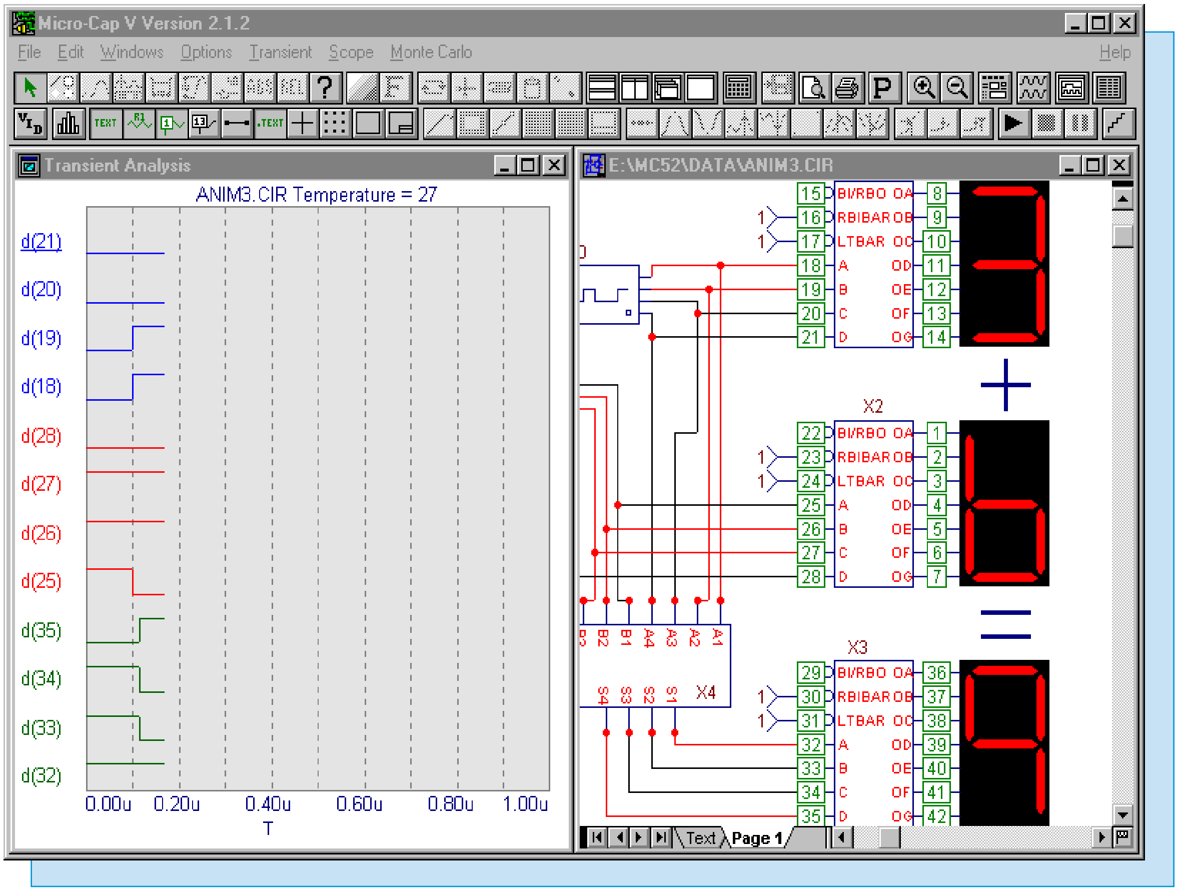
spectrum news

Applications for Micro-Cap™ Users

Winter 1999

Animate Mode



Featuring:

- Animate Mode
- Spark-Gap Macro
- Stepping the Duty Cycle of a Pulse Source
- Fourier Theory and Phase

News In Preview

This issue features a small tutorial on using the animation components and using the animate mode during an analysis. It also contains an article on a spark-gap macro model. This model accepts striking voltage and arc voltage parameters along with its passive characteristic parameters. It provides a solid model for using an arrestor. Next is an article that details how to step the duty cycle of a pulse source. The real strength in this article is learning how to use symbolic parameters for stepping multiple components, parameters, or expressions. Finally, there is an article on Fourier theory and the MC5 DSP operators. This article attempts to clear up a common misperception that deals with the phase aspect of Fourier theory.

Contents

Book Recommendations

Micro-Cap / 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
- *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
- *SMPS Simulation with SPICE 3,* Steven M. Sandler, McGraw Hill, First Edition, 1997.
  ISBN# 0-07-913227-8
- *MOSFET Modeling with SPICE Principles and Practice,* Daniel Foty, Prentice Hall, First Edition, 1997. ISBN# 0-13-227935-5

German

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

Design

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

Micro-Cap V Question and Answer

Question: Is there a way to install the MC5 Ver 2 security key driver without going through the entire installation?

Answer: Insert Disk 4 into your floppy drive. In Windows 3.X or Windows NT 3.5, go to File and Run and type in the following:

```
A:\hinstall /i
```

The computer will be busy for a few seconds and then should return with a message that states that the driver has been installed.

For Windows 9X and Windows NT 4.0, go to the Start menu and choose the Run option and type in the following:

```
A:\hinstall /i
```

Hit OK and a message will return that the security key driver has been installed.

Question: Is there a way to remove the MC5 Ver 2 security key driver?

Answer: Insert Disk 4 into your floppy drive. In Windows 3.X or Windows NT 3.5, go to File and Run and type in the following:

```
A:\hinstall /r
```

The computer will be busy for a few seconds and then should return with a message that states that the driver has been removed.

For Windows 9X and Windows NT 4.0, go to the Start menu and choose the Run option and type in the following:

```
A:\hinstall /r
```

Hit OK and a message will return that the security key driver has been removed.

Question: I'm building a schematic and I wanted to add in a few .define statements onto the schematic page. I went into text mode, clicked in the schematic, and then typed in multiple .define statements into the text dialog box. Why isn't it recognizing the statements?

