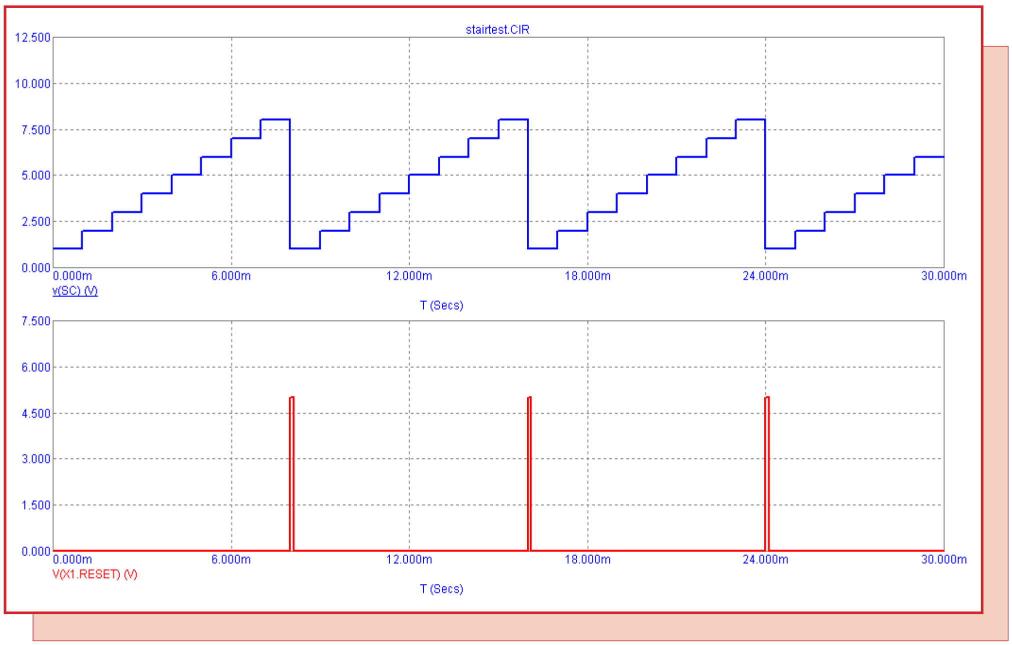
spectrum news

Applications for Micro-Cap™ Users

Fall 2004

News



Staircase Source Macro

Featuring:

- Staircase Source Macro
- Upgrading to Micro-Cap 8
- Using the Micro-Cap 8 IBIS Translator

News In Preview

This newsletter's Q and A section describes how to create a triangle waveform source and how to import numeric output data into Excel. The Easily Overlooked Features section describes how the mouse tracker in an analysis will display the values of stepped parameters when stepping has been enabled.

The first article describes how to model a staircase source. This source is particularly useful in video, audio, and A/D applications.

The second article describes a method to upgrade previous Micro-Cap versions to Micro-Cap 8.

The third article describes the use of the IBIS Translator and takes you through the entire process from having the IBIS file to using the equivalent SPICE model in the schematic.

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, First Edition, 1993. ISBN# 0-07-911525-X

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

MOSFET Modeling

• *MOSFET Models for SPICE Simulation, William Liu, Including BSIM3v3 and BSIM4*, Wiley-Interscience, First Edition, 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 - 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# ISBN 951-0-25672-2

Design

• *Microelectronic Circuits High Performance Audio Power Amplifiers,* Ben Duncan, Newnes, First Edition, 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, First Edition, 1997.
ISBN# 0-07-913227-8

• *Switch-Mode Power Supply SPICE Simulation Cookbook*, Christophe Basso, McGraw-Hill 2001. This book describes many of the SMPS models supplied with Micro-Cap.

Micro-Cap Questions and Answers

Question: How do I create a triangle waveform source?

Answer: A triangle waveform source can be created using the Voltage Source component available in the Waveform Sources group of the Analog Primitives section in the Component menu. When the source is placed in a schematic, select the Pulse tab. Set the PW (pulse width) parameter to 0. The TR and TF parameters will define the rise and fall times of the triangle source respectively. Due to the zero value of the pulse width, the result will be a triangle waveform. For a current triangle waveform, select the part called Current Source, available in the same group, and follow the same technique.

Question: I've run a simulation and specified the waveforms to print their data in the numeric output file. If I open the numeric output file and then copy and paste the tabular data into Excel, all of the data copied gets pasted into a single cell. How do I send this data over to Excel so that it will place each piece of data in its own cell?

Answer: When numeric output is enabled for a waveform and the simulation is run, a column of data for that waveform will be placed in the Numeric Output file which can be accessed under the specific analysis menu (Transient, AC, or DC). This output is just a text file that will have one of the extensions: .TNO, .ANO, or .DNO depending on the type of analysis that was run. In Excel, you can then use the Get External Data / Import Text File option that is available under the Data menu to import the tabular data into Excel. This Excel function lets you choose the delimiter that the import will use. When importing numeric output files, both the Tab and the Space delimiters should be selected.

By default, Micro-Cap will usually place a lot more information in the numeric output than just the tabular data of the waveform such as the operating point results for individual devices. This can potentially take up a large amount of the file and probably is not information that one would want to import into Excel. The Import Text File function in Excel lets you choose the first row to begin the import process on as a way to skip all of the other information and just import the tabular waveform data. Alternatively, in Micro-Cap, the Properties dialog box in the analysis provides a way to limit the type of information that is sent to the numeric output file. In the Numeric Output page of the Properties dialog box, disabling all of the options but Include Numeric Output and Include Waveform Values will result in a numeric output file with just the tabular waveform data.

