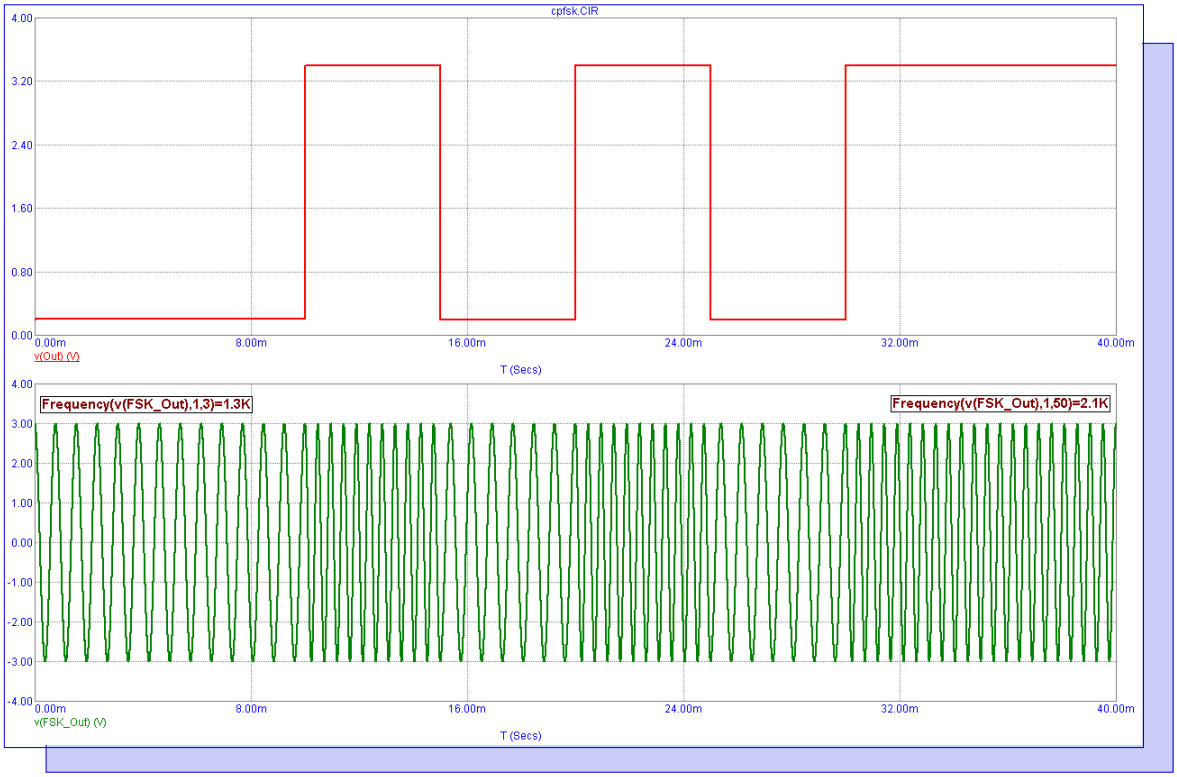
Spectrum Nexus

Applications for Micro-Cap™ Users

Winter 2008

News



CPFSK Modulator Macro

Featuring:

- CPFSK Modulator Macro
- Using the IBIS Components
- New Christophe Basso SMPS Book and Circuit Examples

News In Preview

This newsletter's Q and A section describes a discrepancy that may arise when using the 20*log formulation to plot a waveform in decibels versus using the dB operator. The Easily Overlooked Feature section describes the use of formulas within grid or analysis text.

The first article describes a macro model for a continuous phase frequency shift keying modulator.

The second article describes how to use the IBIS primitive components to easily import IBIS models for use in Micro-Cap simulations.

The third article announces the release of a new Christophe Basso book on the design and simulation of SMPS circuits. Along with the release of the book, Spectrum is making available some of the circuit examples and models available from the book for use with Micro-Cap 9.

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 am plotting the output voltage of my amplifier in units of decibels. When I plot the expression:

20*log(V(Out))

I do not get the answer I would expect. When I plot the following expression:

dB(V(Out))

The simulation is exactly what I am looking for. Why is there a discrepancy between these two when they should be the same calculation?

Answer: The discrepancy between the two expressions lies in how they treat complex numbers which is how a voltage in AC analysis is stored internally in Micro-Cap. The dB operator will always calculate the magnitude of the variable it is working on before it performs its calculation. The expression:

dB(V(Out))

is functionally equivalent to:

dB(Mag(V(Out)))

However, the log operator will perform its calculation directly on the complex form of the variable. Only when the final value is plotted to the screen is the magnitude actually taken. In order to use the 20*log function to plot decibels, the Mag operator must also be included. The equivalent expression to the basic dB expression is:

20*log(Mag(V(Out)))

Almost all of the transcendental operators available in Micro-Cap perform complex mathematical operations so this is an issue to be aware of when using these operators in AC analysis.

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.

Embedding Formulas in Grid or Analysis Text
Formulas may be embedded within a text string in either the schematic or the analysis. This provides a powerful tool for documentation by automatically computing and displaying new design values from formulas. The Grid Text dialog box appears below. To have a formula evaluated within the text string, the Formula checkbox within the Text tab needs to be enabled. The Delimiter text field specifies the delimiters that are to be used to bracket the formula. The default delimiters are the brackets, [ ]. Anything between the delimiters will be evaluated as an expression.