Answer: The problem is occurring because multiple command statements were typed into the dialog box at a time. The text mode should have only one command statement, such as .define, .param, .ic, .model, typed in each time the dialog box is called up. In text mode, click in the schematic, type in one command statement and hit OK. Click in the schematic again, type in the second command statement and hit OK. Repeat as needed.

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 an icon or a menu item.

Connecting Nodes

The basic connection between nodes occurs through the use of the wire object. It is the obvious and often, easiest method for connecting nodes. There are two wire modes available: orthogonal and diagonal. There is no difference between these two modes except for the appearance of the wire on the schematic. The choice between the two modes would be purely for aesthetic reasons.

However, due to the size or configuration of a circuit or the fact that it may span multiple pages, the two wire modes may not be practical or available for use. For this reason, there are three other methods available for connecting components.

Naming Nodes with Grid Text

In addition to the automatic numbering of the nodes that Micro-Cap performs, a node may also be designated with a text name. If two nodes are defined with the same text name, then those two nodes will be connected on the schematic. To name a node, enter Text mode while in the schematic. Click on a wire or a pin of a component. The text dialog box will appear, and a node name can be entered. The node name must start with a letter or an underscore and must not be a reserved variable or operator name. For a text string to designate a node name, the bottom left corner of the text string must be attached to the wire or the pin of a component.

The Tie Component

Another method for connecting nodes is through the Tie component. The Tie component is available under the Component menu in the Analog Primitives / Connectors section. The Tie component is simply attached to a node just like any other component. It has one attribute which is the PART attribute. Any Ties that share the same PART attribute definition will be connected together. A Tie that does not have another matching Tie will have no effect on the node.

The .TIE Statement

The .TIE statement operates in a different manner than the previous two methods. The .TIE statement connects together all of the specified <pin name> pins of the specified <part name> parts. This conveniently connects common pins for the same type of components. It is normally used for power, clock, reset, and preset pins. The syntax for the statement is:

```
.TIE <part name> <pin name>
```

Note that <part name> is the general part name from the Component library not the PART attribute definition. Examples of this statement are:

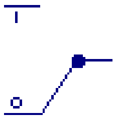
```
.TIE JKFF CLKB
.TIE LF155 VCC
```

The first example connects the CLKB pin of all JKFF components in the schematic. The second example connects the VCC pin of all LF155 components in the schematic. The LF155 will connect only the LF155 components taken from the Analog Library section of the Component menu. An Opamp component, from the Analog Primitives section, that is defined with a LF155 model statement would not be effected because the <part name> is Opamp in that case.

Animate Mode

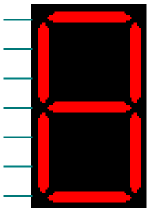
Animate mode is an analysis mode where animation objects can change their display or respond to user clicks during a simulation. Animation mode is primarily designed to operate in transient analysis with digital components. However, it is available in any analysis with any type of circuit. In AC analysis, it will only show the operating point voltages and digital states during the entire simulation. Any animation components will be fixed at their operating point state. In DC analysis, the node voltages, digital states, and animation component displays will change along with the simulated results. The strength of this mode lies in its use during transient analysis. In this mode, a single analysis time step is taken and the node voltages, digital states, and animated devices will update on the schematic. The purpose of the animate mode is to slow the simulation down to show individual node voltage or state changes and to provide diagnostic components to view simulation results on the schematic. There are three animation components: a digital switch, an LED, and a seven segment display. The description of these are as follows:

Digital Switch - This switch is designed to produce either a digital zero or one state. It has a single output at which the specified digital state will appear. During a simulation, the switch may be clicked on to toggle it between its zero and one state. The appearance of the switch's arm will change to indicate which state the switch is currently connected to.



LED - This component is designed to represent the display of a light emitting diode. It has a single input pin. Depending on the digital state or the analog voltage at the input pin, the LED will display a different color on the schematic. The colors the LED uses are defined in the Preferences dialog box under the Analysis/Performance Defaults section. In this section, there is a list of digital states that have a corresponding defined color.

Seven Segment Display - This component is designed to represent the display of a seven segment display. It has an input pin for each segment of the display. All of the inputs are active high. It is intended to work with basic seven segment decoders/drivers although any input will control the display of the corresponding segment.



These three components do not model the electrical characteristics of the parts they represent. They are only available for display purposes. The digital switch is the only one of the three that can effect a simulation. Each of these components must have an I/O model specified for them in order to work with analog components. This I/O model translates any analog voltage into its equivalent digital state.