Easily Overlooked Features

This section is designed to highlight one or two features per issue that may be overlooked because they are not made visually obvious with a toolbar button.

Stepping Information in the Mouse Tracker

When a simulation uses stepping, multiple branches of each waveform will appear in the plot. Sometimes it can be difficult to determine which branch belongs to which value of the stepped parameter. Moving the cursor between branches in cursor mode or using the Label Branches option under the Scope menu are two options to view the stepping information. However, the simplest method is by using the mouse tracker. The mouse tracker displays information that pertains to the point on the plot that the cursor is currently pointing at. This tracker can be enabled by going to the Trackers section under the Scope menu.

In the figure below, a transient simulation has been run where the R1 resistor is stepped from 0 to 5k in 1k steps, and the C2 capacitor was stepped so that its two values were 3nF and 33nF. This stepping produces a total of twelve different branches. Moving the mouse cursor over one of the branches will display its stepping information in the mouse tracker such as in the figure where the tracker displays the following:

8.809m,0.780 C2=3n R1=4K

The first two values are the coordinates that the mouse is currently pointing at. The rest of the tracker information displays the values of the stepped parameters for the waveform branch that is closest to the mouse cursor. In this case, the C2 capacitor is at 3nF and the R1 resistor is at 4k Ohms for the waveform branch that the cursor is pointing to.

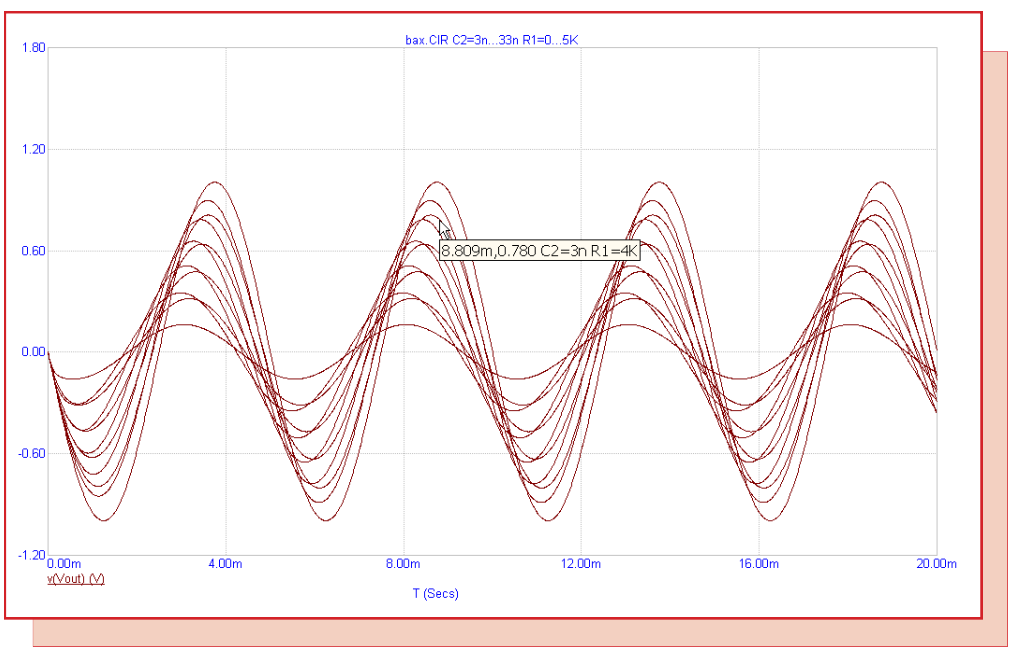


Fig. 1 - Mouse tracker with stepping display

Staircase Source Macro

The staircase source produces a waveform that strangely enough looks like a staircase. This type of source waveform is commonly used in video and audio system simulation. It can also be used in many D/A applications such as in linearity testing or testing circuitry such as a digital amplitude modulator. There are a number of different ways to create a staircase waveform. One can add the output of multiple pulse sources together, use the output of a digital to analog converter, or use the PWL capability of the Voltage Source component. Our solution is a simple and very flexible macro circuit for creating a staircase waveform that is displayed in Figure 2.