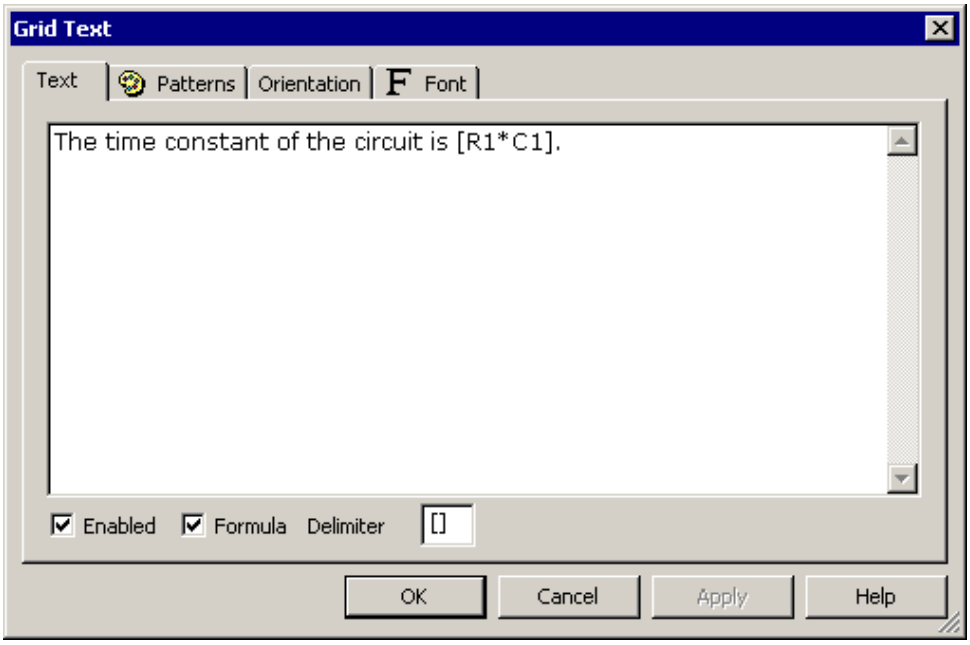


Fig. 1 - Grid Text dialog box

For example, if the following three pieces of grid text are entered in the schematic:

.define C1 6.8n

.define R1 1.3k

The time constant of the circuit is [R1*C1]s.

The last text string will actually appear on the schematic as follows:

The time constant of the circuit is 8.84us.

If either of the C1 and R1 values are modified, then the formula will also be updated to display the new value.

Circuit variables such as V(Out), I(D1), or R(R2) may also be used within a formula. Formulas using circuit variables will only be able to be evaluated when an analysis has just been run.

CPFSK Modulator Macro

Frequency modulation can be performed through various frequency shift keying (FSK) methods. FSK is a way to represent binary information by modifying a sinusoidal carrier signal between two different frequencies. The mark frequency represents the binary one state, and the space frequency represents the binary zero state. One drawback to standard FSK modulation is that at the transition point between the mark frequency and the space frequency, the phase can be discontinuous. These discontinuities can produce unwanted nonlinear effects in some circuits. Continuous phase frequency shift keying (CPFSK) is a variation of FSK which eliminates these phase discontinuities. CPFSK modulation produces a signal with the following expression:

$$y(t) = Ac * \cos(2*PI*(fc*t + Df * \int m(t)*dt))$$

where Ac is the amplitude of the CPFSK signal, fc is the base carrier frequency, and Df controls the frequency deviation. m(t) is the message signal, and integrating this signal provides the continuous phase of the CPFSK. The schematic for a CPFSK modulator macro appears below.

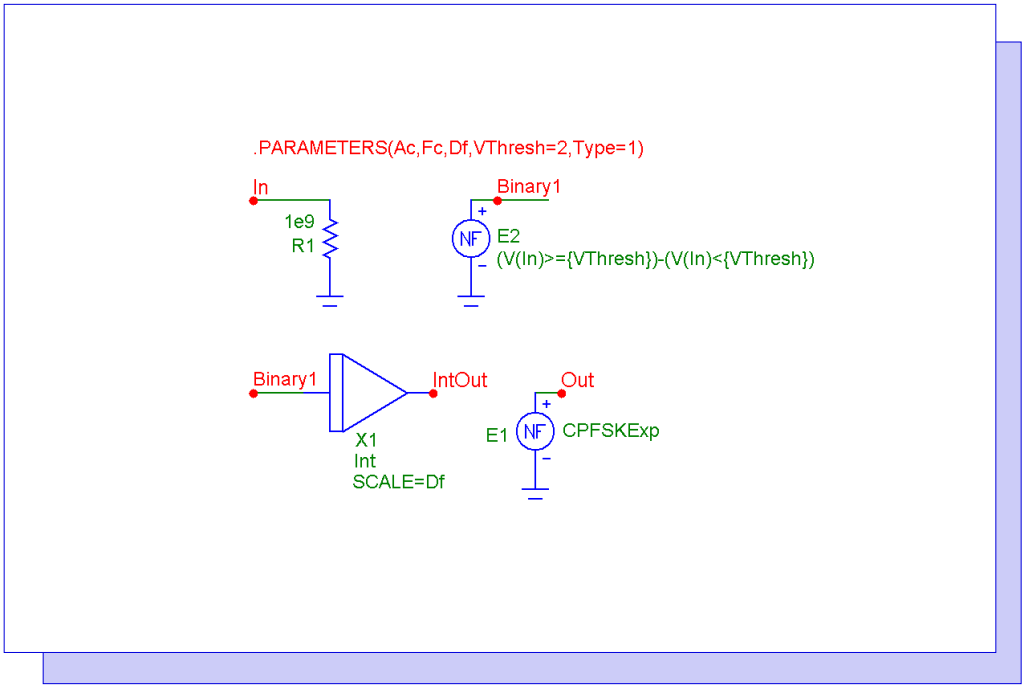


Fig. 2 - CPFSK Modulator macro

The macro circuit has five parameters that are passed through to it: Ac, Fc, Df, VThresh, and Type. The Ac, Fc, and Df parameters all perform the same functions as their counterparts in the above CPFSK modulation expression. The VThresh parameter defines the analog voltage value that will be the threshold voltage in determining whether the message signal should be considered a binary one or a binary zero. At or above this threshould, the message signal will be regarded as a binary one. Below this threshold, the message signal will be regarded as a binary zero. The Type parameter sets whether the higher CPFSK frequency is assigned to the mark frequency or the space frequency. If Type=1, the mark frequency is set to the higher frequency. If Type=2, the space frequency is set to the higher frequency.