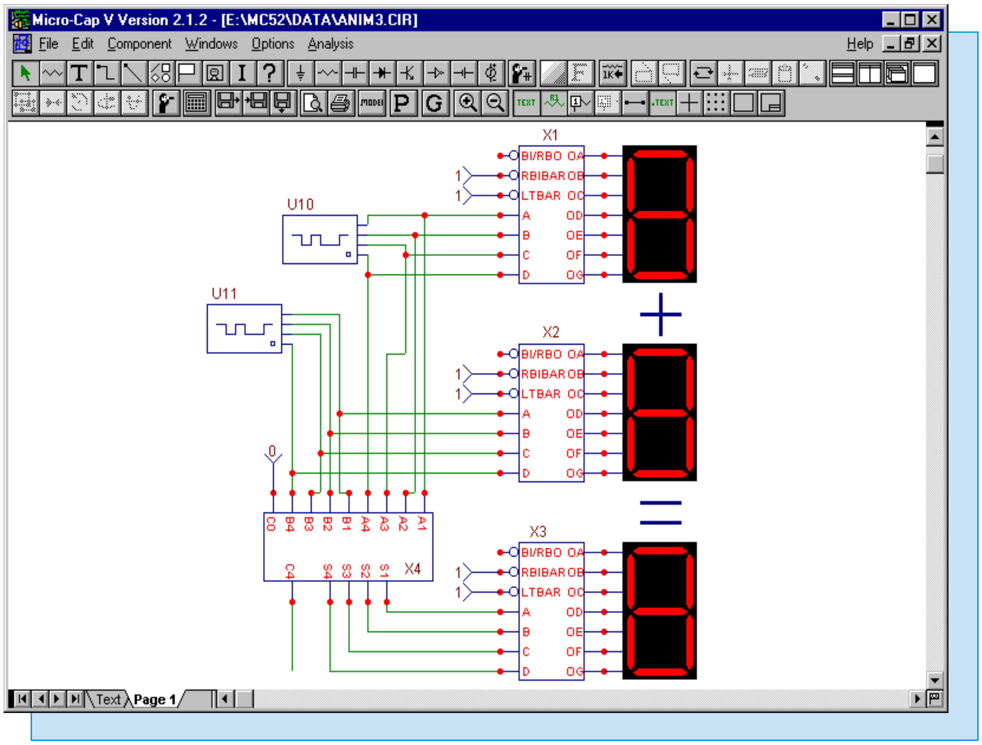
The circuit in Figure 1 uses the seven segment components to display the outputs of three seven segment decoders. The circuit consists of two 4-bit digital stimuli, three 7448 BCD to seven segment decoders, one 7483A 4-bit binary full adder, seven fixed digital components, and three seven segment displays. The digital stimuli each have their FORMAT attribute defined as 4, and the COMMAND attribute has been defined as Ina for one and Inb for the other. Ina and Inb are then defined in the text area as:

```
.define Ina
+0ns 0
+100ns 3
+200ns 1
+300ns 0
+400ns 7
+500ns 3
+600ns 5
+700ns 2
+800ns 6
+900ns 3
```

```
.define Inb
+0ns 7
+100ns 6
+200ns 1
+300ns 4
+400ns 1
+500ns 2
+600ns 1
+700ns 0
+800ns 2
+900ns 0
```

These .define statements set the hexadecimal value of the digital stimuli outputs at the specified times. The fixed digital components are used to fix the RBIBAR and LTBAR pins on the 7448 decoders to a high state and to fix the input carry pin of the 7483A to a low state. The two stimuli are then input to the 7483A adder. The output of the adder is connected to the input of a 7448 decoder. Each of the stimuli are also connected to the input of a 7448 decoder. All three decoders have a seven segment display attached to the output to display the results. The seven segments displays will display the value of the digital stimuli and the resultant sum produced from the adder.

Only a transient analysis can be run on this circuit since there are no analog devices. The simulation time has been set to 1u. Clicking on the Scope menu and then Animate Options invokes the dialog box in Figure 2. There are three options available for the analysis: Don't Wait, Wait for Key Press, Wait for Time Delay. Don't Wait runs the analysis as it normally would. Wait for Key Press forces the user to press a key to calculate the next time point. Wait for Time Delay means that the delay specified in the Time Delay text field must elapse before the next time point is calculated. For this circuit, the time delay method has been chosen with a delay of 1s.

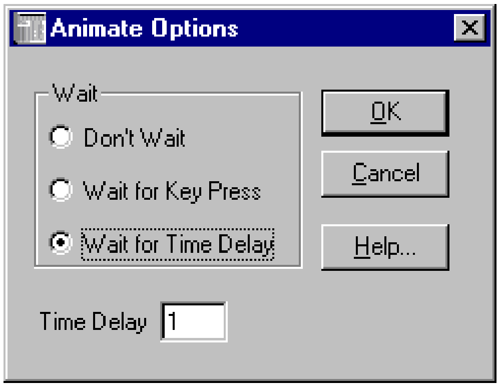


Micro-Cap V Version 2.1.2 - [E:\MC52\DATA\ANIM3.CIR]

Fig. 1 - Seven Segment Display Circuit

For animate mode, a split screen should be used to be able to view both the waveform plot and the schematic at the same time. The schematic can be moved within the window by dragging with the right mouse button. Two of the display states are shown in Figures 3 and 4. Figure 3 occurs between 100ns and 200ns when the value of Ina is 3 and the value of Inb is 6, and the third seven segment display shows the resulting sum of 9. Figure 4 occurs between 600ns and 700ns when the value of Ina is 5 and the value of Inb is 1, and the third seven segment display shows the resulting sum of 6.

The waveforms in the plot also show the value of Ina (nodes 21-18), Inb (nodes 28-25), and the resulting output (nodes 35-32) of the adder. As can be easily seen in the figures, it is much easier to view the basic output through the animation components than through the actual waveforms.



