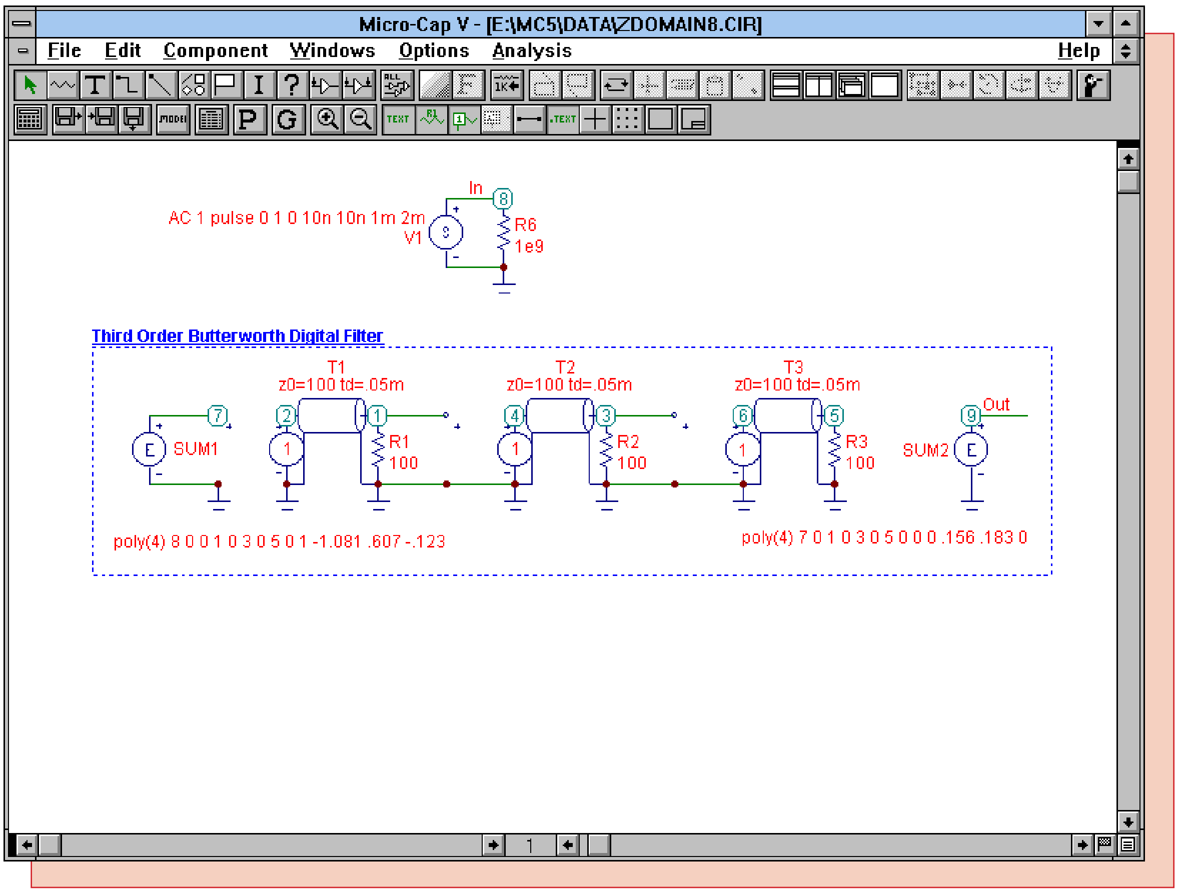
spectrum news

Applications for Micro-Cap™ Users

Fall 1996

Z Domain Simulation



Featuring:
- Tolerancing a Battery
- Z Domain Simulation
- Stepping a Triangle Source Peak
- Changing the Power Supplies of Digital Parts

News In Preview

This is the first issue of the newsletter that has been designed for on-line distribution. Let us know what you think of it and what we can improve.

Inside is a method for using a pulse source to represent a battery being toleranced, and a way to step the peak of a triangle source whose idea can be used in many other applications. One topic shows two methods for being able to simulate in the Z domain for such circuits as switched capacitor and digital filters. Finally, we include a tutorial on the different ways to change the power supplies on digital parts that covers both digital primitives and digital subcircuits.

Newsletter Contents

Book Recommendations

- *Electronic Devices and Circuits Using MICRO-CAP II*, Richard H. Berube, Merril 1992. ISBN# 0-02-309160-6
- *Electronic Devices and Circuits Using MICRO-CAP III*, Richard H. Berube, Merril 1993. ISBN# 0-02-309151-7
- *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
- *Semiconductor Device Modeling with SPICE*, Paolo Antognetti and Giuseppe Massobrio McGraw-Hill, Second Edition, 1993. ISBN# 0-07-002107-4
- *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

New Email Addresses

We can now be reached at the following email addresses.

support@spectrum-soft.com - This address is for technical support or any suggestions for the program.

sales@spectrum-soft.com - This address is for sales support and information.

parts@spectrum-soft.com - This address is to request parts that you would like to see added to future versions of Micro-Cap. These requests can either be for a generic type such as a current regulator diode or a specific part such as a 1N5283. If you request a specific part, please let us know what type it is and who manufactures it.

New Working Demo/Student Version

A new working demo has been released to replace our old demo. This demo is limited in the following ways:

- The circuit has been limited to 50 components or 100 nodes.
- Depending on the size of the circuit, the simulation may be up to three times slower than the professional version.
- The component library contains only a few hundred parts.
- Text files must be smaller than 30kB to view in Micro-Cap V.

This is a good test for evaluating the capabilities of the professional version. You may request a demo from us or download it through our web page at the address below.

This is also an excellent program for students to use. This version will be available for free to any professor who calls Spectrum Software. It will only be mailed directly to professors who can then pass it on to the students. However, anyone may download this at:

http://www.spectrum-soft.com/demo.html

Micro-Cap V Question and Answer

Caller: When I use the Copy Front Window to Clipboard command to copy one of my schematics from MC5 to Microsoft Word, the bitmap doesn't look right in the Word page. Lines are missing and the graphic looks nothing like it does in MC5. What is going on?

Tech: All of the information for the schematic is in the word file. If you zoom in on the graphic you should be able to see the schematic clearly. At normal magnification, Word will not display all of the information for speed and resolution reasons. Viewing the schematic in the Clipboard Viewer, found in Windows Main group, will also display how the graphic really looks.

Caller: When I print the same Word file on my LaserJet III, the wires and components appear as dashed lines. How do I fix this?

Tech: The problem that is occurring in this instance is that the schematic that was copied over was in color. When the LaserJet tries to print this, it tries to turn those colors into shades of gray. It does this by spacing black dots to represent the gray. Since the wires and components are only one or two pixels thick, it ends up printing dashed lines. The way to get around this is that you need to set the color preferences in MC5 for that circuit to black. Open up the Preferences dialog box, under the Options menu, and you will see preference settings that start with the circuit name. Set the appropriate ones to black. Upon reentering the schematic, all of the circuit should be black except for the text. Choose Select All under the Edit menu. Choose Color under the Edit menu. Set the color to black. When deselected, all of the text will now be black. Now perform the Copy operation, and Word should be able to print it fine.

Caller2: In MC4, there was the option to send plotter outputs to a File, but in MC5, I can find no such option. Is there a way to do this?

Tech: MC5 uses standard Windows printing routines. The only way to set plotter output to a file is through the Print Manager. In the Print Manager, choose Printer Setup under the Options menu. In the Setup, add a plotter driver to your list of devices. Then click on the Connect command button. One of the options in the Ports list box is File. Set the port to File and then close the Print Manager. Upon entering MC5, one of the printer options will be to send plotter output to a file.

Caller3: I just bought a new Powerbook to run my MC4 Macintosh version on. However, it is markedly slower running an analysis than my desktop Mac even though it is a much faster processor. Why does this happen?

Tech: The problem occurs due to one of the Powerbook's power saving features - Processor Cycling. If the Powerbook doesn't register any events such as a keyboard press or a mouse click, it will automatically slow the processor down in order to save power. When an analysis is running, none of these events are occurring, and the processor is limited. To fix this, you will need to shut off the Processor Cycling.
In Control Panels go to Powerbook.
Turn the switch to Easy.
Press down the Option key and while holding that down, switch over to Custom. There should be a new menu that refers to Processor Cycling.
Turn the cycling off and then reboot the Macintosh.

Tolerancing Batteries

For Monte Carlo analysis, the only parameters that may be toleranced are those that appear within a model statement. One component that sometimes needs to be toleranced is a battery. The battery, however, only uses a Value attribute. The way to get around this is to use a pulse source to represent a battery.