The In node is the node where the message signal will be input. The R1 resistor is present at this input node for two reasons. The first reason is to provide a DC path to ground for the In node so

that any element may be connected to it. The second reason is that the NFV sources are only able to work with analog references. If the input waveform happens to be a digital waveform, this resistor will force Micro-Cap to convert it to its equivalent analog voltage for use with the E2 nonlinear function voltage source. The value of the resistor is set to 1e9 ohms so that it will not have a loading effect.

The E2 NFV source converts the message signal into a binary waveform whose analog voltage levels are at 1 and -1. The E2 source has its VALUE attribute defined with the following expression:

(V(In)>={VThresh}) - (V(In)<{VThresh})

When the voltage at node In is greater than or equal to the specified VThresh parameter, the first half of the expression evaluates to true and the output of the source is 1. When the voltage at node In is less than VThresh, then the second half of the expression evaluates to true and the output of the source is -1.

The binary waveform is then input into the X1 Int macro which performs the integration of the message signal that provides the continuous phase performance of the modulator. The Scale value of the integrator macro has been defined with the Df parameter which controls the frequency deviation of the CPFSK output.

Finally, the output of the CPFSK modulator is created through the E1 NFV source. The E1 source has its VALUE attribute set with the variable CPFSKExp. This variable is defined through the following If statement:

```
.if Type==2
.define CPFSKExp Ac*cos(2*PI*(Fc*t - V(IntOut)))
.else
.define CPFSKExp Ac*cos(2*PI*(Fc*t + V(IntOut)))
.endif
```

where V(IntOut) is the voltage output of the Int macro. This source reproduces the basic CPFSK equation shown at the beginning of the article. Depending on the value of the Type parameter, one of these CPFSK expressions will be used. The only difference between the two is whether the IntOut voltage is added or subtracted within the Cos expression.

A simple test circuit for the CPFSK modulator is shown in Figure 3. The circuit consists of a simple two bit shift register that feeds the CPFSK macro. The two D flip-flops are defined with zero gate delays. The clock input to the flip-flops has a period of 5ms. The preset inputs have been wired to a Fixed Digital component which produces a constant one state on those pins. The clear inputs have a short zero state pulse of 100ns to initialize the Q outputs of the flip-flops to zero at the beginning of the simulation. The CPFSK macro has its parameters set as the following:

AC = 3
FC = 1700
DF = 400
VTHRESH = 2
TYPE=1

This combination of parameters will produce a 3 volt sinusoidal output whose mark frequency is at 2100Hz and space frequency is at 1300Hz. The input voltage threshold is set at 2 volts.

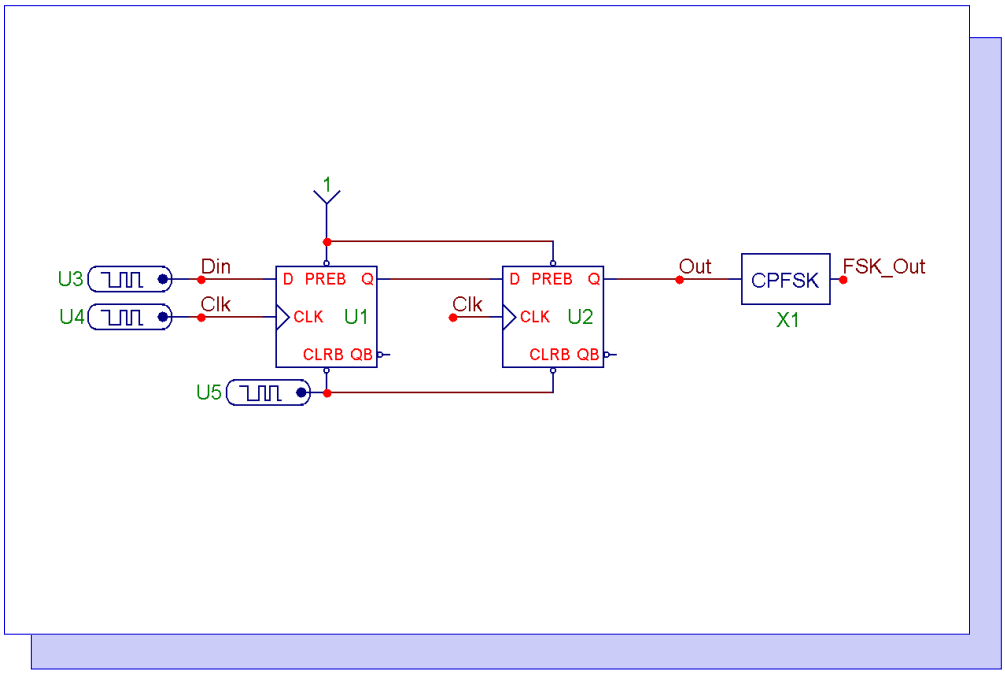


Fig. 3 - CPFSK macro example circuit

The analysis plot in Figure 4 shows the results of a 40ms transient simulation. The top waveform, V(Out), is the input into the modulator. The bottom waveform, V(FSK_Out), is the modulated output.

The two performance tags in the bottom plot calculate the frequencies of the third cycle and the fiftieth cycle of the modulated output. The third cycle occurs when the modulator input is at a zero state and shows the space frequency. The fiftieth cycle occurs when the modulator input is at a one state and shows the mark frequency. Both frequencies are calculated at the expected values of 1300Hz and 2100Hz.