Animate Options

Wait

○ Don't Wait

○ Wait for Key Press

● Wait for Time Delay

Time Delay 1

Fig. 2 - Animate Options Dialog Box

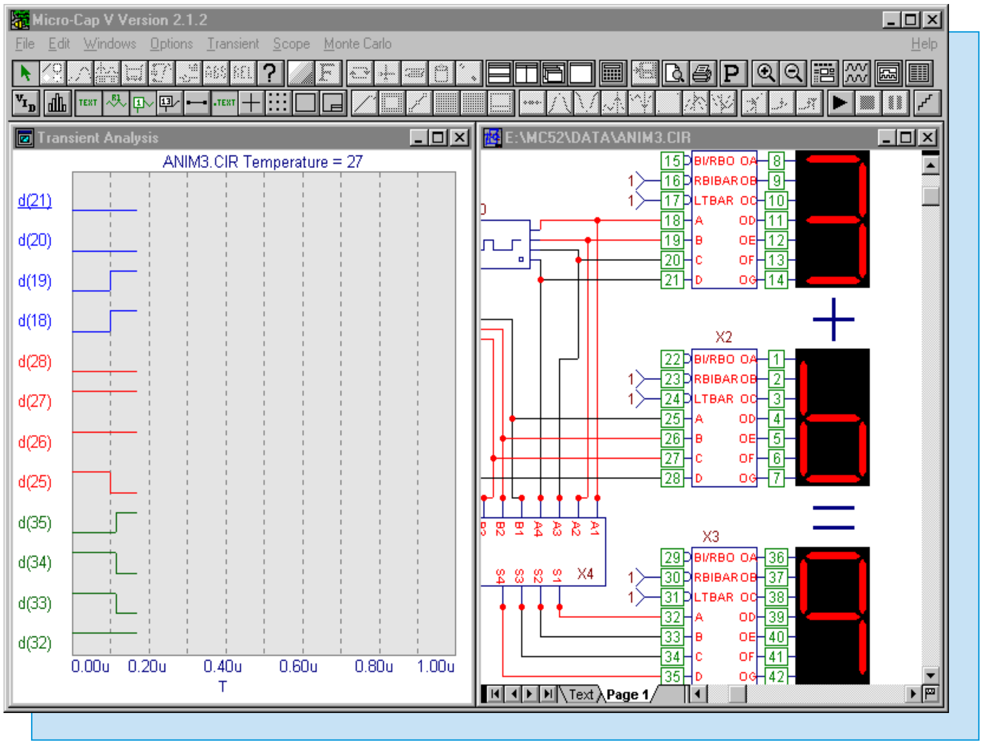

Micro-Cap V Version 2.1.2
File Edit Windows Options Transient Scope Monte Carlo Help
Transient Analysis
ANIM3.CIR Temperature = 27
d(21)
d(20)
d(19)
d(18)
d(28)
d(27)
d(26)
d(25)
d(35)
d(34)
d(33)
d(32)
0.00u 0.20u 0.40u 0.60u 0.80u 1.00u
T
E:\MC52\DATA\ANIM3.CIR
BI/RBO
RBIBAR
LTBAR
X2
X3
X4
Text
Page 1

Fig. 3 - Animation Mode Analysis Results A

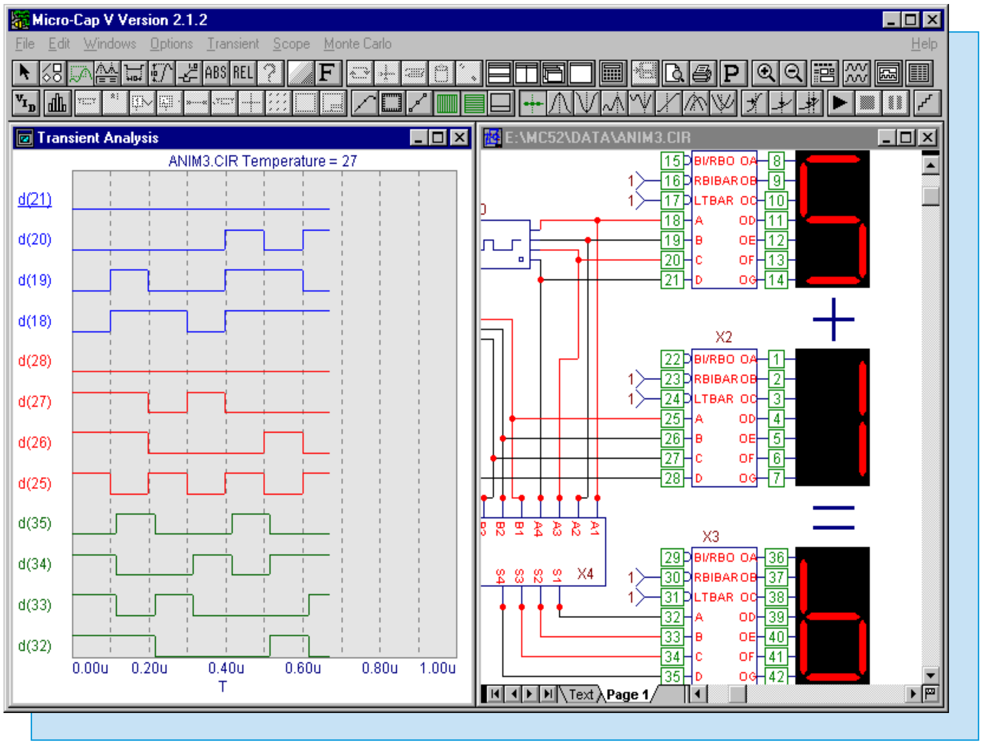

Micro-Cap V Version 2.1.2
File Edit Windows Options Transient Scope Monte Carlo Help
Transient Analysis
ANIM3.CIR Temperature = 27
d(21)
d(20)
d(19)
d(18)
d(28)
d(27)
d(26)
d(25)
d(35)
d(34)
d(33)
d(32)
0.00u 0.20u 0.40u 0.60u 0.80u 1.00u
T
E:\MC52\DATA\ANIM3.CIR
BI/RBO
RBIBAR
LTBAR
X2
X3
X4
Text
Page 1

Fig. 4 - Animation Mode Analysis Results B

Spark-Gap Macro

The spark-gap arrestor, also known as the surge arrestor, is used to block transient surges such as from lightning strikes. The spark gap is filled with an inert gas and uses two electrodes that provide a very high impedance to the circuit, and as long as the voltage across it is less than the striking voltage, the current is approximately zero. When a surge hits the arrestor, the two electrodes act as a short circuit. The voltage across the spark-gap then drops to its glow voltage and if the current continues to increase, drops down to its arc voltage. The spark-gap will continue to conduct until the current through it falls below a sustaining value. Spark-gaps can be found in such applications as aircraft ignition systems, oil burners, and magnetron protection.