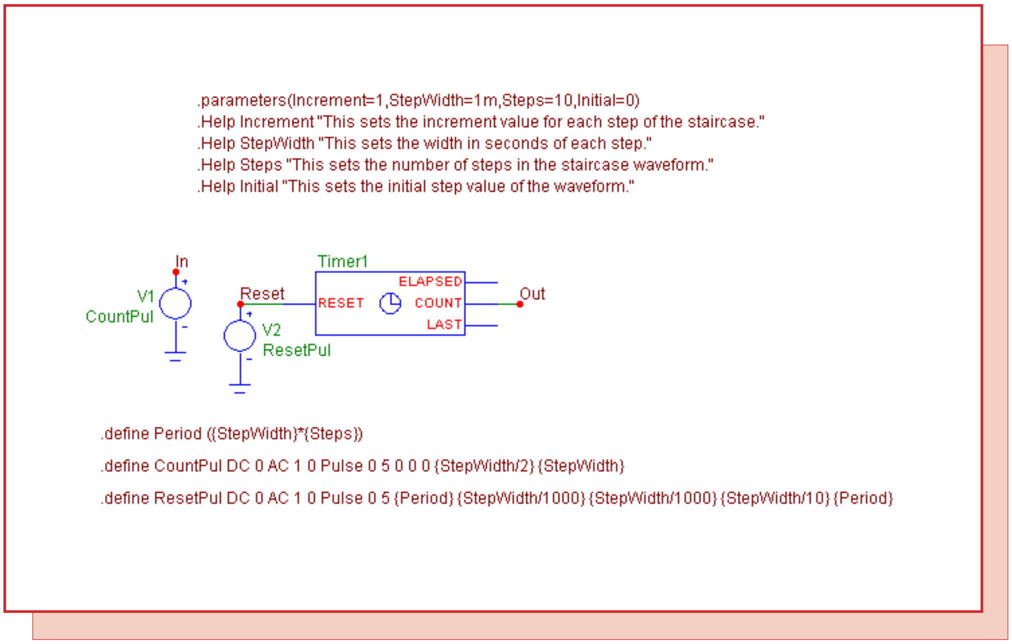


Fig. 2 - Staircase Source Macro Circuit

This macro circuit produces a periodic staircase waveform. The staircase macro has four parameters that are passed to it: Increment, StepWidth, Steps, and Initial. The Increment parameter defines the incremental voltage value for each step of the staircase. The StepWidth parameter defines the width in seconds of each step. The Steps parameter defines the total number of steps that the staircase waveform will consist of. The Initial parameter defines the initial voltage value that the staircase will begin at for each period of the waveform. The period of the waveform is calculated by multiplying the StepWidth and the Steps parameters through a define statement.

The macro circuit consists of two voltage sources and a timer component. The first voltage source, V1, defines the voltage at node In. The VALUE attribute for the source has been defined as CountPul with a corresponding define statement of:

.define CountPul DC 0 AC 1 0 Pulse 0 5 0 0 0 {StepWidth/2} {StepWidth}

This statement produces a pulse with a 50% duty cycle whose period is equivalent to the StepWidth parameter. The source rises immediately to 5 volts at the beginning of each period, and then drops to 0 volts at the midpoint of the period. This source will be used to trigger the counter that resides within the timer component.

The timer component is new to Micro-Cap 8. The timer is designed to monitor for a specific event in the simulation. The timer counts the total number of events since the start of the simulation or the end of the last reset pulse, the elapsed time since an event occurred, and the last time the event occurred. For the staircase macro, only the counting feature is of interest. The attributes for the timer are as follows:

INPUTEXPR = (V(In) > 2.5) * (t >= {StepWidth})
ELAPSED_SCALE = 1
INCREMENT = {Increment}
INITIAL = {Initial}
MIN = -1e9
MAX = 1e9

The INPUTEXPR attribute is a boolean expression that defines the event to be monitored. In this case, it will record an event when the voltage at node In becomes greater than 2.5 volts and the simulation time equals or exceeds the StepWidth parameter. Since the V1 source at node In creates a 5 volt pulse every StepWidth seconds, the count will be incremented every StepWidth seconds. The simulation time requirement of the INPUTEXPR attribute prevents the timer from recording the very first pulse in the simulation as the V1 source will also create a 5V pulse at time = 0 which should be ignored by the timer.

The ELAPSED_SCALE attribute is used to scale the voltage on the Elapsed pin of the timer. This pin is not used in this macro.

The INCREMENT attribute defines the value that the count will be incremented by each time a valid event is recorded. In this case, it is defined with the parameter Increment that is being passed into the macro.

The INITIAL attribute defines the initial value that the count will be set to when the simulation starts or when a reset pulse is triggered. In this case, it is defined with the parameter Initial that is being passed into the macro.

The MIN and MAX attributes simply define the minimum and maximum values that the counter is capable of counting to.

The current count value of the timer is produced as a voltage on the Count pin of the timer. This count voltage models the staircase waveform.

The second voltage source, V2, is connected to the Reset pin of the timer. The VALUE attribute for the source has been defined as ResetPul with a corresponding define statement of:

.define ResetPul DC 0 AC 1 0 Pulse 0 5 {Period} {StepWidth/1000} {StepWidth/1000} {StepWidth/10} {Period}

The Period variable used here is calculated by multiplying the Steps and StepWidth parameters. This statement creates a five volt pulse every Period seconds after an initial time delay of Period seconds. The pulse width is a tenth of the passed StepWidth parameter which makes sure that the reset doesn't mask the next valid count event. The Reset pin of the timer will reset the count value to its Initial value whenever the voltage on the pin becomes equal to or greater than one volt. The V2 source waveform produces the periodic capability of the staircase waveform.

