 3=Centered pulse modulation."
.Help PWMHigh "High voltage value of the PWM output."
.Help PWMLow "Low voltage value of the PWM output."
```

Fig. 7 - PWM macro circuit

The macro circuit has two pins. The In pin is the input pin for the reference signal. The Out pin is the output pin for the PWM signal. The modulation signal is generated within the macro circuit. The macro has six parameters: ModHigh, ModLow, ModFreq, ModType, PWMHigh, and PWMLow. ModHigh and ModLow define the high and low voltage values of the modulating signal. The ModFreq parameter defines the frequency of the modulating signal. ModType defines whether the modulating signal is a trailing edge sawtooth, a leading edge sawtooth, or a triangle waveform. The PWMHigh and PWMLow parameters define the high and low voltage values of the PWM output signal.

The R1 resistor provides a DC path to ground at the input node. The V1 voltage source creates the modulation signal. Its VALUE attribute is defined with the symbolic variable ModRamp which is set through the following series of define statements in the Text page of the macro circuit.

```
.define ModPeriod (1/ModFreq)
.define ModTrans (ModPeriod/10000)
```

```
.if ModType==1
.define ModRamp Pulse ModLow ModHigh 0 {ModPeriod - ModTrans} {ModTrans} 0
+ {ModPeriod}
.elif ModType==2
.define ModRamp Pulse ModHigh ModLow 0 {ModPeriod - ModTrans} {ModTrans} 0
+ {ModPeriod}
.elif ModType==3
.define ModRamp Pulse ModLow ModHigh 0 {ModPeriod/2} {ModPeriod/2} 0 {ModPeriod}
.endif
```

The ModPeriod variable converts the ModFreq parameter into its equivalent period in units of seconds. The ModTrans variable is used to set the transition time for the sawtooth modulation signal. It sets the transition time to the modulation period divided by 10000. This creates a quick transition but provides a finite time in order to aid convergence.

There are three define statements that define the ModRamp variable which creates the modulation signal. In all three cases, the Pulse capability of the voltage source is used to create the signal. An If statement is used in conjuction with the ModType parameter to determine which define statement will be used with the macro.

When the ModType parameter is set to 1, the first ModRamp statement is used. This statement creates the trailing edge sawtooth waveform. The low voltage is set to ModLow. The high voltage is set to ModHigh. The rise time is set to the modulation period minus the transition time. The fall time is set to the transition time, and the period is set to ModPeriod. Both the delay time and the pulse width are set to 0.

When the ModType parameter is set to 2, the second ModRamp statement is used. This statement creates a leading edge sawtooth signal. This operates similar to the first ModRamp statement except that the low voltage is set to ModHigh, and the high voltage is set to ModLow.

When the ModType parameter is set to 3, the third ModRamp statement is used. This statement creates the triangle waveform. The low voltage is set to ModLow, and the high voltage is set to ModHigh. The rise and fall times are set to both be half of the period. The period is set to ModPeriod. Again, both the time delay and the pulse width are set to 0.

Finally, the E1 NFV source creates the PWM output signal. The function source has been defined with the following expression:

```
Vo+Va*tanh(100k*(v(In,Ramp)))
```

This creates a smooth transitioning comparator function which creates the pulse of the modulator. The hyperbolic tangent operator produces a value between 1 and -1. The 100k provides a high gain that minimizes the transition between the two limits. Essentially, when the differential voltage between nodes In and Ramp is positive, the tanh function returns 1. When the differential voltage is negative, the tanh function returns -1. The Vo and Va variables are set through the two following define statements.