The spark-gap macro circuit appears in Figure 5. This macromodel was derived based on an article by Christophe Basso titled "SPICE model simulates spark-gap arrestor" which appears in the July 3, 1997 issue of EDN. This model does not simulate the glow transition of a spark-gap. There are seven parameters for this macro. The parameters are defined as follows:

VTHRES - Voltage at which the spark-gap strikes
VARC - Voltage across the spark-gap once struck
ISUS - Sustaining current under which the arc is stopped
RNEG - Negative resistance once struck
LPL - Lead inductance
RPL - Flux loss associated with LPL
CPAR - Gap capacitance
CARC - Arc capacitance

The macro has two operating states: off and on. In its off state, the macro only has a small leakage current flowing through it, and the model is represented by the R2 resistor and C1 capacitor. The

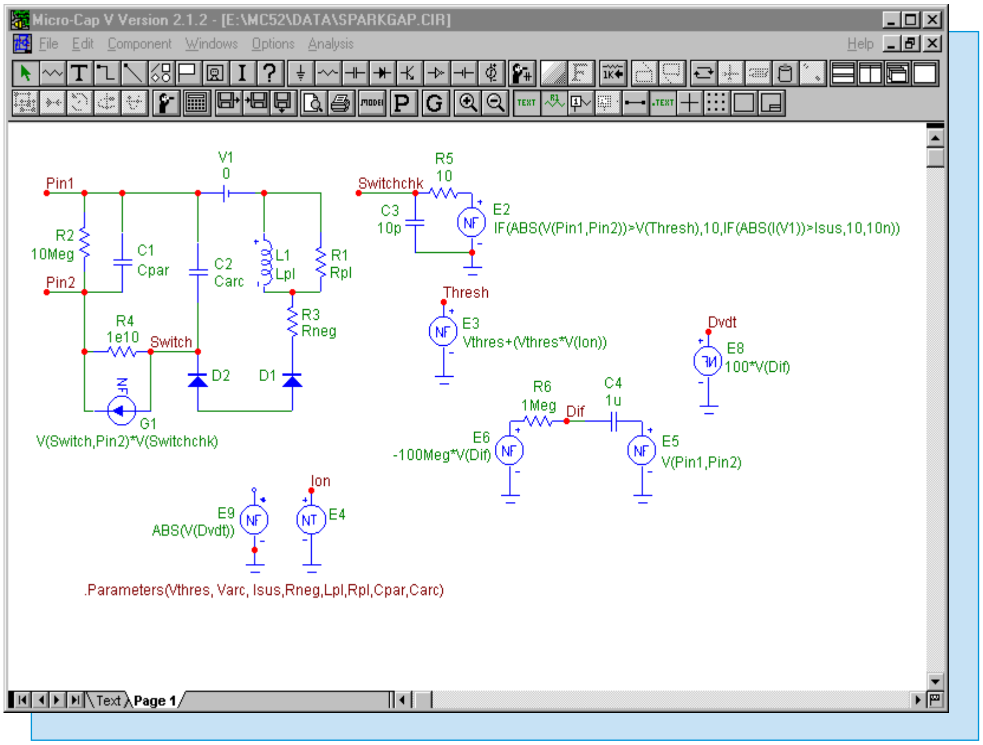

Micro-Cap V Version 2.1.2 - [E:\MC52\DATA\SPARKGAP.CIR]
File Edit Component Windows Options Analysis Help
Pin1
Pin2
V1
0
R2
10Meg
C1
Cpar
C2
Carc
L1
Lpl
R1
Rpl
R3
Rneg
R4
1e10
Switch
D2
D1
G1
V(Switch,Pin2)*V(Switchchk)
Switchchk
R5
10
C3
10p
E2
IF(ABS(V(Pin1,Pin2))>V(Thresh),10,IF(ABS(I(V1))>Isus,10,10n))
Thresh
E3
Vthres+(Vthres*V(Ion))
Dvdt
E8
100*V(Dif)
R6
1Meg
Dif
C4
1u
E6
-100Meg*V(Dif)
E5
V(Pin1,Pin2)
Ion
E9
ABS(V(Dvdt))
E4
.Parameters(Vthres, Varc, Isus,Rneg,Lpl,Rpl,Cpar,Carc)
Text
Page 1

Fig. 5 - Spark-Gap Macro

R2 resistor places a 10Meg ohm resistance across the two pins of the spark gap, and the C1 capacitor models the gap capacitance, defined by the CPAR parameter, in the off state.

When the voltage across the spark-gap reaches the striking voltage, the spark-gap transitions to its on state and begins conducting. At this point, the macro is essentially comprised of the L1 inductor, the C2 capacitor, the R1 and R3 resistors, and the D1 and D2 diodes. The L1 inductor and the R1 resistor simulate the lead inductance, defined by the LPL parameter, and the flux loss corresponding to the lead inductance, defined by the RPL parameter. The capacitance of the macro is modelled by the C2 capacitor which simulates the arc capacitance as defined by the CARC parameter. The R3 resistor models the series resistance as defined by the RNEG parameter. The D1 and D2 diodes are back to back zener diodes which have the following model statement residing in the text area:

.MODEL DCLAMP D (BV=VARC)

where the parameter VARC defines the breakdown voltage of each of the diodes. When the spark-gap is conducting, the voltage of the spark-gap will be controlled by the zener voltage VARC.