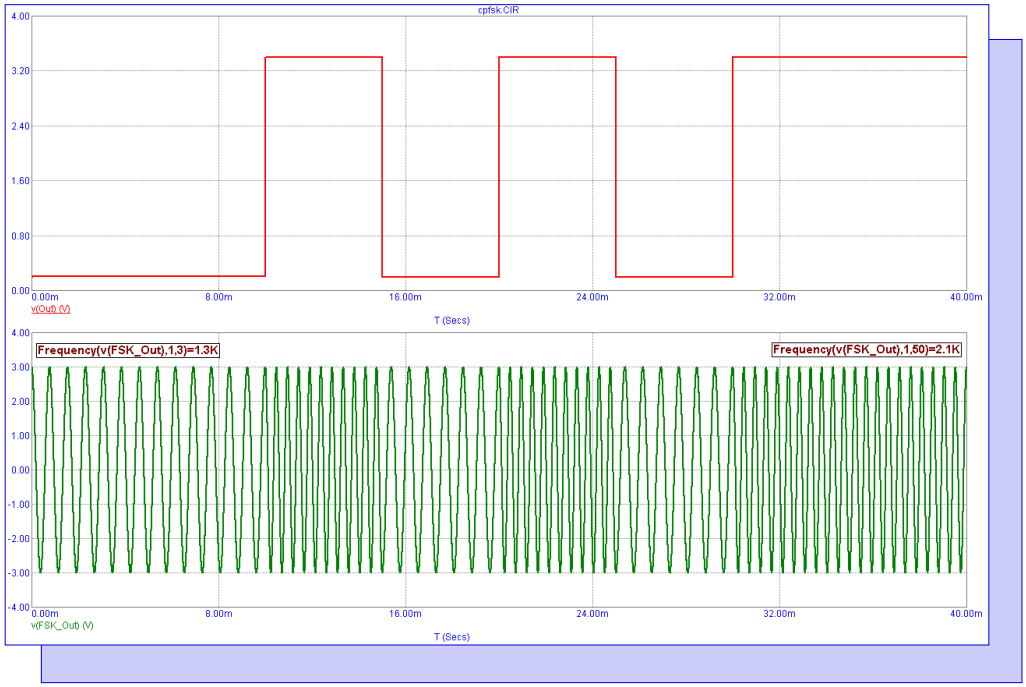


Fig. 4 - CPFSK modulator analysis

Using the IBIS Components

The IBIS model format was created to model the input and output buffer specifications of integrated circuits without revealing proprietary information. These models are widely available from manufacturers and can be used in numerous simulations such as with transmission line or signal integrity analysis.

A common misconception of many users that we have talked to is that they expect the IBIS model to simulate the actual device. IBIS models do not model the internal logic of the device. They only simulate the I/O buffer structure. For simulation of the actual device, a SPICE model needs to be used instead. The typical IBIS file contains the current and voltage characteristics, package parasitics, and protection device characteristics of the I/O buffer structures.

While Micro-Cap contains the IBIS Translator that converts IBIS files into SPICE libraries, the simpler method to using IBIS models in a schematic is to use the IBIS components available in the Analog Primitives / IBIS section of the Component menu.

Placing an IBIS component in a schematic invokes the IBIS Model Creator dialog box which lets a single pin/signal be selected for extraction from a specified IBIS file. The IBIS component then creates a subcircuit model for either an input or an output buffer model from the information available within the IBIS source file. This subcircuit model is created dynamically when the IBIS component is placed in the schematic, and the subcircuit is stored in the Models text page of the schematic.

There are four IBIS components available in Micro-Cap. The four components along with their descriptions are:

IBIS_Input1 - Input buffer model with internal power supplies
IBIS_Input3 - Input buffer model with external power supplies
IBIS_Output3 - Output buffer model with internal power supplies
IBIS_Output5 - Output buffer model with external power supplies

For the IBIS_Input1 and the IBIS_Output3 components, the power supplies for the devices are created within the subcircuits using the supply values specified in the IBIS file. For the IBIS_Input3 and IBIS_Output5 components, the user has to connect the power pins on the components to supplies in the schematic.

Using the components
The first step in using an IBIS component is to get the appropriate IBIS file from the manufacturer of the device. These files are usually available on the company's website so they are easy to download. For this example, an IBIS file from NXP that contains IBIS models for the I/O buffers of the 74AHC00 quad 2-input Nand gate will be used. This IBIS file should be stored in the Library folder of Micro-Cap.

The next step is to enter one of the IBIS components in the schematic. The IBIS_Input3 component is selected from the Analog Primitives / IBIS section and is placed in the schematic. Upon placement, the IBIS Model Creator dialog box shown in Figure 5 will appear.

The IBIS Model Creator dialog box consists of the following items:

Input File: This field specifies the IBIS file name. The file should have an IBIS extension (IBS). The Browse button can be used to browse the library or other folders for IBIS input files.

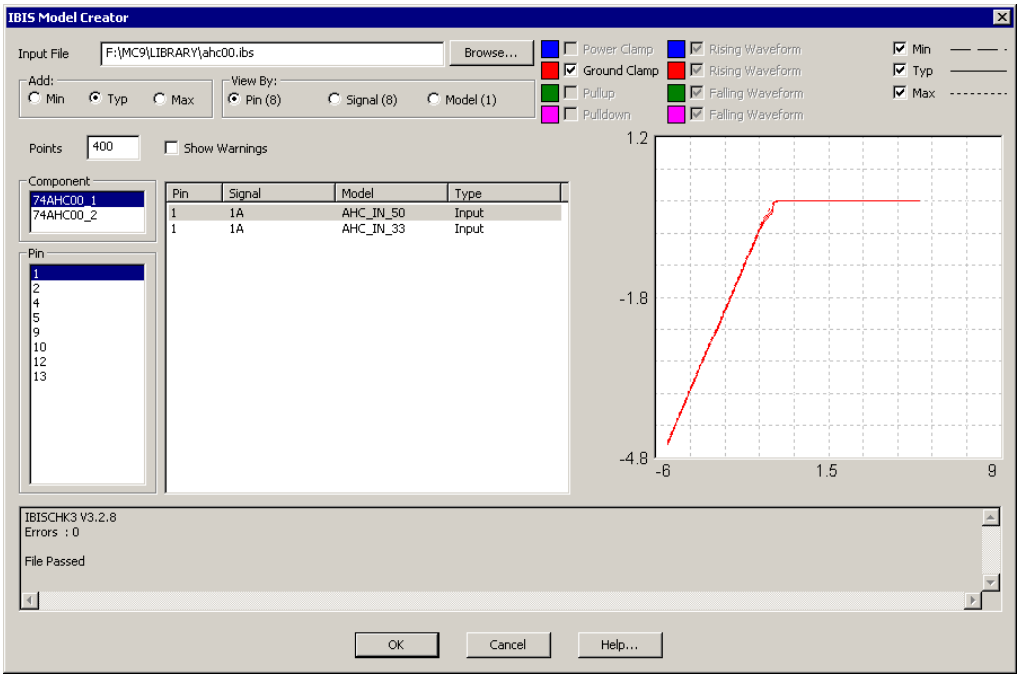


Fig. 5 - IBIS Model Creator dialog box for the AHC00 input models

Add: This determines whether the minimum, typical, or maximum values from the IBIS file are to be used when creating the SPICE model.