The pulse source is defined as a two level voltage source. The VZERO parameter defines the zero level, and the VONE defines the one level. The value of the battery should be placed in the VZERO parameter. The only important timing parameter for this case is P1. This is the time at which the pulse source begins to rise to its VONE level. Therefore, the value of the pulse source will be VZERO until time is equal to P1. If P1 is greater than the simulation time then the voltage will be a constant VZERO for the entire simulation, mimicking a battery. Since VZERO is a model parameter, a Lot or Dev statement may be placed after it to give it a tolerance. The other parameters of the pulse source may be any values as long as they follow the rule P1≤P2≤P3≤P4≤P5.

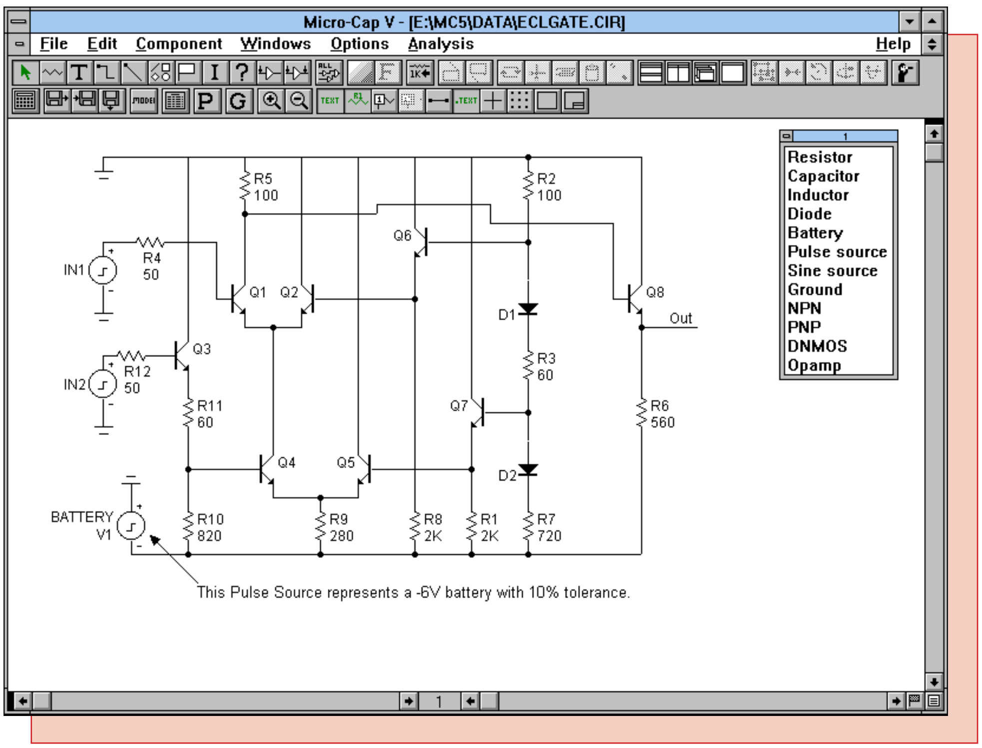


Fig. 1 - Use of the Pulse Source as a Battery

Figure 1 displays the sample circuit ECLGATE.CIR in which a pulse source has replaced the battery. In this circuit, the pulse source is representing a -6V battery. Note the orientation of the pulse source. The following model statement was used.

.MODEL BATTERY PUL (VZERO=6 LOT=10% VONE=0 P1=10 P2=11 P3=50 P4=51 P5=100)

This places a 10% tolerance on the 6V value. A Monte Carlo analysis may now be run on this circuit.

Figure 2 displays the results of a 100 run Monte Carlo analysis. The time range for this simulation was 20ns which falls well within the range of 10s set by the P1 parameter. Figure 3 displays the Monte Carlo results in histogram form showing the effect the battery variation had on the Out node.

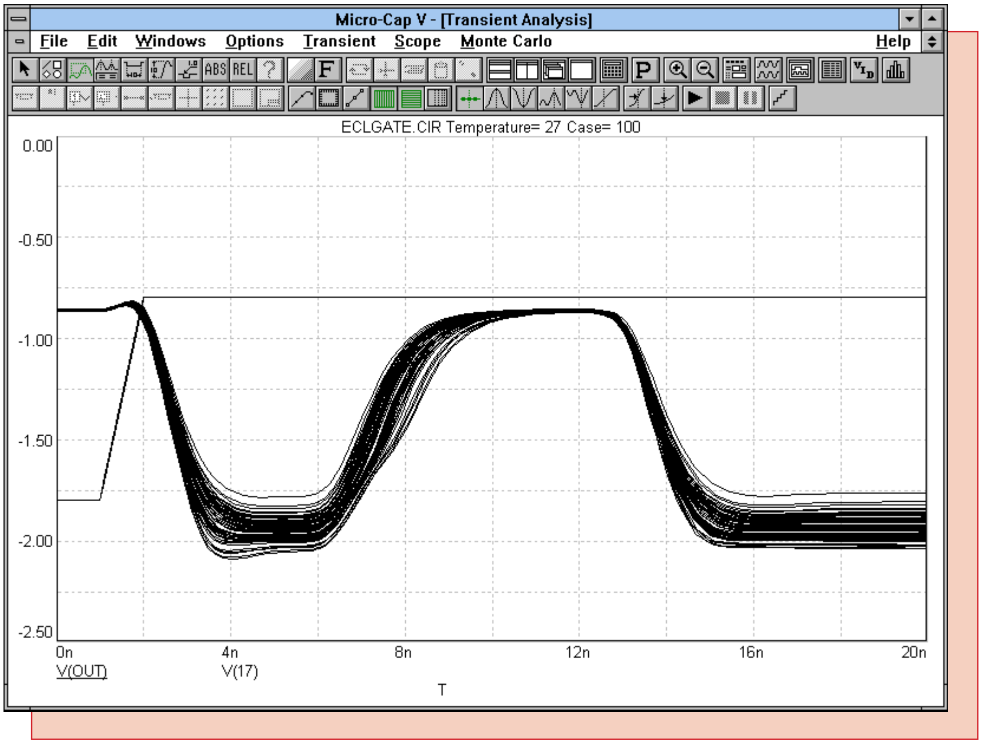


Fig. 2 - ECLGATE.CIR Monte Carlo Results

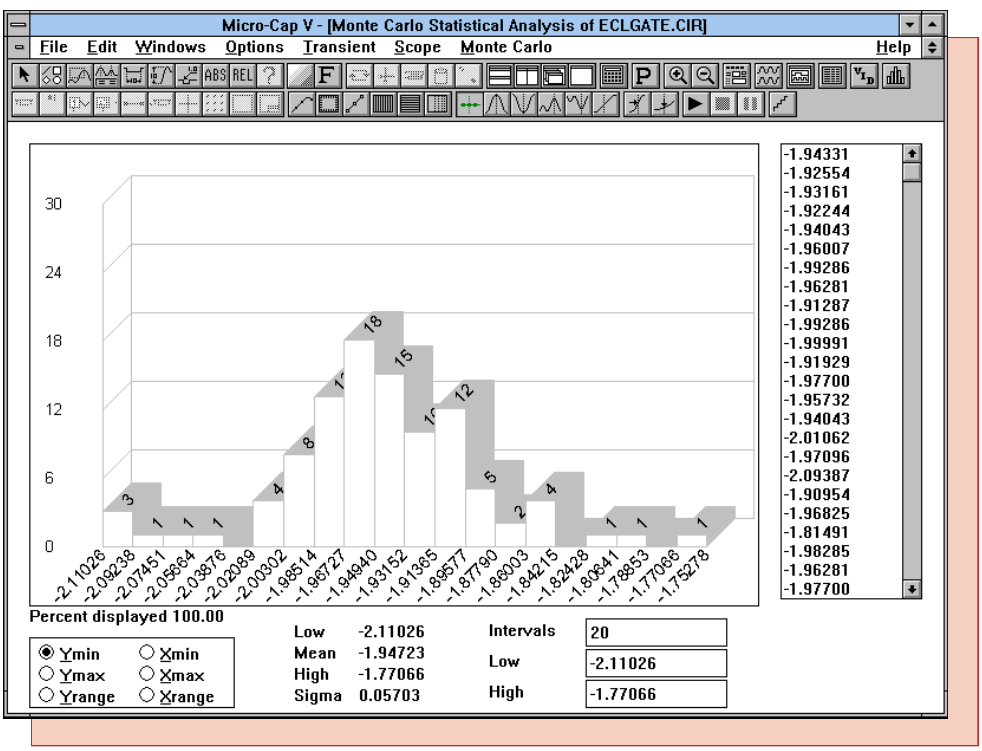


Fig. 3 - ECLGATE.CIR Histogram Results

Z Domain Transfer Functions

The Z transform is a means to mathematically describe a sampled data system. The Z transform is given by the following equation:

$$V(z) = \sum_{n=0}^{\infty} V(nT)\ Z^{-n}$$

z^{-n} is a delay operator, n is the number of previous unit delays, T is the sampling interval, and V(nT) are the discrete analog samples. This equation cannot be represented directly in MC5. MC5 can run its equivalent Laplace transform though. With a Laplace function source, any z domain transfer functions may be converted to a Laplace transfer function through the substitution:

$$z=e^{sT}$$

One of the common uses of z transforms is to simulate discrete-time systems such as digital and switched capacitor filters. The following example is adapted from an article by B. Al-Hashimi and M. Moniri in the April 11, 1996 edition of EDN titled "Spice provides z-domain circuit simulation." This example simulates a third order, elliptic, lowpass switched-capacitor filter. This filter is represented by the following z domain transfer function.

$$H(z) = \frac{0.10285(z+1)(z^2-0.70621z+1)}{(z-0.55889)(z^2-1.1579z+0.76494)}$$

Figure 4 displays the MC5 equivalent circuit.