The state of the spark-gap is controlled through the switch between the nodes Switch and Pin2 that is comprised of the R4 resistor and G1 nonlinear function current (NFI) source. When the G1 NFI current is small, the 1E10 ohms of the R4 resistor act as an open circuit. Otherwise, the G1 source shorts the R4 resistor causing the back to back zener diode network to connect across the spark-gap pins. The G1 source has its VALUE attribute defined as:

V(Switch,Pin2)*V(Switchchk)

which multiplies the voltage across itself by the voltage produced at the node Switchchk. The voltage at node Switchchk is determined by a number of factors such as the dv/dt of the voltage applied to the spark-gap, the voltage across the spark-gap, and the current through the spark-gap. Nonlinear function voltage (NFV) sources and a nonlinear table VofV source are used to compute these factors in order to control the switch. The E5 NFV source has its VALUE attribute defined as:

V(Pin1,Pin2)

which produces the value of the voltage across the spark-gap. This voltage is fed into a classic differentiator consisting of the C4 capacitor, the R6 resistor, and the E6 NFV source. The E6 NFV source has its VALUE attribute defined as:

-100Meg*V(Dif)

In calculating the standard loop equations for that differentiator, the voltage at node Dif will be a scaled equivalent of the dv/dt of the voltage across the spark-gap. The basic loop equation is as follows:

V(Dif) = I*R6 - 100Meg*V(Dif)

where the V(Dif) on the left hand side is swamped out by the 100Meg*V(Dif) on the right hand side. Substituting 1Meg for R6, the equation reduces to:

100Meg*V(Dif) = 1Meg*I
V(Dif) = I/100

Due to the small value of V(Dif), the voltage across the capacitor is essentially equal to the voltage from E5 so the current can be determined by the equation:

I = C*d(V(E5))/dt

which plugged into the V(Dif) equation above gives the final result as approximately:

V(Dif) = 1E-8*d(V(E5))/dt

This resulting voltage at node Dif is subsequently fed into the E8 NFV source. The E8 source has its VALUE attribute defined as:

100*V(Dif)

which converts the voltage at node Dif into a voltage equivalent to the dv/dt of the voltage across the spark-gap in units of V/us. For example, if the voltage at node Dvdt is 1 volt, then the dv/dt of the voltage across the spark-gap is 1V/us. The E9 NFV source with its VALUE attribute defined as:

ABS(V(Dvdt))

takes the absolute value of the voltage at node Dvdt and uses this value as the input to the E4 nonlinear table source. The E4 source has its TABLE attribute defined as:

(0,0) (1U,.5565M) (10M,1M) (100M,86M) (1,217M) (10,521M) (100,956M)

This table source produces a corresponding voltage at node Ion that is dependent on the rate of the dv/dt of the voltage across the spark-gap. For example, if the dv/dt of the spark-gap voltage is 1V/us, then the table source will produce a voltage of 217mV. Currently, the table values are only set to handle transients of up to 100V/us. The table values were taken from the V_{Ignition}/dv/dt curves of a spark-gap data sheet. This voltage is then used in the E3 NFV source which has its VALUE attribute defined as:

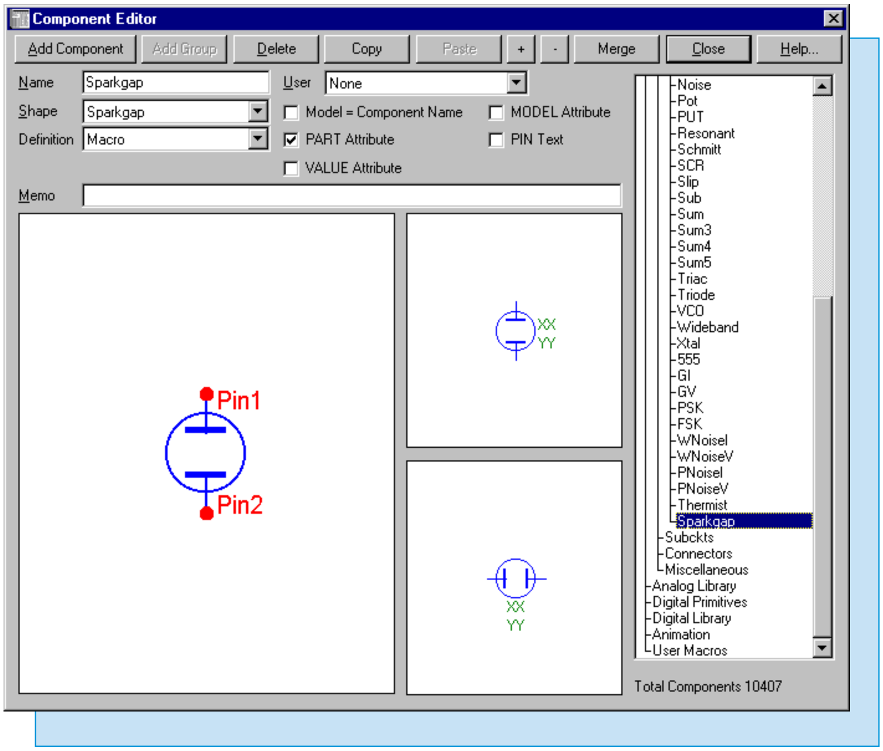
Vthres+(Vthres*V(Ion))

This source produces the actual striking voltage used by the spark-gap at node Thresh. The ignition voltage is determined from the VTHRES parameter and the voltage at node Ion. The offset produced by the Vthres*V(Ion) product represents the fact that the gas within the spark-gap needs a certain amount of time to ionize so that faster transients will reach a higher voltage before the gas is ionized and the spark-gap trips. Finally, the E2 NFV source produces the voltage at Switchchk that controls the G1 source. The R5 resistor and C3 capacitor provide a slight delay for the voltage from the E2 source. The E2 NFV source has its VALUE attribute defined as:

IF(ABS(V(Pin1,Pin2))>V(Thresh),10,IF(ABS(I(V1))>Isus,10,10n))

This equation states that if the absolute value of the voltage across the spark-gap is greater than the voltage at node Thresh, or if the voltage is less than V(Thresh) but the current through the spark-gap, which is the current through the V1 battery, is greater than the parameter ISUS, then a value of 10V will be produced at node Switchchk which will place the switch in the on state. If both of these conditions are not true then the voltage at node Switchchk will be 10nV which will place the switch in the off state.

The Component Editor settings for the spark-gap macro appear in Figure 6. The Name is defined with the same name as the macro circuit. The Sparkgap shape was created in the Shape Editor for this macro. The definition is defined as macro, and two pins are defined for the component. The two pins are Pin1 and Pin2 which are the labels on the nodes across the spark-gap in the macro circuit.