A sample transient analysis for the staircase source is displayed in Figure 3. The simulation time is 30ms and the staircase macro's parameters have been defined as follows:

Increment = 1
StepWidth = 1m
Steps = 8
Initial = 1

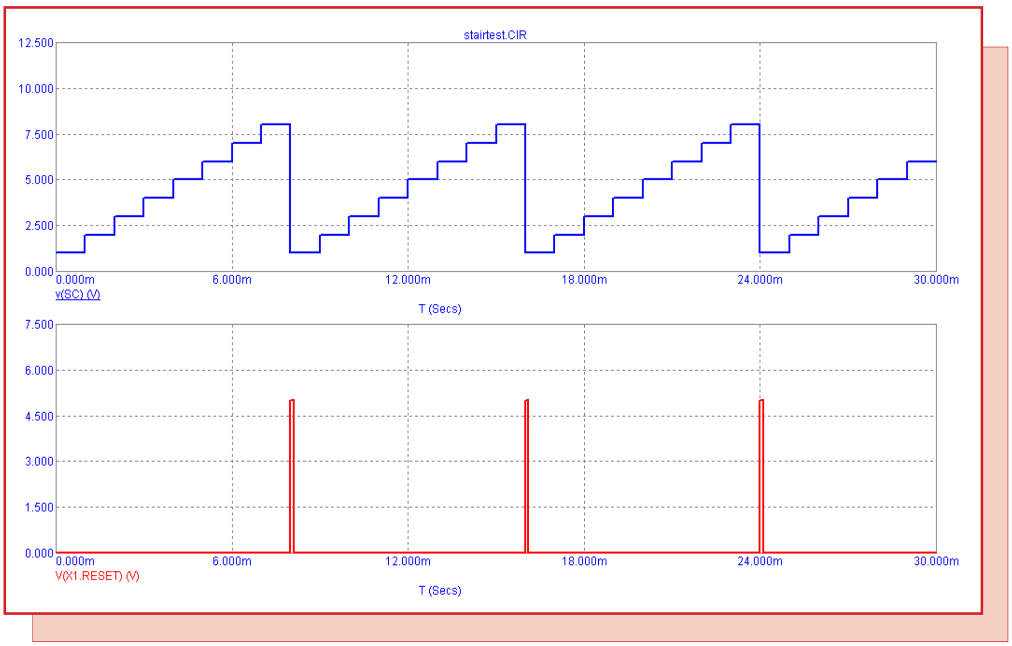


Fig. 3 - Staircase Source Output

The staircase produces the expected eight steps going from 1 volt on the first step to 8 volts on the final step before being reset. Note that the period of the staircase waveform is 8ms (1m*8). The internal reset node voltage of the macro has also been plotted with the expression:

V(X1.Reset)

where Reset is the node name inside the macro, and X1 is the staircase macro Part designator.

Upgrading to Micro-Cap 8

When upgrading to a new version of Micro-Cap, the user will typically want to import data from their previous version of Micro-Cap. The type of data may consist of files such as schematics, macro circuits, and libraries, schematic information for components, shapes, and packages, or user defined settings and preferences. The following sections will provide tips and methods for how to make a smooth upgrade between versions of Micro-Cap.

Installation Location

Do not install Micro-Cap 8 in the same folder as the previous version of Micro-Cap. Install Micro-Cap 8 in a new folder. This folder may be located anywhere on the hard drive. Installing in the same folder as a previous version could overwrite files that may still be needed for specific simulations. It may even prevent a user from being able to run the older version if necessary. Keeping the older version in a separate directory provides a nice, simple backup as Micro-Cap 8 will convert files to an updated format when they are loaded in the program.

User Settings

A feature new to Micro-Cap 8 is the Migrate command which is available under the File menu. The Migrate command provides a means to transfer user settings and preferences along with component, shape, and package information. Upon selecting this command, the dialog box in Figure 4 will appear.

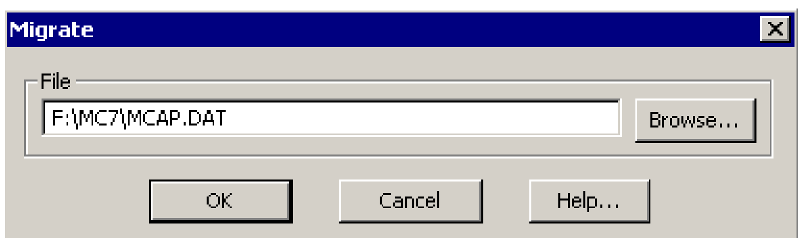


Fig. 4 - Migrate File dialog box

This dialog box will prompt you for the location of the MCAP.DAT file (MC5.DAT if upgrading from Micro-Cap 5). The MCAP.DAT file will be located in the main directory of the previous Micro-Cap installation. This file stores general program information for Micro-Cap including default colors and fonts, Global Setting values, toolbar settings, Preferences option settings and values, and loaded component and shape files. Upon hitting the OK button, the information that is applicable to Micro-Cap 8 will be transferred over.

Component, Shape, and Package Information

As part of the Migrate command, when the OK button is clicked in the process above, a new dialog box will be invoked that displays a list of all of the component files (.CMP), shape files (.SHP), and package files (.PKG) that were loaded in the previous Micro-Cap version. This dialog box is displayed in Figure 5. Simply enable the checkbox next to the file or files that are to be migrated over. Any combination of files may be selected. If no files are selected, just hit Close to exit the Migrate process and only the MCAP.DAT information will have been transferred. Once the desired files have been enabled for migration, hit the Migrate button to start the process.