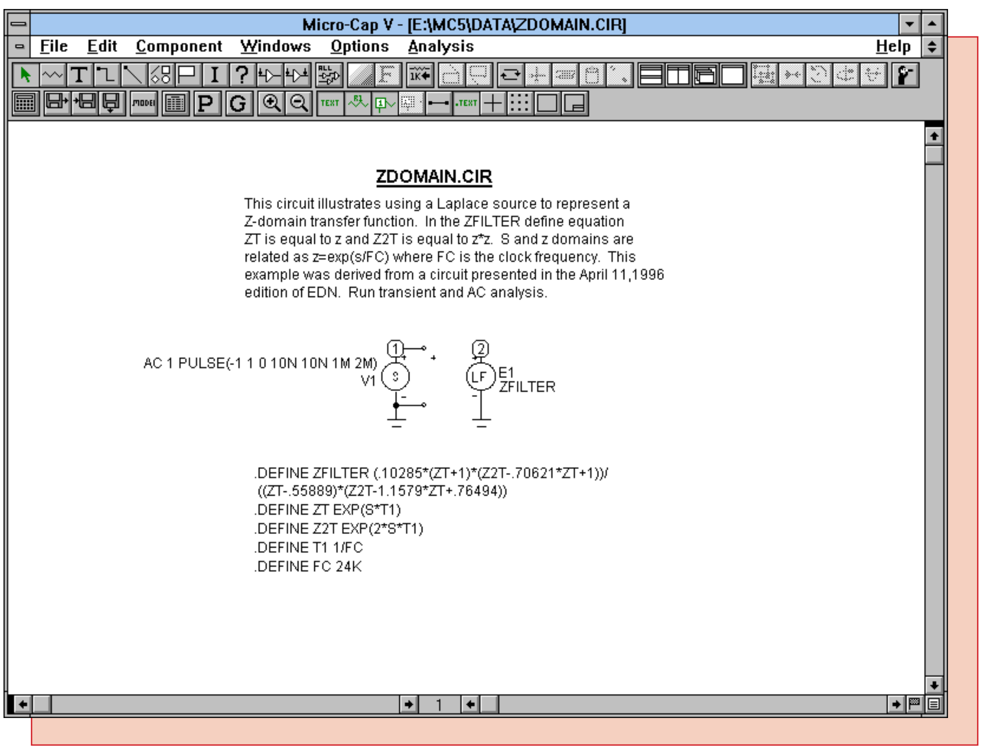


Fig. 4 - Laplace Equivalent

The circuit contains two components. One is the SPICE independent source, V, that provides the input stimulus, and the other is a Laplace function VofV source which represents the switched capacitor filter. The V source is defined as a 1ms pulse with 10ns rise and fall times and a 2ms period for transient analysis. For AC analysis, it is defined as an AC small signal source of magnitude 1V. The Laplace source's Laplace attribute is given the name ZFILTER. ZFILTER is defined through the following five .define statements.

```
.DEFINE ZFILTER (.10285*(ZT+1)*(Z2T-.70621*ZT+1))/
+ ((ZT-.55889)*(Z2T-1.1579*ZT+.76494))
.DEFINE ZT EXP(S*T1)
.DEFINE Z2T EXP(2*S*T1)
.DEFINE T1 1/FC
.DEFINE FC 24K
```

In the above define statements, FC is the clock frequency value whose reciprocal is the sampling interval, T1. T is not used as the variable in this case due to a conflict with the time variable, T. ZT is defined as the Laplace equivalent of z, and Z2T is defined as the Laplace equivalent of z^2. Finally, the transfer function is included in the define statement for ZFILTER. The + on the second line of the ZFILTER define statement is a continuation character and has no effect on the equation. Figure 5 displays the transient results of this circuit. The transient analysis clearly shows a sampling interval of 41.6us.

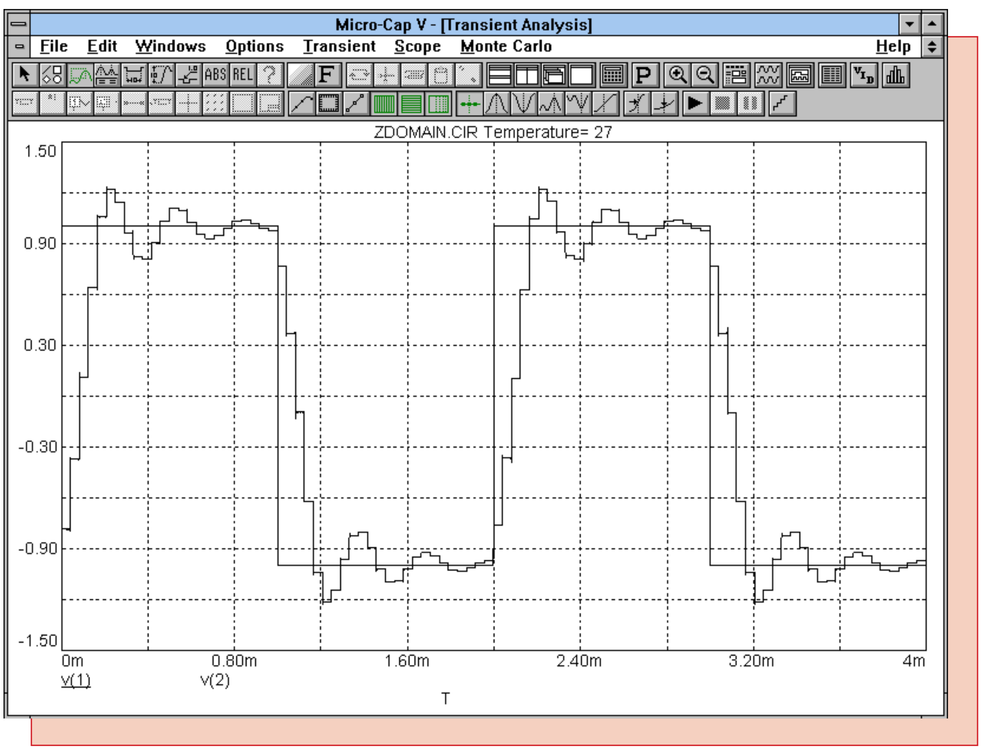


Fig. 5 - Z Domain Transient Analysis

Figure 6 gives the AC analysis results of this circuit. The specifications of this filter are: the passband is 3.2kHz, the passband ripple is 0.9dB, and the stopband attenuation is greater than or equal to 22dB at 4.3kHz. The AC analysis results match these specs.