View By: This determines whether pin, signal, or model names are displayed in the Item List.

Points: This determines the number of data points the translator uses in the PWL tables.

Show Warnings: If the parser issues a warning, this check box determines whether it will be printed in the Message Box.

Component: This selects one of the component names in the file.

Item List (Pins or Signals or Models): This shows either the pin, signal, or model names depending upon the View By setting. Selecting one or more of the items in this list displays them in the IBIS Display.

IBIS Display: This shows all the available IBIS models available for the selected item in the Item List. This list selects the actual buffer model that will be created.

Plot Controls: The checkboxes in the upper right let you display the specified plots for the buffer that have been defined within the IBIS file.

Plot Area: This is where the IBIS file plots are displayed.

In Figure 5, the Input File field has been set to the AHC00.IBS file from NXP that is now present in the Library folder. All of the input buffer models within the AHC00.IBS file are now available to create a subcircuit model for.

There are two components available in this IBIS file: 74AHC00_1 and 74AHC00_2. The 74AHC00_1 models the device with a SO 14-pin package, and the 74AHC00_2 models the device with a TSSOP 14-pin package. In this case, the 74AHC00_1 is selected.

The View By field has been set to Pin, so all of the input pins for the 74AHC00 device are shown in the Item List. Selecting pin 1 in the list shows that there are two available models in the IBIS file for this pin: AHC_IN_50 and AHC_IN_33. The AHC_IN_50 model is for use with a 5V supply, and the AHC_IN_33 model is for use with a 3.3V supply.

The descriptions mentioned above for the components and models were found within comments in the IBIS file. The IBIS file can be viewed within any text editor.

With the AHC_IN_50 model selected for pin 1, clicking the OK button in the IBIS Model Creator dialog box will then display the standard Attribute dialog box for components. At this point, the subcircuit model has been created and given the name AHC_IN_50_1_TYP which combines the model name, the pin name, and whether it is using minimum, typical, or maximum values. The IBIS button in the Attribute dialog box can be used to invoke the IBIS Model Creator dialog box again. Clicking OK in the Attribute dialog box will then add the IBIS component and place the subcircuit into the Models page of the schematic.

To model an IBIS output buffer, the same procedure is used. In this case, the IBIS_Output5 component is selected from the Analog Primitives / IBIS section and is placed in the schematic. With the AHC00.IBS file set as the input file, the IBIS Model Creator dialog box appears as below.

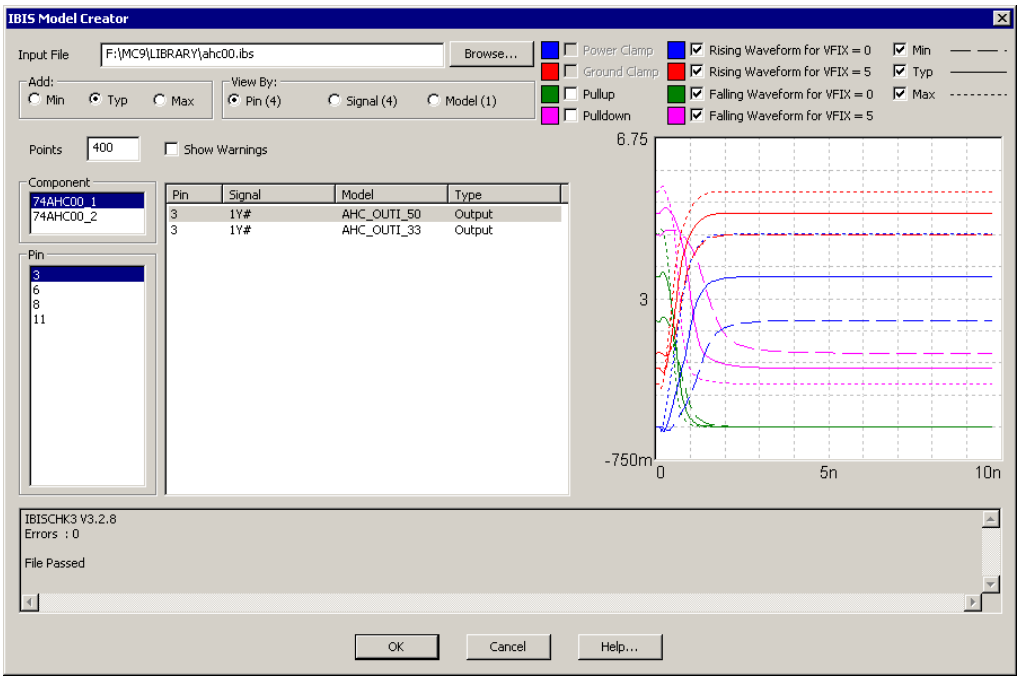


Fig. 6 - IBIS Model Creator dialog box for the AHC00 output models

The View By field has been set to Pin, so all of the output pins for the 74AHC00 device are shown in the Item List. Selecting pin 3 in the list shows that there are two available models in the IBIS file for this pin, AHC_OUTI_50 and AHC_OUTI_33, which respectively model the output buffer with a 5V supply and a 3.3 volt supply.

With the AHC_OUTI_50 model selected for pin 3, clicking the OK button then displays the Attribute dialog box. The subcircuit created in this instance is called AHC_OUTI_50_3_TYP. Clicking OK again adds the part and the subcircuit model to the schematic file.

Example schematic

Using the IBIS input and output buffer models that were created through the above procedure, a simple circuit has been created to simulate the 74AHC00 I/O characteristics. The circuit is shown in the figure below.