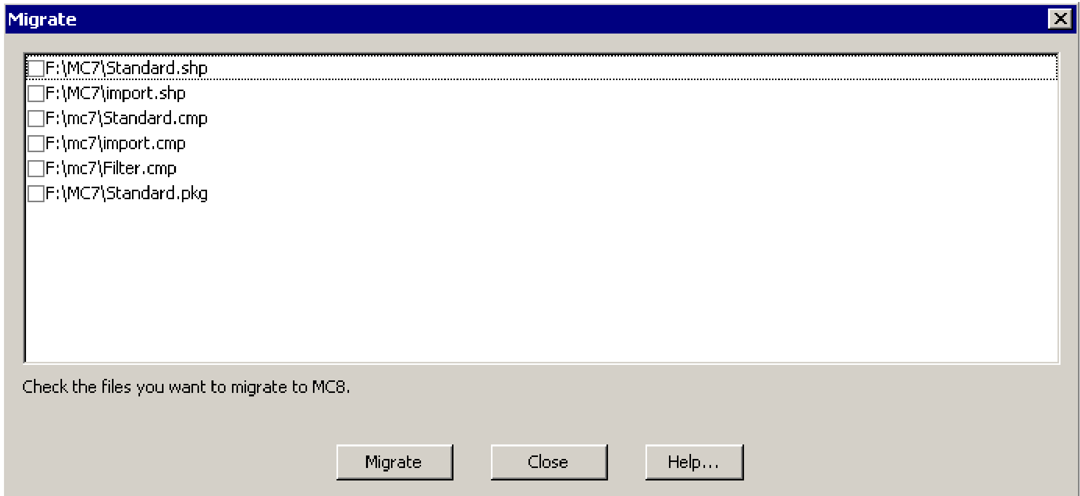


Fig. 5 - Component, Shape, and Package Migrate dialog box

If you are migrating component files over to MC8, and have created your own shapes for these components in the previous version of Micro-Cap, then you must also migrate the corresponding shape files. Otherwise, the shape that your component references will not exist in MC8 and you will get a default shape such as a battery. Similarly, when migrating packages, the corresponding component files also need to be migrated. Once the Migrate button is clicked, each of the enabled files is scanned and any component, shape, or package that does not exist in Micro-Cap 8 will be imported in. The finished results of a migration are displayed in Figure 6.

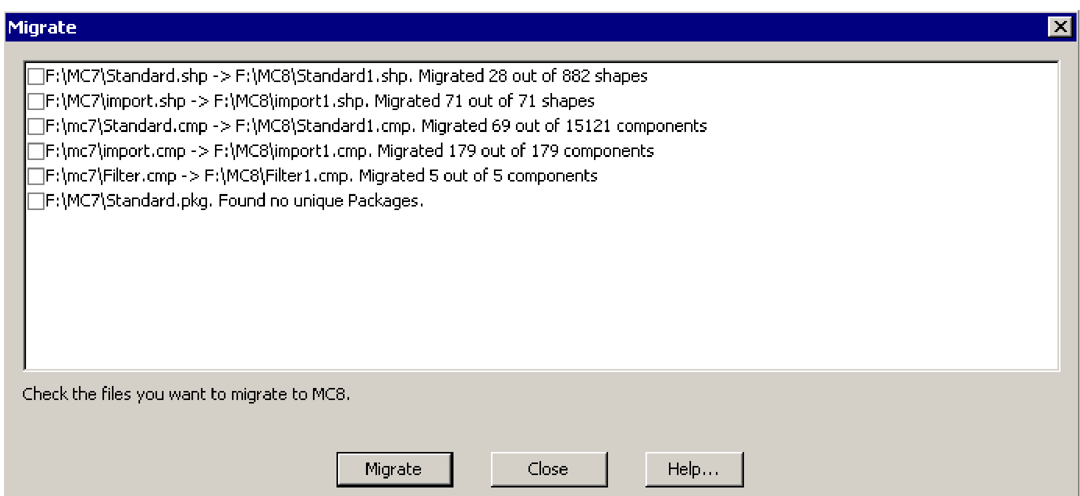


Fig. 6 - Component, Shape, and Package Migration Results

The results display the file name and location that the data was transferred to along with the number of elements that were retrieved from the file. For example, in this case the Standard.cmp file that was used by MC7 has had 69 of its components transferred to a Standard1.cmp file that

MC8 will access. These components will now be available for placement in a Micro-Cap 8 schematic. Depending on the group that the migrated components were stored in for the previous version of Micro-Cap, you may now see duplicate groups such as Analog Primitives when you click on the Component menu. The duplication is due to the fact that the migrated components were stored in a separate .CMP file for Micro-Cap 8, and both the new .CMP file and the Standard.cmp file that comes with Micro-Cap 8 contain the same group name. The Component Editor can be used to either rename the group or to move components between groups or files.

Library and Macro Files

While the schematic information for the components are imported into Micro-Cap 8 through the Migrate command, the underlying simulation models for the components will not be transferred over in this process. The simulation models are typically stored as macro circuit files (.MAC) or within library files (.LIB, .LBR). The macro and library files for Micro-Cap 8 should be stored in the library directory that is specified by the Model Library field within the Path dialog box which can be invoked by selecting the Paths option under the File menu. The default library path upon installation will be:

<drive>:\MC8\LIBRARY\

where <drive> is the letter of the hard drive partition that Micro-Cap was installed on. Any necessary macro and library files should be copied over from the old version into this library directory of Micro-Cap 8. Be careful not to just copy over all macro and library files that are in the old version. Doing this may overwrite those files that are installed with Micro-Cap 8, and many of those files have been updated from previous versions.