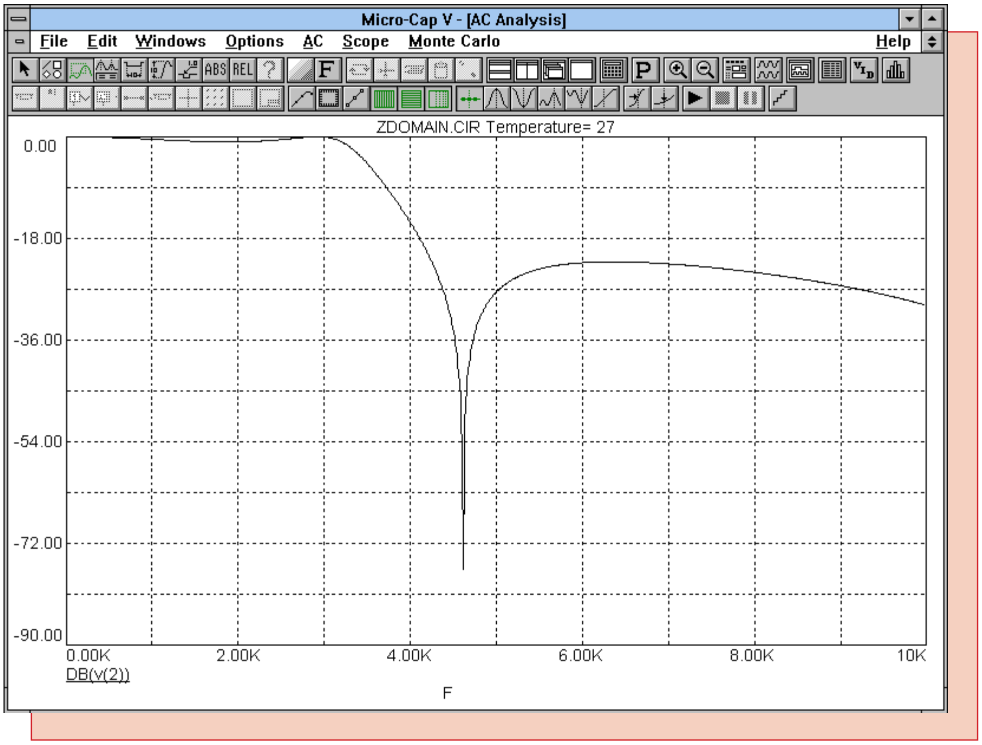


Fig. 6 - Z Domain AC Analysis

Using Transmission Lines as the Z Domain Delay Element

A second method for simulating a Z domain transfer function is to implement the z^{-1} terms with ideal transmission lines. The transmission line equivalent of z^{-1} appears in Figure 7.

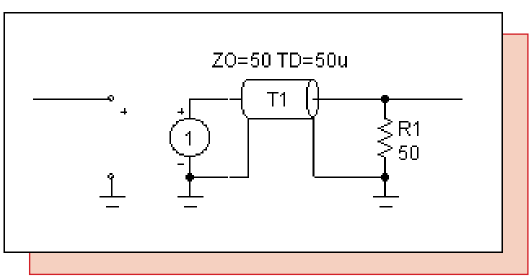


Fig. 7 - z^{-1} Transmission Line Equivalent

The transmission line is specified by its characteristic impedance, ZO, and its propagation delay time, TD. The TD parameter represents the sampling interval of the digital filter. The value of the characteristic impedance is not important as long as the line is terminated by a resistor of equal value. The source at the input of the transmission line is just a VofV dependent source with a gain of 1. This is just a buffer to isolate the z^{-1} elements when they are placed in series.

To demonstrate the use of the transmission line for Z domain analysis, we will analyze a third order Butterworth digital filter. The transfer function for the digital filter is as follows:

$$H(z) = \frac{.156z^{-1}+.183z^{-2}}{1-1.081z^{-1}+.607z^{-2}-.123z^{-3}}$$

Figure 8 is the corresponding block diagram. With the use of transmission lines, the block diagram becomes an easier tool to design from rather than the transfer function. This diagram was created using Micro-Cap's graphic elements.

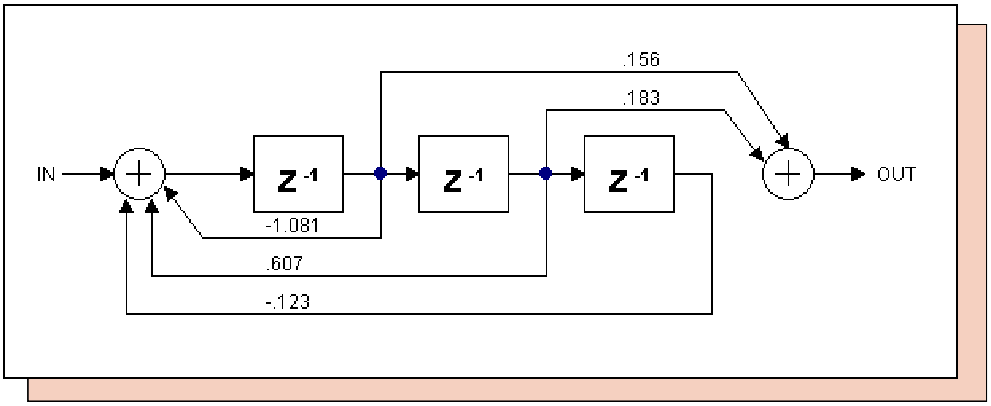


Fig. 8 - The Block Diagram of the Digital Filter

The block diagram above is simulated by the circuit in Figure 9.

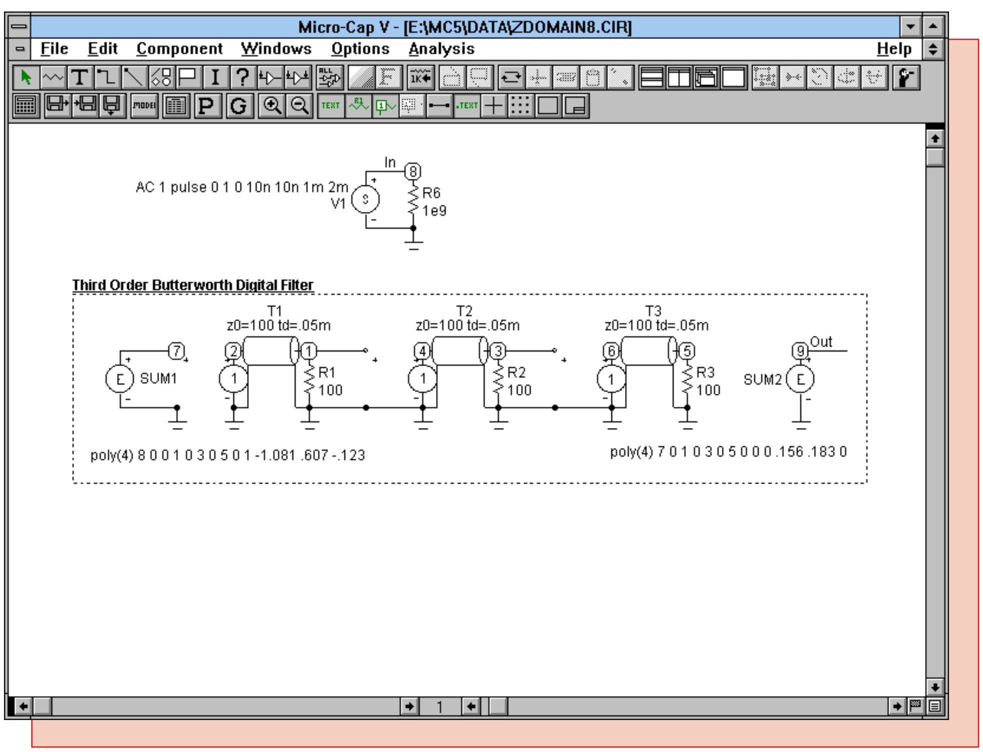


Fig. 9 - Z Domain Transmission Line Circuit

The input to the filter is again a 1ms pulse in transient analysis and a 1V AC small signal source in AC analysis. For each z^{-1} element, one of the transmission line blocks from Figure 7 has been placed in the circuit. Going from left to right, these produce the z^{-1}, z^{-2}, and z^{-3} delays. Each transmission line has ZO defined as 100 ohms, with a corresponding 100 ohm resistor terminating the line. The TD parameter of the transmission lines has been defined in order to give the filter a 20kHz sampling frequency.

The other two elements in the filter are EVofV SPICE dependent sources. These represent the two sum nodes that appear in the block diagram. These two sources sum all of the elements specified by the block diagram together by using the POLY operator. The POLY operator performs the following function:

POLY(4) 8 0 0 1 0 3 0 5 0 1 -1.081 .607 -.123

is equal to

0+V(8,0)*1+V(0,1)*(-1.081)+V(0,3)*.607+V(0,5)*(-.123)

SUM1 defines the sum at the input, and SUM2 defines the sum at the output. For a typical transfer function such as:

$$H(z) = \frac{b0+b1z^{-1}+b2z^{-2}+b3z^{-3}}{1+a1z^{-1}+a2z^{-2}+a3z^{-3}}$$

the POLY sources would be defined as:

SUM1 - POLY(4) IN 0 0 Z^{-1} 0 Z^{-2} 0 Z^{-3} 0 1 a1 a2 a3
SUM2 - POLY(4) SUM1 0 Z^{-1} 0 Z^{-2} 0 Z^{-3} 0 0 b0 b1 b2 b3

where IN is the input node number, Z^{-1} is the node number at the output of the first delay element, Z^{-2} is the node number at the output of the second delay element, Z^{-3} is the node number at the output of the third delay element, and SUM1 is the node number at the SUM1 element.