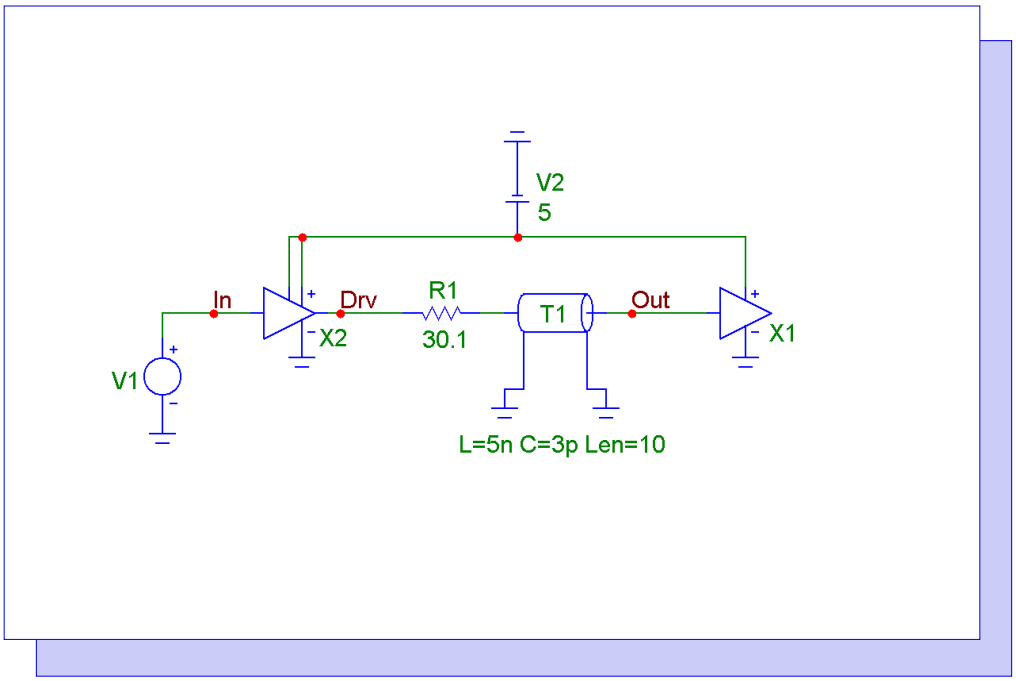


Fig. 7 - IBIS components example schematic

The IBIS_Output5 device, X2, models the output buffer of a 74AHC00 component. This device has five pins. The input pin is tied to a pulse source, V1, that produces a 5V, 40ns pulse that simulates both a rising and falling transition through the output buffer. The negative power supply is grounded. The positive power supply along with the enable pin are tied to a 5V battery. The output pin is connected to a 30.1 ohm resistor which in turn is connected to a transmission line component. The transmission line has its VALUE attribue defined as:

L=5n C=3p Len=10

This simulates the inductance and capacitance of the signal route between the output buffer and the input buffer. Finally, the IBIS_Input3 device, X1, models the input buffer of a 74AHC00 component. This device has three pins. The input pin is connected to the other side of the transmission line. The negative power supply is grounded, and the positive power supply is set to 5V. The IBIS input buffer has no output pin as it is intended to model the load placed on a signal when the buffer is connected to it.

The goal of this simulation is to see how the signal propogates between the output of a 74AHC00 device and one of the inputs of another 74AHC00 device going through a route that has the basic characteristics modelled by the resistor and transmission line.

The resulting transient analysis is displayed below. Three waveforms have been plotted. V(In) plots the pulse signal created by the V1 voltage source. V(Drv) plots the voltage waveform at the output of the X2 output buffer. V(Out) plots the voltage waveform at the input of the input buffer.

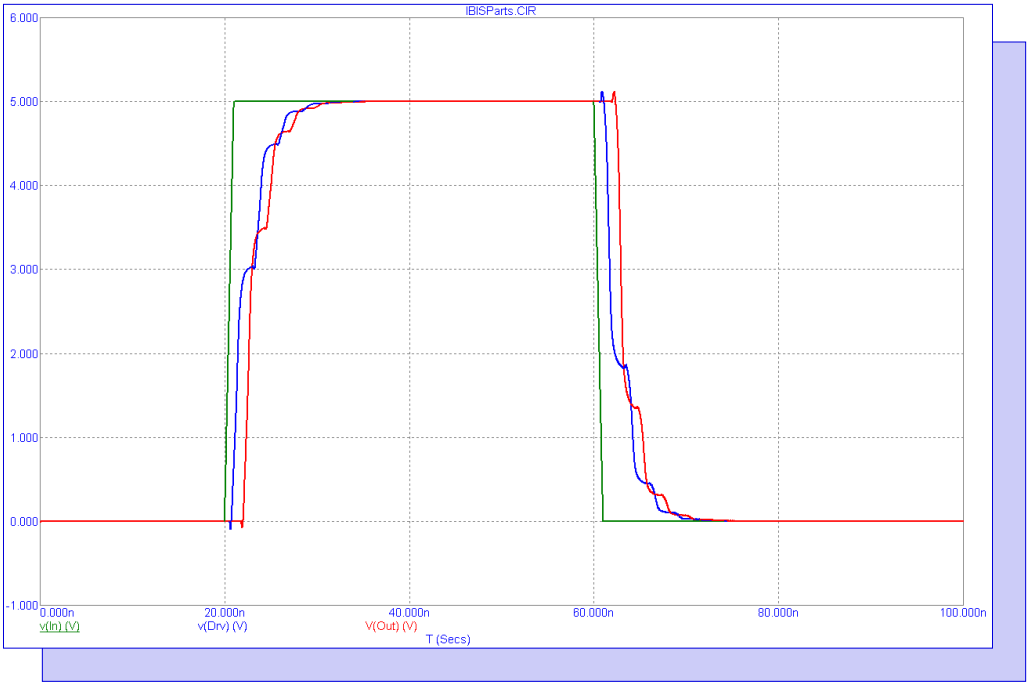


Fig. 8 - IBIS components example transient analysis

New Christophe Basso SMPS Book and Circuit Examples

Christophe Basso has just come out with his latest book on the design and simulation of SMPS circuits titled " Switch-Mode Power Supplies: SPICE Simulations and Practical Designs". The following blurb is the book description on Amazon:

> Switch-Mode Power Supplies: SPICE Simulations and Practical Designs is a comprehensive resource on using SPICE as a power conversion design companion. This book uniquely bridges analysis and market reality to teach the development and marketing of state-of-the art switching converters. Invaluable to both the graduating student and the experienced design engineer, this guide explains how to derive founding equations of the most popular converters, design safe, reliable converters through numerous practical examples, and utilize SPICE simulations to virtually breadboard a converter on the PC before using the soldering iron.
>
> Filled with more than 600 illustrations, Switch-Mode Power Supplies: SPICE Simulations and Practical Designs enables you to:
>
> Derive founding equations of popular converters
> Understand and implement loop control via the book-exclusive small-signal models
> Design safe, reliable converters through practical examples
> Use SPICE simulations to virtually breadboard a converter on the PC
> Access design spreadsheets and simulation templates on the accompanying CD-ROM
>
> Inside This Powerful SPICE Simulation and Design Resource
>
> Introduction to Power Conversion
> Small-Signal Modeling
> Feedback and Control Loops
> Basic Blocks and Generic Models
> Simulation and Design of Nonisolated Converters
> Simulation and Design of Isolated Converters-Front-End Rectification and Power Factor Correction
> Simulation and Design of Isolated Converters-The Flyback
> Simulation and Design of Isolated Converters-The Forward

Christophe Basso has generously let us distribute some of the models and circuits that are available in the book. These files are available in the Winter 2008 section of the Newsletter page of our website. The circuit files are in Micro-Cap 9 format. View the Readme.txt file within the .ZIP file for information on using these files.

The schematic in Figure 9 is a voltage mode buck converter that uses a voltage mode PWM Controller average model that auto transitions between discontinuous and continuous conduction modes. A switch is used to simulate the load changing from 2.5ohms to 5ohms for a 2ms portion of the simulation. The transient analysis of this circuit is displayed in Figure 10.

The schematic in Figure 11 is the same voltage mode buck converter. This schematic has been configured to plot the open loop gain of the circuit. The gain and phase plots of the circuit are shown in Figure 12.

These are two of the approximately eighty circuits that Christophe Basso has made available to us that cover a wide range of SMPS simulation.