Once the macro files have been transferred to the MC8 library directory, they will be immediately available for simulation. For the library files, one more step needs to be taken. *Each of the transferred libraries needs to have their file names referenced in the NOM.LIB file.* The NOM.LIB file is a text file that resides in the library directory. It contains a list of all of the library files that Micro-Cap will automatically access. If a library is not listed in here, Micro-Cap will not be able to locate any of the models within the library without an explicit reference in the schematic file. To add library references to this file, open up the NOM.LIB in Micro-Cap 8 or a basic text editor such as Wordpad. For each library, place a .lib statement on a new line in the file such as:

.lib "myfile.lib"

Once the library is referenced in the NOM.LIB file, any models within the file will now be available for simulation.

Circuit Files

The circuit files should be transferred over to the data directory of Micro-Cap 8. The data directory will be specified in the Data field of the Path dialog box. The default data path upon installation will be:

<drive>:\MC8\DATA\

The conversion of a schematic file to the new Micro-Cap 8 format will occur when the file is initially loaded into Micro-Cap 8 but will only be permanently converted when the schematic file is saved. For those users who need to convert a schematic back to an older format, the File menu contains a Translate option that will let a schematic file be converted into MC5, MC6, or MC7 format.

User Definitions

The User Definitions are stored in the MCAP.INC file (DEF.MC5 for Micro-Cap 5) that resides in the main Micro-Cap directory. This file is easily accessible by going to the Options menu and selecting User Definitions. This file contains user defined command statements that will automatically be included whenever a simulation is entered. Typically, the command statements are .define statements but other statements are also valid. The easiest method for transferring this information is through a copy and paste operation. Load both the old and new MCAP.INC files in a text editor or their respective versions of Micro-Cap. Then just copy and paste the information that needs to be transferred. If none of the command statements in the installed MCAP.INC are of interest, then the file can also be overwritten.

Miscellaneous Files

There are other files that may need to be brought over to Micro-Cap 8, but that may not be crucial in running most simulations. These files should be copied to the appropriate library or data directory. Files such as Model (.MDL) or filter impedance files (.RES, .IND, .CAP) should be placed in the library directory. Files such as State Variable (.TOP), S Parameter (.S2P), User Source (.USR), or Digital File Stimulus files (.STM) should be placed in the data directory.

Using the Micro-Cap 8 IBIS Translator

The IBIS (Input/Output Buffer Information Specification) model format was developed by Intel as a means to create analog models that represent digital integrated circuit I/O characteristics. Many semiconductor manufacturers have IBIS files available as it provides an accurate model that does not reveal proprietary process or circuit design information. These models are most widely used for transmission line and signal integrity analysis. One important note to make of these models is that they do not model the internal logic of the specified device. They only simulate the I/O buffer structure. The typical IBIS file contains the current and voltage characteristics, package parasitics, and protection device characteristics of the I/O buffer structures. While IBIS files can not be used directly in Micro-Cap, there is an IBIS translator available that will convert these files into their SPICE equivalent. This article describes the process from having the IBIS file to using the equivalent SPICE model in the schematic.

The IBIS Translator

The IBIS file used for this example was downloaded from the Philips Semiconductors website. The file models the I/O buffer structure of the 74AHC00 quad 2-input nand gate. The recommended location to place the IBIS file is in the Library directory of Micro-Cap. Once you have the IBIS file in the desired folder, you can invoke the IBIS translator by going to the File menu, clicking Translate, and then selecting the IBIS to SPICE File option. The IBIS translator appears in Figure 7. The Input File field should specify the name of the IBIS file that is to be translated. The Output File field specifies the name of the SPICE library file that will be created in the translation.