The transient analysis and AC analysis of this circuit are shown in Figures 10 and 11, respectively. In the transient analysis, note the 50us sample intervals of the V(Out) waveform. The AC analysis displays the periodic aspect of digital filters. As can be seen, the AC waveform repeats itself every 20kHz which corresponds to the sampling frequency of the filter.

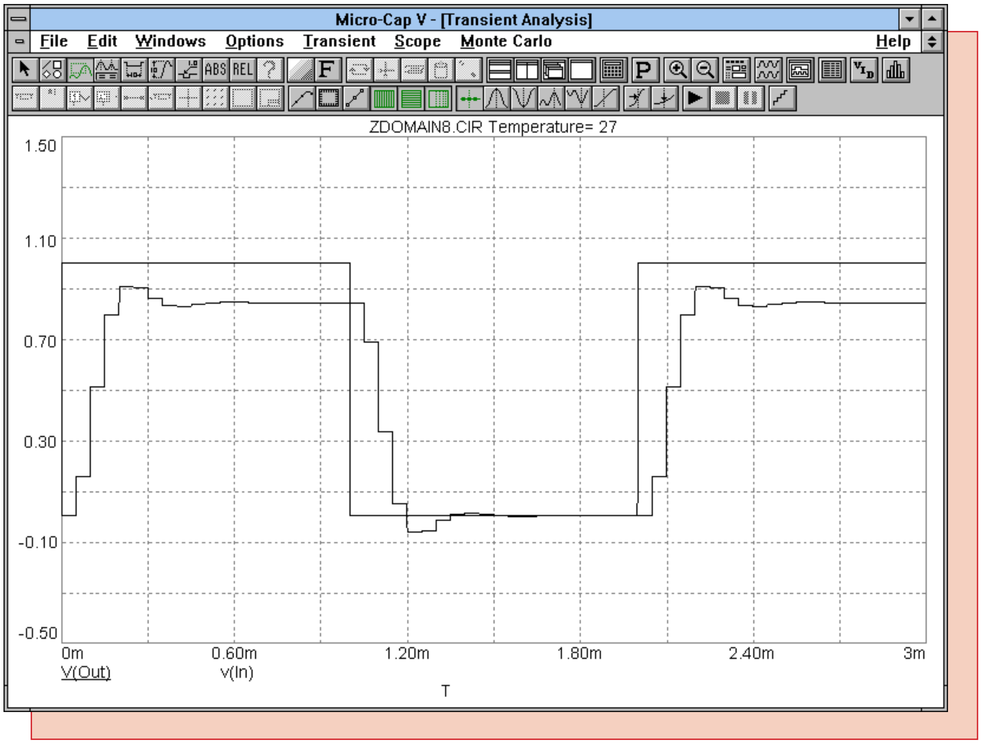


Fig. 10 - The Transient Analysis of the Transmission Line Circuit

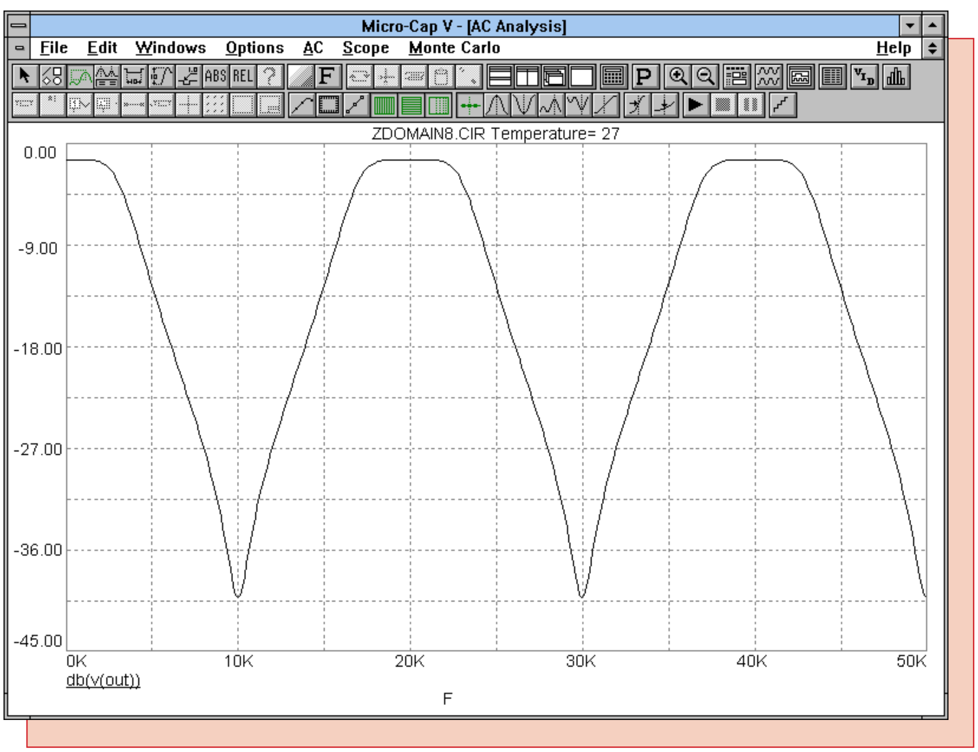


Fig. 11 - The AC Analysis of the Transmission Line Circuit

Stepping a Triangle Source Peak

Currently in MC5, it is possible to step the value of a component or a parameter in a model statement. In order to step the peak of a triangle source, two parameters of the pulse source would have to be stepped or a parameter in an equation would have to be stepped. There is not a way to do this directly in the Stepping dialog box although parameter stepping and simultaneous multiple component stepping will be added in a future version. The way to get around this is to step a component and reference its value in an equation from a nonlinear function source. The reason we don't use the pulse source model statement is that a model parameter must have a static definition when entering an analysis, but a voltage reference is a dynamic variable. The circuit used to step the peak of a triangle source appears in Figure 12.

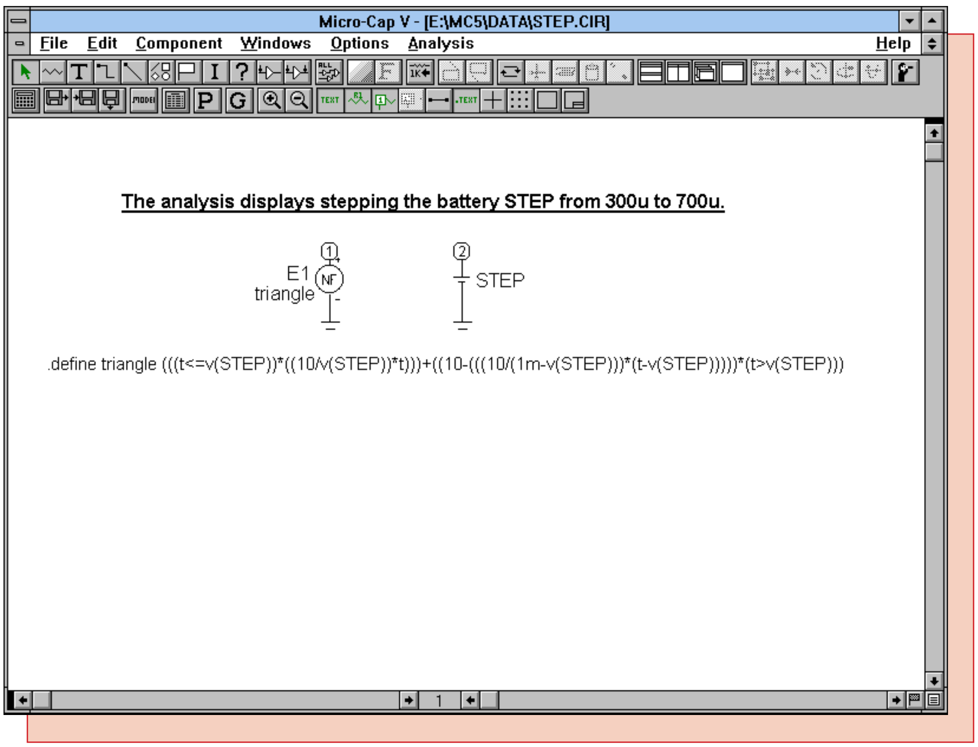


Fig. 12 - Triangle Source Peak Stepping Circuit

The circuit merely consists of a battery and a nonlinear function voltage source. The battery is the component that will actually be stepped. The value of the battery is going to represent the time when the triangle source hits its peak. The triangle source is produced by the following equation:

(((t<=v(STEP))*((10/v(STEP))*t)))+((10-(((10/(1m-v(STEP)))*(t-v(STEP)))))*(t>v(STEP)))

This triangle is actually a combination of two different line equations that are delineated by relational operators. The relational operators produce a value of 0 or 1, and let you define time ranges that a particular part of the equation will be active in. The first part is:

(t<=v(STEP))*((10/v(STEP))*t)

This states that while the time is less than or equal to the value of the STEP battery than the equation will be a line with the slope of 10/v(STEP). When the time becomes larger than v(STEP) then this equation will produce a 0.