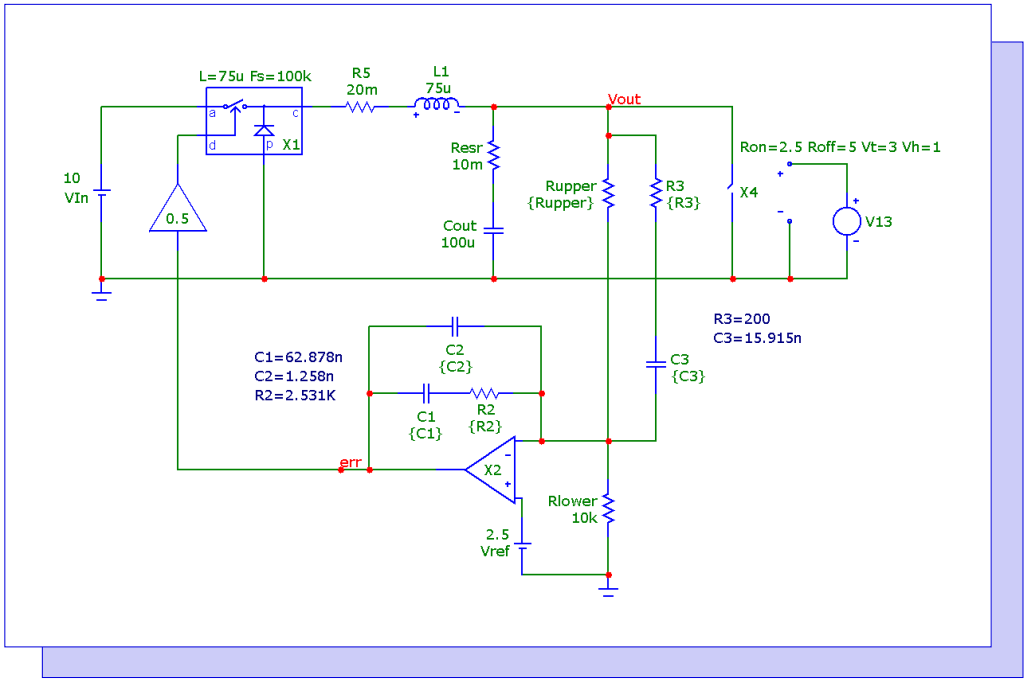


Fig. 9 - Voltage mode buck converter schematic

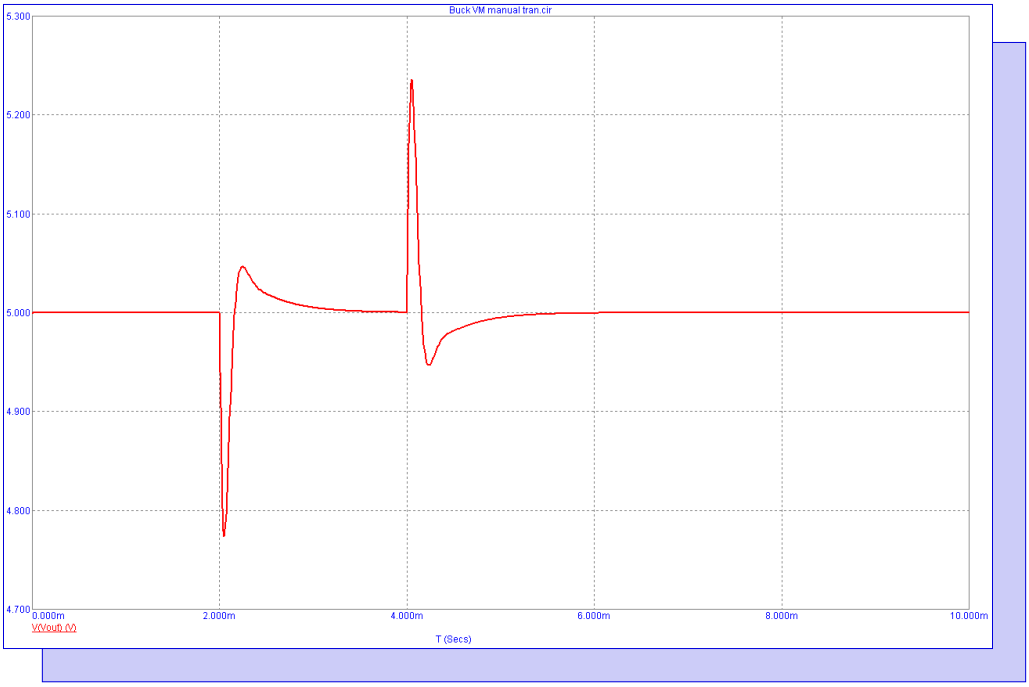


Fig. 10 - Step load response of the voltage mode buck converter

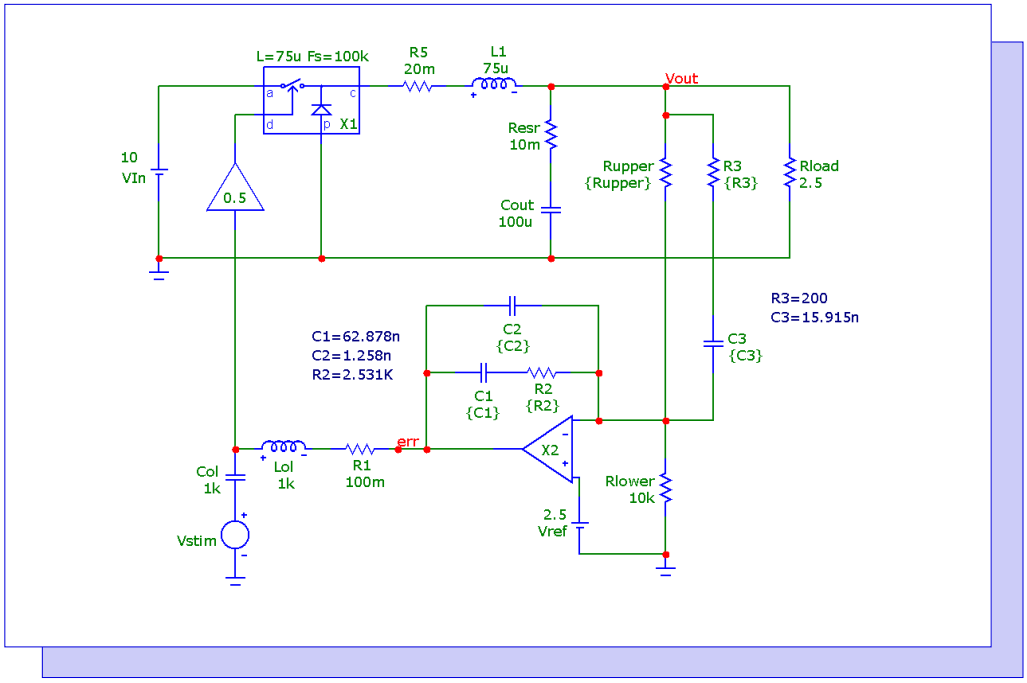


Fig. 11 - Open loop voltage mode buck converter schematic

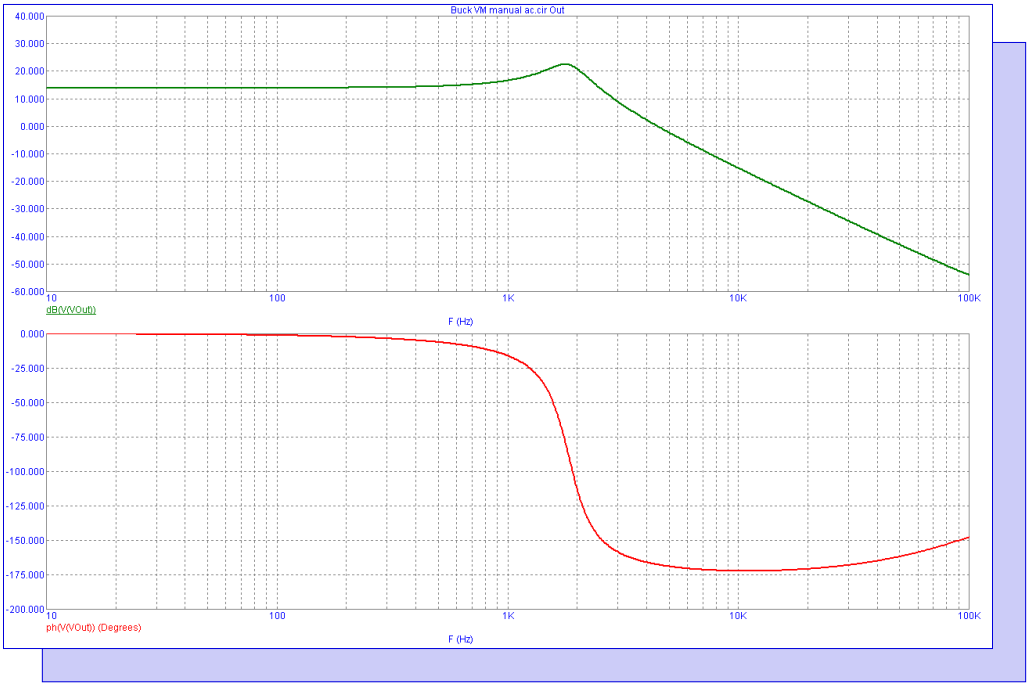


Fig. 12 - Open loop response of the voltage mode buck converter

Product Sheet

Latest Version numbers

Micro-Cap 9 ..... Version 9.0.3
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