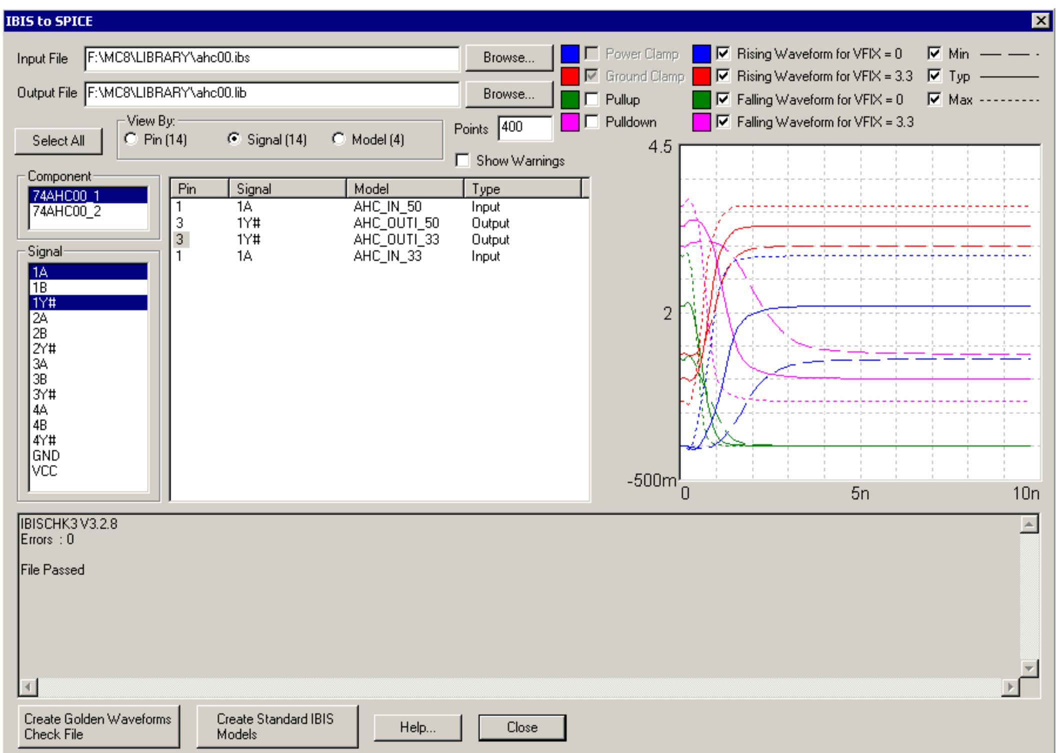


Fig. 7 - IBIS Translator

The next step is to select the models that are to be created in the translation. The View By options determine the list of objects that are to be displayed. The Pin option displays the list in terms of the pin numbers on the IC. The Signal option displays the list in terms of the signal names of the IC. The Model option displays the list in terms of the model names defined within the IBIS file. The Component field selects the part within the IBIS file whose pins/signals/models will be displayed in the list. Typically, the difference between the components are based on package variations. In this example, the 74AHC00_1 defines the 74AHC00 component within a SO 14 pin package. The 74AHC00_2 defines the 74AHC00 within a TSSOP 14 pin package. This infomation can only be viewed by opening the IBIS file in a text editor and reading the comments that are assigned to each component. Below the Component field is the Item List which displays either the pin names, signal names, or model names depending on the selection in the View By field. Multiple items can be selected in this list by holding down the CTRL key when clicking on a name. The IBIS List display is the large display in the middle of the translator. This display shows all of the appropriate information for the items selected in the Item List. The items within the IBIS List display will be the ones that have their equivalent SPICE models created during the translation.

For this example, the Signal option has been selected in the View By field to display a list of the signal names in the Item List. The Component selected is the 74AHC00_1. In the Item List, the signals 1A and 1Y have been selected in order to create a typical input and output for the device. Note that in the IBIS List, there are two entries for each of the signals that have been selected. This particular IBIS file has data for the 74AHC00 component at two power supply voltages so models will be created using both the 5V and 3.3V power supply data.

The next step is to perform the translation to the SPICE models. Click on the Create Standard IBIS Models button. A SPICE library file will be created at this point and saved to disk. A second option is to click the Check Golden Waveforms Check File button. This command is mainly used to test the data within the IBIS file. For typical system testing the Create Standard IBIS Models command should be used. Click Close. The new SPICE library will now be displayed in Micro-Cap.

The comment section at the top of the SPICE library will contain a section such as:

```
* Output Buffer Models:
*    AHC_OUTI_50_3_TYPVCC
*    AHC_OUTI_50_3_MINVCC
*    AHC_OUTI_50_3_MAXVCC
*    AHC_OUTI_33_3_TYPVCC
*    AHC_OUTI_33_3_MINVCC
*    AHC_OUTI_33_3_MAXVCC
*
* Input Buffer Models:
*    AHC_IN_50_1_TYPVCC
*    AHC_IN_50_1_MINVCC
*    AHC_IN_50_1_MAXVCC
*    AHC_IN_33_1_TYPVCC
*    AHC_IN_33_1_MINVCC
*    AHC_IN_33_1_MAXVCC
*
* Total: 12 Pin Models Created
```

These are the SPICE subcircuit models that have been created in the library to model the IBIS information. For this component, six models have been created for each of the pins that were selected. This is due to the fact that the translator will create minimum, typical, and maximum models at each power supply for any selected pin. The subcircuit name is produced from a combination of the IBIS model name, the power supply voltage, the pin number, and the minimum/typical/maximum data. For example, the following subcircuit name:

AHC_OUTI_50_3_MAXVCC

models the AHC_OUTI model at the 5V power supply for pin 3. It simulates the maximum curve data of the IBIS file.

Adding the Subcircuits to the Component Library

The subcircuits now need to be placed in the component library for use in a schematic. In the latest version of the Micro-Cap 8 component library, there are two generic subcircuits called IBIS_In3 and IBIS_Out5. These provide a basic template for most parts created through the IBIS translator. For those users that don't have these subcircuits in their component library, download the files associated with this edition of the newsletter. There are two files called IBIS.CMP and IBIS.SHP available. Place these in the main MC8 directory. Enter the Component Editor and click the Merge icon. Select the IBIS.CMP file as the file to merge. Click the Merge button. The two subcircuits will be added to the component library.

Once these two components are available in your component library, adding additional IBIS parts is easy. In the Component Editor, click on the Import Wizard icon. Enter the name of the SPICE library that was created by the IBIS translator. Click the Next button twice. Then click the Finish button. All of the IBIS subcircuits from the SPICE library will be added to the component library. These parts are now available for placement in a schematic. Since the SPICE library may contain many models, the Add Part Wizard may also be used to add just a single subcircuit from the library instead.

IBIS Circuit Example

A sample IBIS circuit is displayed in Figure 8. In this example, the 74AHC00 IBIS output buffer is used to drive between zero and four of the 74AHC00 IBIS input buffers. For the IBIS output buffers, the AHC_OUTI_50_3_TYPVCC model is used, and for the IBIS input buffers, the AHC_IN_50_2_TYPVCC is used. Each of the pulse sources is a 1MHz 5V pulse.