The second part of the equation is:

(10-((10/(1m-v(STEP)))*(t-v(STEP))))*(t>v(STEP))

This part is zero until the time reaches a value greater than the battery voltage and then it becomes the active portion of the equation. It defines a line starting at 10 with a slope (10/(1m-v(STEP))) so that it will reach zero at 1ms. This source is not repetitive.

In transient analysis, the dc.value parameter of the STEP battery is now stepped from 300u to 700u in increments of 100u. The simulation time has been set to 1ms which is the period that was assumed in the triangle source equation. As can be seen in Figure 13, the peak of the triangle source is being stepped in a manner equivalent to the battery.

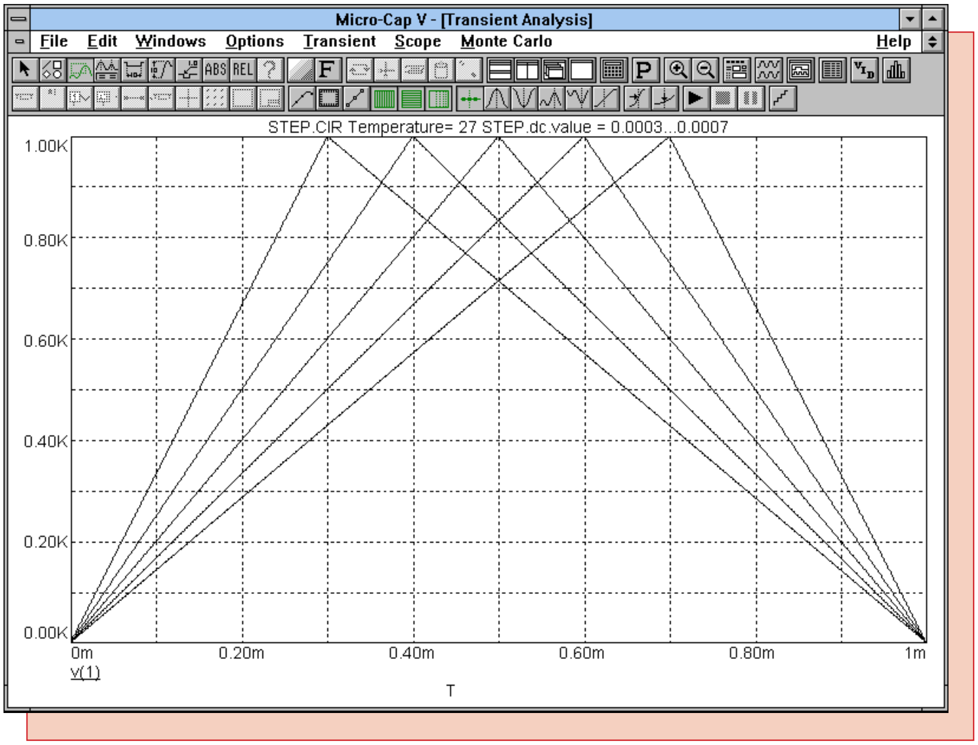


Fig. 13 - Triangle Source Peak Stepping Analysis

Changing the Power Supplies of Digital Parts

The default power supplies of all digital parts in MC5 is set at 5V. In many applications, these supplies must be adjusted for accurate simulation. There are a few methods for changing the power supplies which are dependent on whether it is local or global, on what type of digital family that is to be changed, and on whether it is a digital primitive or a subcircuit. Currently, there are two power supply subcircuits contained in the DIGIO.LIB (digital input/output library): one for the 74/TTL families and one for the CD4000 family. One thing to keep in mind when changing the power supplies is that all of the components in the digital library have their propagation delays and constraint times defined in accordance with 5V operation. Editing the digital library may be a necessity if the delays or constraints are a factor in the simulation.

Changing the Default Power Supply

The two default power supply subcircuits appear below:

```
*****74/TTL POWER SUPPLY SUBCIRCUIT*****

.subckt DIGIFPWR AGND
+  optional: DPWR=$G_DPWR DGND=$G_DGND
+  params: VOLTAGE=5 REFERENCE=0
V1 DPWR AGND {VOLTAGE}
R1 DPWR AGND 1E9
V2 DGND AGND {REFERENCE}
R2 DGND AGND 1E9
R3 AGND 0 1m
.ends

*****CD4000 POWER SUPPLY SUBCIRCUIT*****
.param CD4000_VDD=5V
.param CD4000_VSS=0V

.subckt CD4000_PWR AGND
+ optional: VDD=$G_CD4000_VDD VSS=$G_CD4000_VSS
+  params: VOLTAGE={CD4000_VDD} REFERENCE={CD4000_VSS}
V1 VDD AGND {VOLTAGE}
R1 VDD AGND 1E9
V2 VSS AGND {REFERENCE}
R2 VSS AGND 1E9
R3 AGND 0 1m
.ends
```

The method to edit these supplies is simple. First of all, open the DIGIO.LIB file that is present in the DATA subdirectory. Find the power supply subcircuits in the file. For the 74/TTL power supply subcircuit, you only need to edit the line that begins with '+ params:' and place in the appropriate value for the VOLTAGE parameter. The REFERENCE parameter is by default 0V and may also be changed. The new line should appear similar to:

```
+ params: VOLTAGE=3.3 REFERENCE=0
```

For the CD4000 power supply subcircuit, edit the lines that begin with '.param'. These two .param statements control the supply rails. CD4000_VDD controls the positive power supply and CD4000_VSS controls the ground or negative supply. The edit should appear similar to:

```
.param CD4000_VDD=15V
```

There are a couple of drawbacks to this method. The first one is that this changes the value used by all circuits that reference the default value. Any circuits that have been created previously would now use the new value which may not be desired. Using one of the older circuits would require changing the DIGIO.LIB back to its original values or to edit the circuit using one of the other methods mentioned in this article. The second drawback only occurs with the 74/TTL power supply subcircuit. Some of the families that reference this subcircuit, such as the LS family, are only designed to be run at 5V. Changing the power supply would cause these parts to run incorrectly when placed in a mixed mode configuration.

Changing the Power Supply for all CD4000 Parts in a Circuit

Instead of changing the default value, the CD4000 family offers a simple way to change the power supply of all CD4000 components in a single circuit. Notice how in the CD4000 power supply subcircuit there were the two '.param' statements. These statements set the default value in the DIGIO.LIB file. However, if a '.param' statement is present in the schematic, it will override the values present in the DIGIO.LIB file. Therefore, to change the power supply for the circuit, all one has to do is add the following statement either in the schematic or in the text area.

```
.param CD4000_VDD=10V
```

All of the CD4000 components in this circuit would then use a single 10V supply.

Using Separate Power Supplies in a Circuit

Sometimes, there may be a need to have separate power supplies for digital parts in one circuit. In this case, a new power supply subcircuit and a new I/O model statement will need to be created. The best way to do this is to copy the appropriate power supply subcircuit and I/O model to a new file, so that it won't be overwritten in the case of an upgrade. You can use these as shells to easily create new ones. The example we are going to use is to create a 3.3V power supply for the 74HC family. The new power supply will appear as follows:

```
.subckt DIGPWR3V AGND
+  optional: DPWR=$G_DPWR DGND=$G_DGND
+  params: VOLTAGE=3.3 REFERENCE=0
V1 DPWR AGND {VOLTAGE}
R1 DPWR AGND 1E9
V2 DGND AGND {REFERENCE}
R2 DGND AGND 1E9
R3 AGND 0 1m
.ends
```

The only changes made from the 74/TTL power supply subcircuit is the name of the subcircuit and the VOLTAGE parameter value.

The new I/O model for the 3.3V HC family appears below:

```
.model IO_HC_3V uio (
+  DRVH=50 DRVL=50
+  INLD=3p
+ ATOD1="ATOD_HC" ATOD2="ATOD_HC_NX"
+ ATOD3="ATOD_HC" ATOD4="ATOD_HC_NX"
+ DTOA1="DTOA_HC" DTOA2="DTOA_HC"
+ DTOA3="DTOA_HC" DTOA4="DTOA_HC"
+  TSWHL1=3.472ns TSWHL2=3.472ns
+  TSWHL3=3.472ns TSWHL4=3.472ns
+  TSWLH1=3.209ns TSWLH2=3.209ns
+  TSWLH3=3.209ns TSWLH4=3.209ns
+  DIGPOWER="DIGPWR3V")
```

The only changes made from the IO_HC model were the name of the model and the value of the parameter DIGPOWER. Note that the DIGPOWER parameter now contains the name of the power supply that was just created. Minor adjustments may need to be made on the impedances and the switching times, but the changes above will give you a solid model for simulation.