Component Editor

Name: Sparkgap

User: None

Shape: Sparkgap

Definition: Macro

□ Model = Component Name

□ MODEL Attribute

☑ PART Attribute

□ PIN Text

□ VALUE Attribute

Memo:

Total Components 10407

Fig. 6 - Spark-Gap Component Editor Settings

A test circuit for the spark-gap model appears in Figure 7. This is a simple self-relaxing configuration. The spark-gap macro has its VALUE attribute defined as:

Sparkgap(90,10,500m,-1,130n,2.5k,1p,3p)

whose values correspond to a Siemens A81-C90X arrestor. The input voltage source, V1, has been defined as a sine source with a 320V amplitude at a frequency of 50Hz. The diode is a 1N4007 model.

The resulting transient analysis appears in Figure 8. The simulation has been run for 200ms with a Maximum Time Step of 20us. However, the two waveforms have been zoomed in so that it is displaying the waveforms between 177ms and 178ms. This is the time range where the spark-gap reaches its striking voltage. As can be seen in the analysis results, the spark-gap voltage, V(5), has reached the striking voltage of 90V and trips at that point where it falls to its arc voltage value of 10V. The current through the spark-gap, I(R3), spikes up at that point. The spark-gap continues conducting until the current falls below the specified sustaining level of .5A at which point it resumes acting like an open circuit.

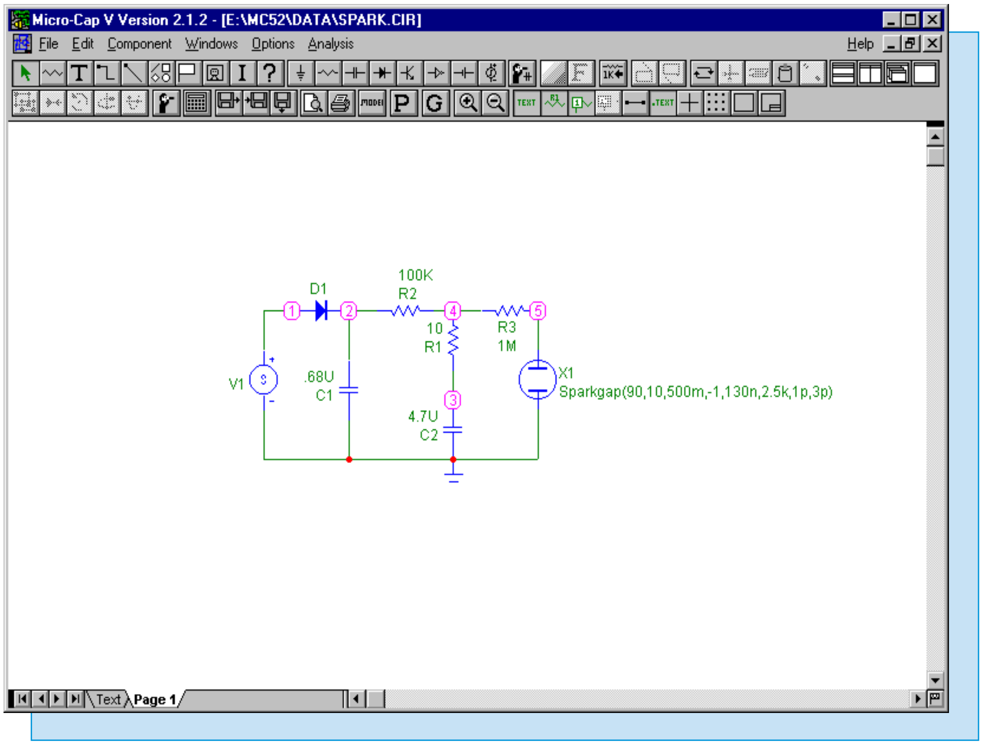

Micro-Cap V Version 2.1.2 - [E:\MC52\DATA\SPARK.CIR]
File Edit Component Windows Options Analysis Help
D1
100K
R2
10
R1
R3
1M
V1
.68U
C1
4.7U
C2
X1
Sparkgap(90,10,500m,-1,130n,2.5k,1p,3p)
Text
Page 1