The resulting transient analysis is displayed in Figure 9. The plots show a single input waveform since all five are the same. Each of the five outputs is also plotted. The top plot shows the entire simulation run. The bottom plot zooms in on the falling waveform portion of the plot. As expected for each additional input buffer that the AHC_OUTI_50_3_TYPVCC buffer drives, the load on that buffer is greater, and the resulting fall time is increased.

This is a simplified example just to show the use of the IBIS buffers. A more complex simulation could use transmission line models to take into account the length of the wiring between each of the gates and would most likely need to use buffers from multiple components.

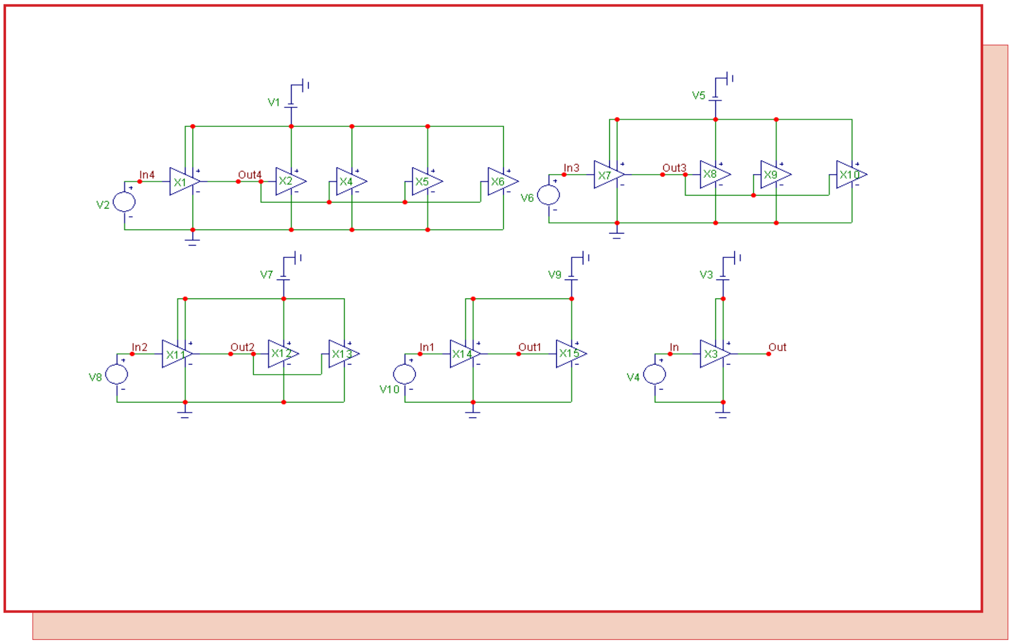


Fig. 8 - IBIS Circuit Example

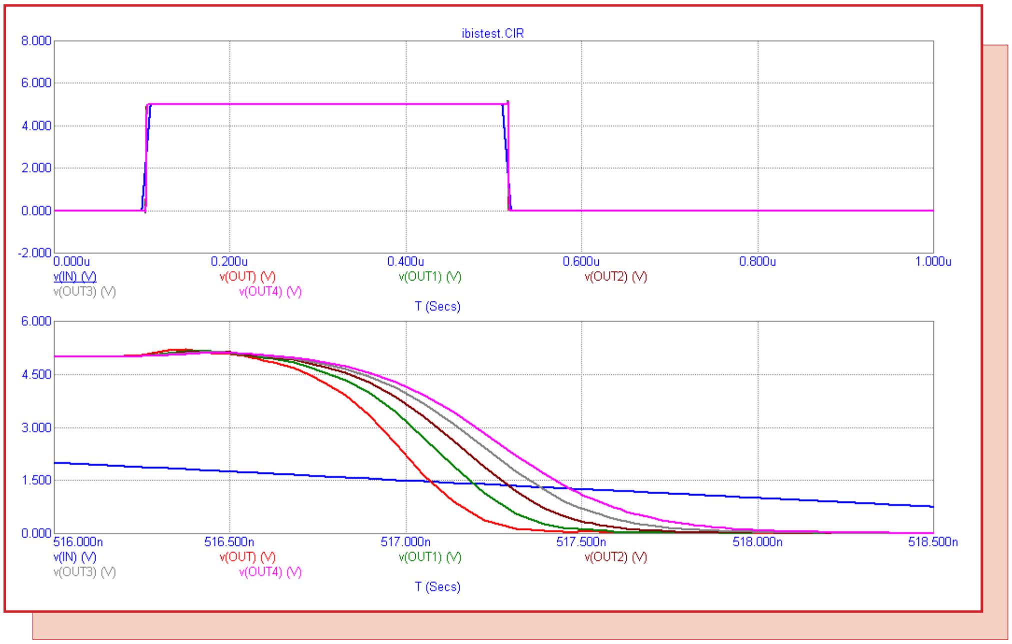


Fig. 9 - IBIS Analysis

Product Sheet

Latest Version numbers

Micro-Cap 8 .... Version 8.0.6
Micro-Cap 7 .... Version 7.2.4
Micro-Cap 6 .... Version 6.3.3
Micro-Cap V .... Version 2.1.2

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