Place the name of the new library in the Nom.lib file, found in the DATA subdirectory, in the same format that all of the other libraries appear. You need to do this so that MC5 can access the library through its internal .lib statement. To use this new model, you would place IO_HC_3V into the I/O MODEL attribute of a digital primitive or if using one of the subcircuits from the digital library, you would replace all instances of IO_HC with IO_HC_3V throughout the subcircuit. Any other HC parts can still access the IO_HC model that would give them a 5V power supply.

Using an On Schematic Power Supply with a Digital Primitive

Having a digital primitive access a power supply that exists on the schematic is as simple as changing one of its attributes. Each primitive has two attributes that control the power supply nodes. The two attributes are POWER NODE and GROUND NODE. By default, these are set to $G_DPWR and $G_DGND which are the nodes that are produced by the 74/TTL power supply subcircuit. These attributes may be set to any nodes that exist in the schematic. Below is the Attribute dialog box of the U1 part from the schematic in Figure 15. The POWER NODE attribute references the Power node that is at the top of the 3.3V battery.

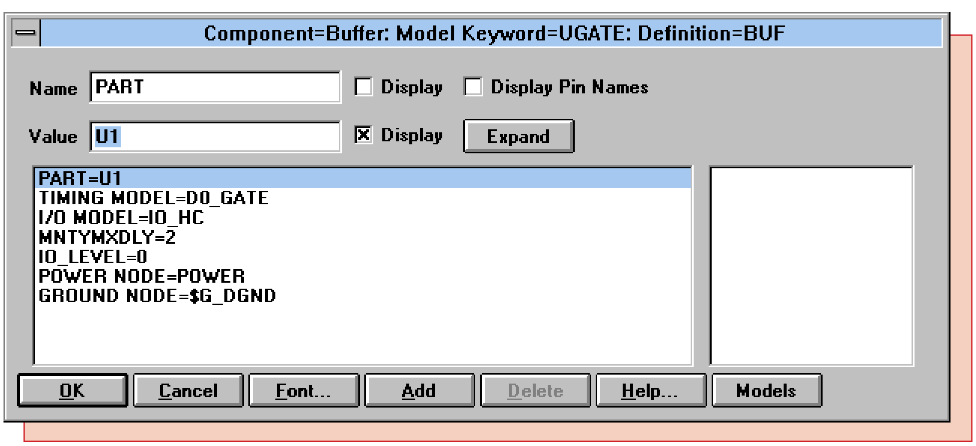


Fig. 14 - U1 Attribute Dialog Box

The node number itself may also be used in the attribute, but naming the node is a better procedure since the node numbers may change if the circuit is edited. The schematic in Figure 15 has two identical buffers except that one references a 3.3V power supply and the other references the default 5V power supply. The resistors have been added to be able to view the analog voltage display. Figure 16 displays the transient results for the output of each of these buffers.

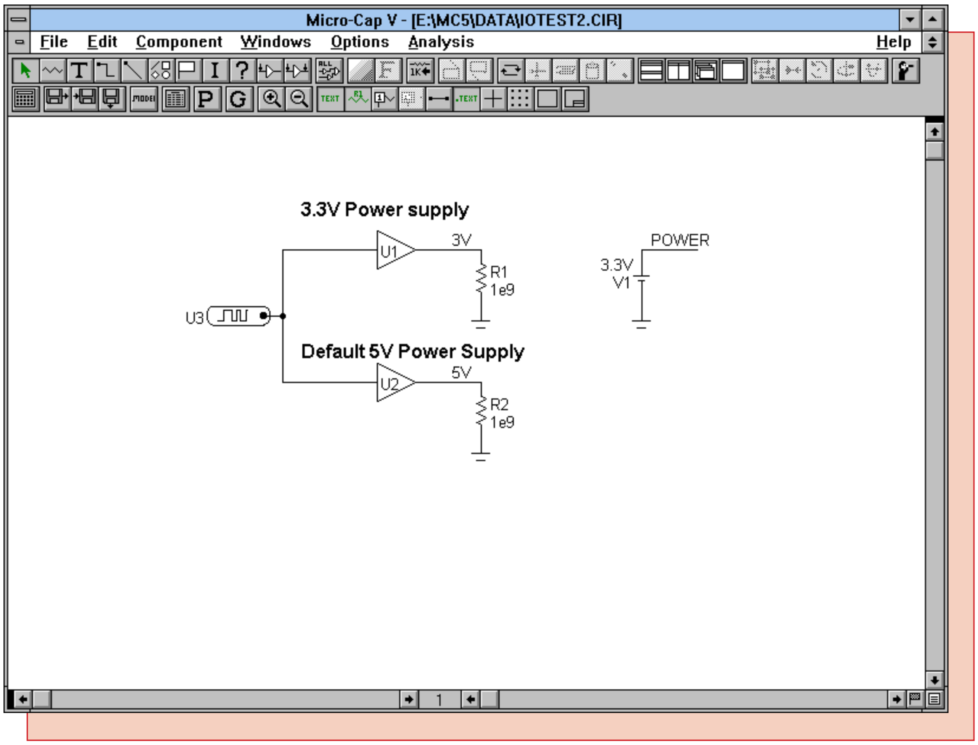


Fig. 15 - IOTEST2.CIR - Digital Primitive On Schematic Power Supply Circuit

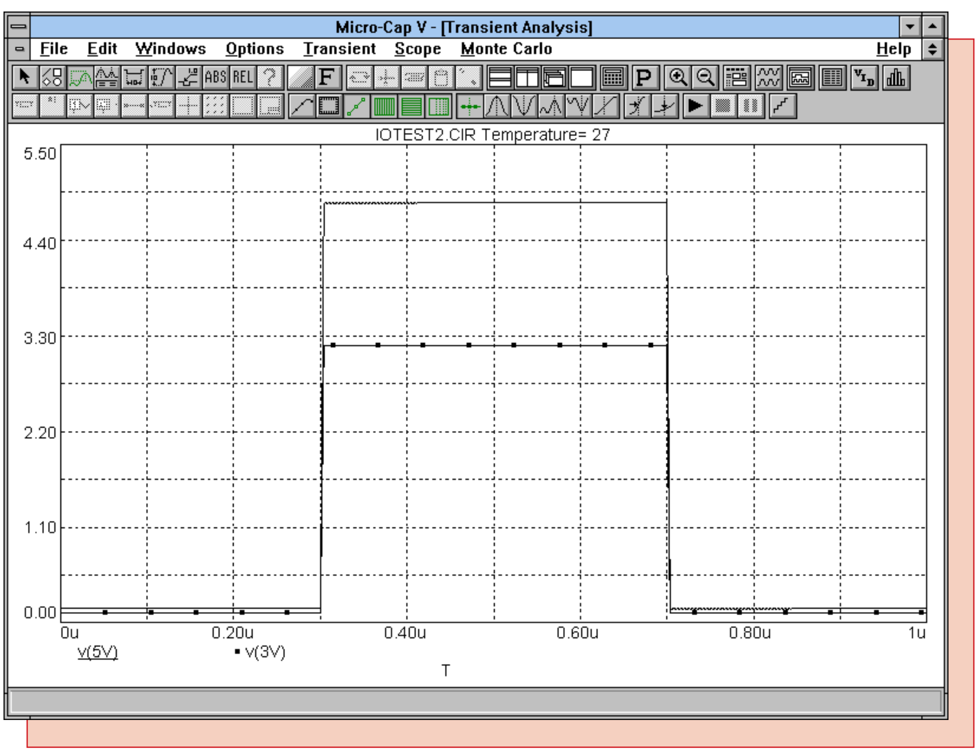


Fig. 16 - Transient Results of IOTEST2.CIR

Using an On Schematic Power Supply with a Digital Subcircuit

The power and ground nodes of all components in the digital library have been defined as optional nodes such as in the following subcircuit header.

```
.SUBCKT CD4009UB A ABAR
+ OPTIONAL: VDD=$G_CD4000_VDD VSS=$G_CD4000_VSS
+ PARAMS:  MNTYMXDLY=0 IO_LEVEL=0
```

The VDD node is the power node and the VSS node is the ground node. The OPTIONAL keyword lets you add one or more nodes to the subcircuit call. If the nodes are included in the circuit, they override the default node values which in this case are derived from the CD4000 power supply subcircuit.