```
.define Va (PWMHigh-PWMLow)/2
.define Vo (PWMHigh+PWMLow)/2
```

These two variables force the output voltage of the pulse signal to the high and low values of PWMHigh and PWMLow.

A simple circuit was created to demonstrate the different modulation types. Three of the PWM macros are placed in the schematic. Each of the macros use the same reference signal which is a voltage source that has been set to produce a 200Hz sine wave that has a DC offset of .5V with an amplitude of .5V. The macros all share the following parameter set:

ModHigh = 1
ModLow = 0
ModFreq = 5k
PWMHigh = 1
PWMLow = 0

The only parameter that is varied is the ModType parameter. Each of the macros has been defined to use a separate modulation type. A 5ms transient simulation is run on the circuit. The results appear below.

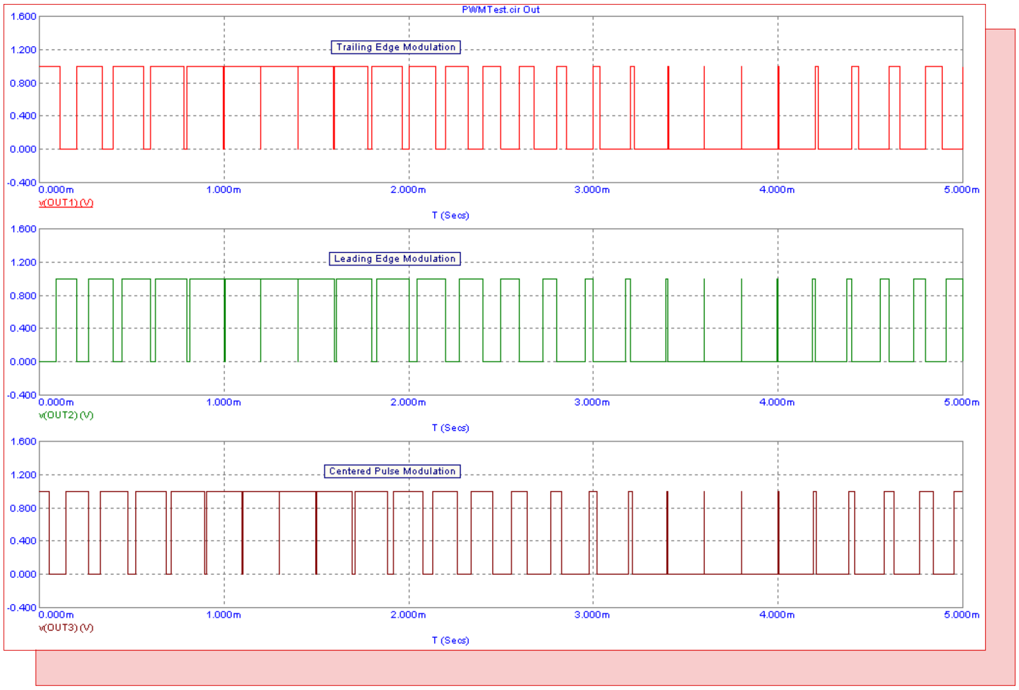


Fig. 8 - PWM output signals

The top plot shows the PWM output when the ModType parameter has been set to 1 so that the modulation signal is the trailing edge sawtooth waveform. The middle plot shows the PWM output when the ModType parameter has been set to 2 so that the modulation signal is the leading edge sawtooth waveform. The bottom plot shows the PWM output when the ModType parameter has been set to 3 so that the modulation signal is the triangle waveform.

The corresponding modulation signals for the simulation are shown in Figure 9. The top plot is the trailing edge sawtooth waveform. The middle plot is the leading edge sawtooth waveform. The bottom plot is the triangle waveform. These modulation signals are plotted by referencing the node Ramp within the macro circuit. For example, to plot the modulation signal within a PWM macro that has the part name X1, the following expression is used:

V(X1.Ramp)

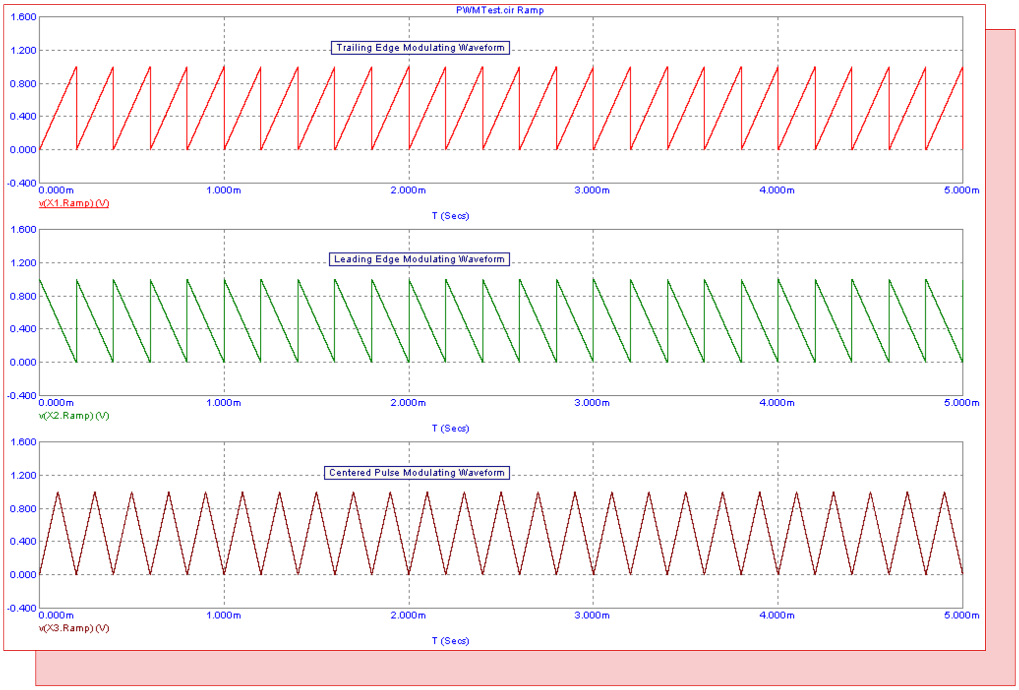


Fig. 9 - Modulation signals

Product Sheet

Latest Version numbers

Micro-Cap 9	Version 9.0.7
Micro-Cap 8	Version 8.1.3
Micro-Cap 7	Version 7.2.4

Spectrum's numbers

Sales	(408) 738-4387
Technical Support	(408) 738-4389
FAX	(408) 738-4702
Email sales	sales@spectrum-soft.com
Email support	support@spectrum-soft.com
Web Site	http://www.spectrum-soft.com
User Group	micro-cap-subscribe@yahoogroups.com