In order to include these nodes in a schematic, the pins VDD and VSS must be added to the component in the Component Editor. Figure 18 displays the modified CD4009UB part within the Component Editor. The shape for the CD4009UB was changed to Inv4 so that it would have power and ground leads. Clicking in the Shape area of the Component Editor lets you add pins to the shape. The VDD and VSS pins were added and specified as analog pins since the power supply subcircuit consists only of analog components. The power and ground nodes are now ready to be accessed in a schematic.

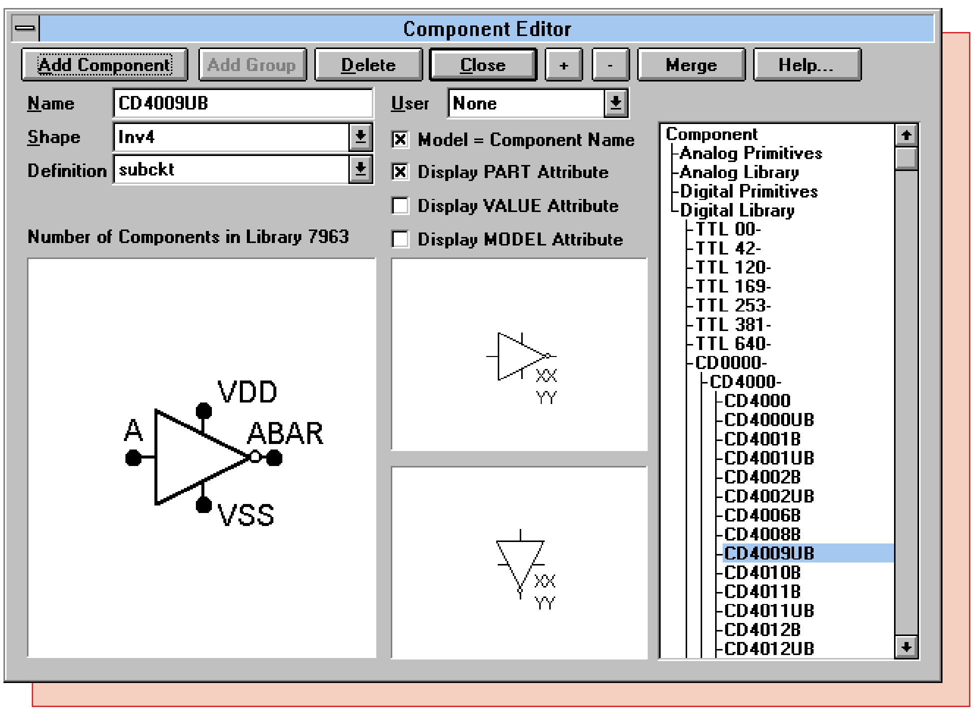


Fig. 17 - CD4009UB in the Component Editor

Figure 18 shows a sample circuit using the CD4009UB. If the VDD and VSS pins are added, they must be connected to either a voltage source or ground. The simulation, shown in Figure 19, simply steps the V1 battery from 5V to 15V in 5V increments and views the voltage waveform at the Out node.

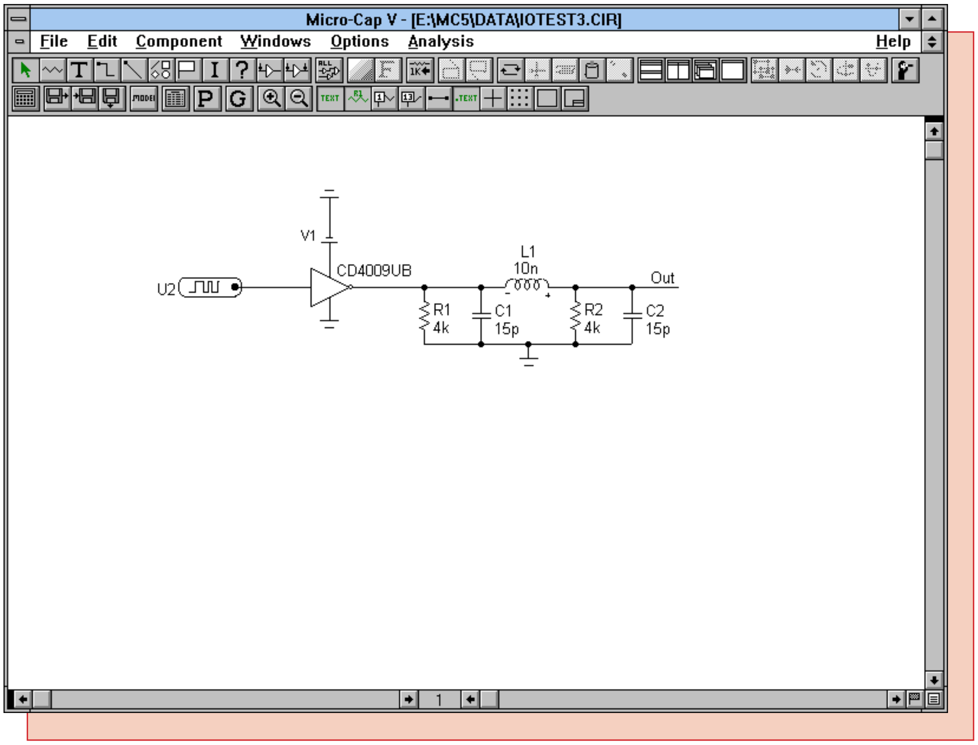


Fig. 18 - IOTEST3.CIR - Digital Subcircuit On Schematic Power Supply Circuit

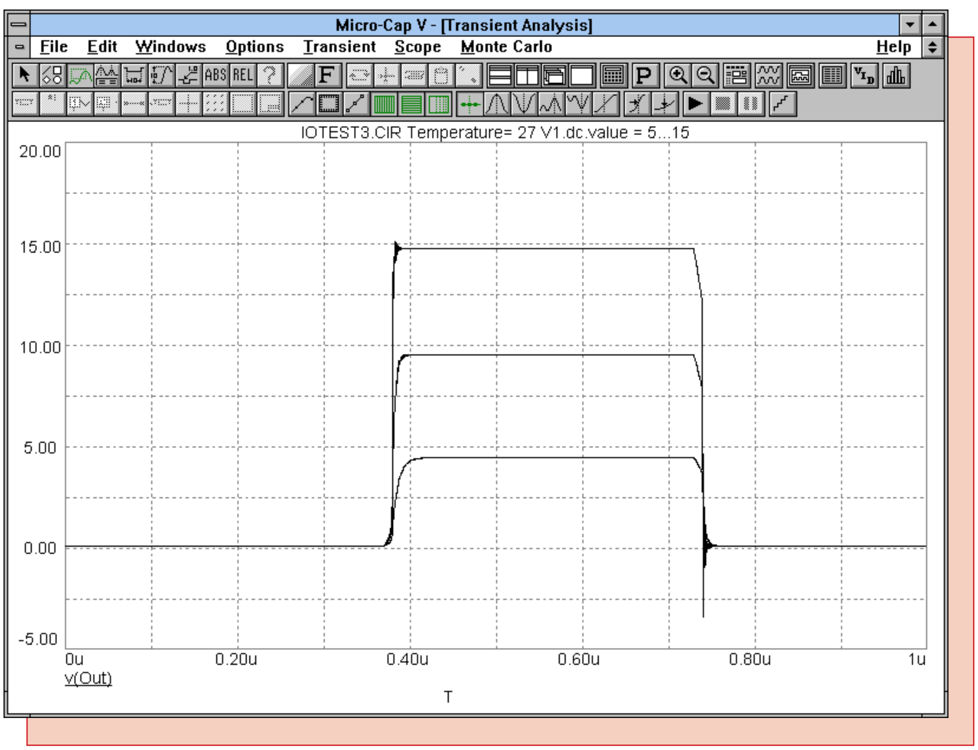


Fig. 19 - Transient Results of IOTEST3.CIR

Product Sheet

Latest Version numbers

Micro-Cap V IBM/NEC .... Version 1.2
Micro-Cap IV IBM/NEC/MAC .... Version 3.04

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

Spectrum's Products

- Micro-Cap V IBM $3495.00
- Micro-Cap V NEC $3495.00
- Micro-Cap IV IBM $2495.00
- Micro-Cap IV MAC $2495.00
- Micro-Cap IV NEC $2495.00

You may order by phone or mail using VISA or MASTERCARD. Purchase orders accepted from recognized companies in the U.S. and Canada. California residents please add sales tax.