Fig. 7 - Spark-Gap Test Circuit

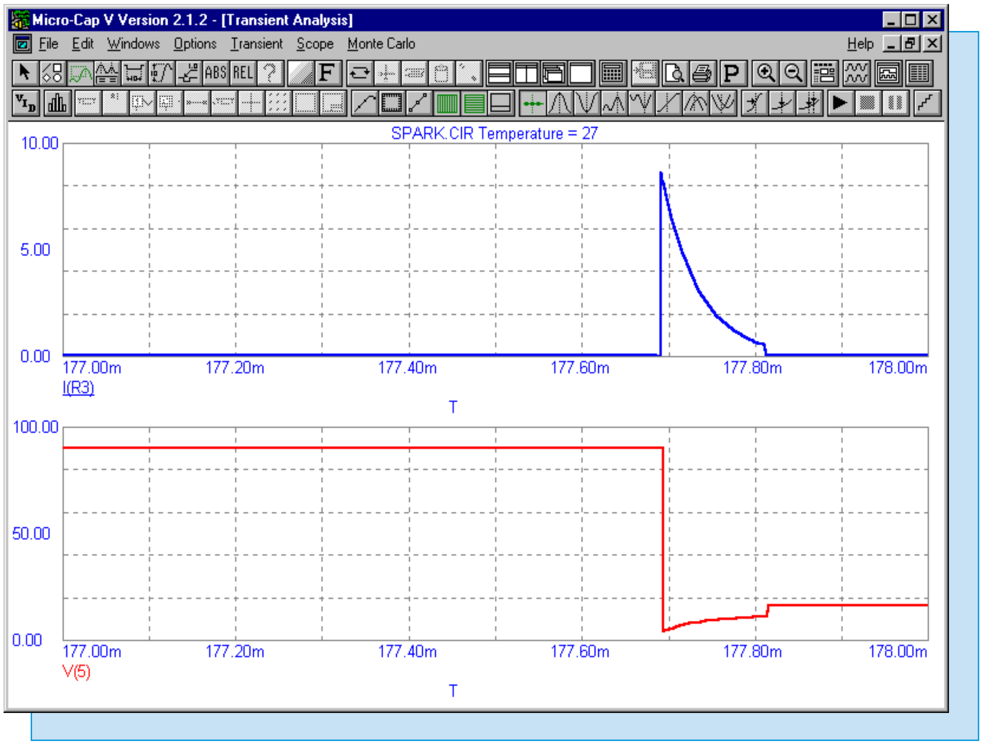


Fig. 8 - Spark-Gap Analysis Results

Stepping the Duty Cycle of a Pulse Source

There are a couple of ways to step the duty cycle of a pulse source. Since up to ten parameters may be stepped in the Stepping dialog box, you can set the stepping for each timing parameter in the pulse model statement separately. However, the timing parameters in the pulse source model statement must meet the following condition:

P1 <= P2 <= P3 <= P4 <= P5

Therefore, any timing parameter that is stepped needs to track the other timing parameters to insure that this condition is not violated. This can lead to a good deal of editing in the Stepping dialog box. An easier method is to use a symbolic parameter and to step that parameter.

The circuit in Figure 9 uses a pulse source as an input that feeds into a TTL inverter. The pulse source has been defined with the following model statement:

```
.Model VDuty Pul (Vzero=0 Vone=3.5 P1=1n P2=2n P3=Fall P4=Fall+1n P5=100n )
```

where Fall is a symbolic parameter that controls the Vone to Vzero transition of the pulse source. In this case, the pulse source will start to fall from its Vone value at Fall seconds. It will reach its Vzero value at Fall+1n seconds with the fall time always being equal to 1ns. The P1, P2, and P5 values are fixed for this circuit although the symbolic parameter can be applied to them also if necessary. The Fall parameter is then defined through a .Define statement that has been placed in the text area such as:

```
.Define Fall 40n
```

Instead of having to step both P3 and P4 in the Stepping dialog box, only the Fall parameter needs to be stepped. For this analysis, the Fall parameter will be stepped from 30ns to 70ns in linear steps of 10ns. This is done by selecting Symbolic in the Parameter Type section of the Stepping dialog box and setting the text fields as below:

Step What	Fall
From	30ns
To	70ns
Step Value	10ns

The transient analysis results of this circuit appear in Figure 10. The output of the pulse source appears in the top plot of the analysis. As can be seen in the analysis results, the duty cycle of the pulse steps from 30% to 70% as expected. The waveform at the bottom is the voltage output of the inverter.

This is a powerful stepping technique. It is not confined to a single model or component type. The same symbolic parameter may be used in as many components, expressions, or model parameters as needed. For model statements, any type of static equation may be used within the statement. Dynamic variables, such as V(1) or I(R1), are not allowed within a model statement.

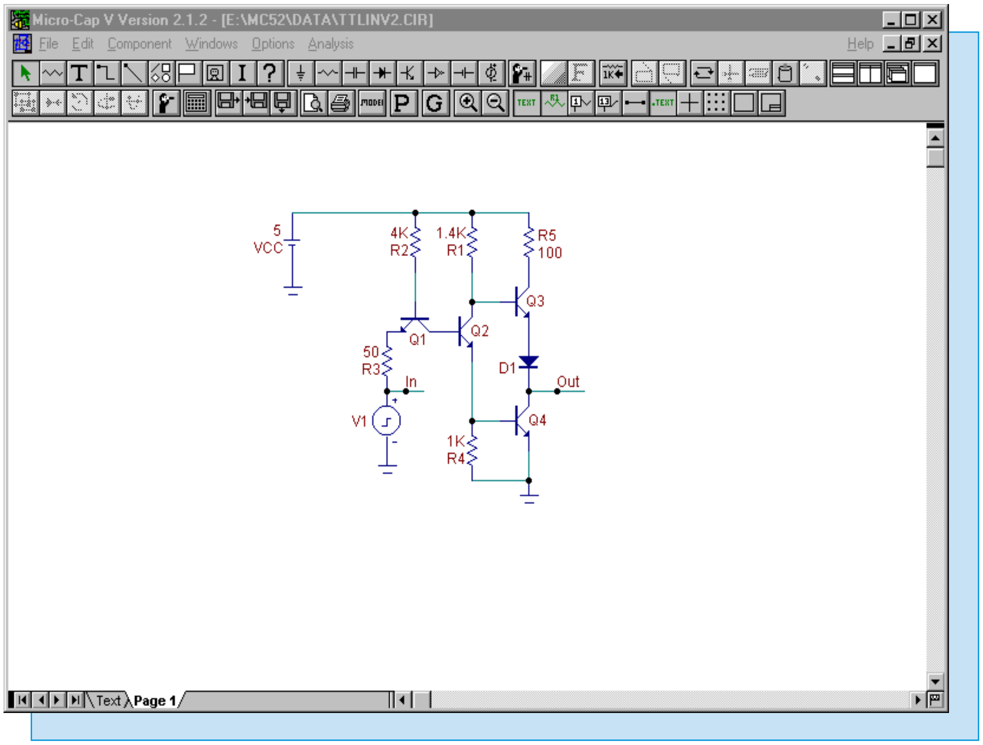

Micro-Cap V Version 2.1.2 - [E:\MC52\DATA\TTLINV2.CIR]
File Edit Component Windows Options Analysis Help
5
VCC
4K
R2
1.4K
R1
R5
100
Q3
Q1
Q2
50
R3
D1
In
Out
V1
Q4
1K
R4
Text
Page 1

Fig. 9 - Duty Cycle Stepping Circuit

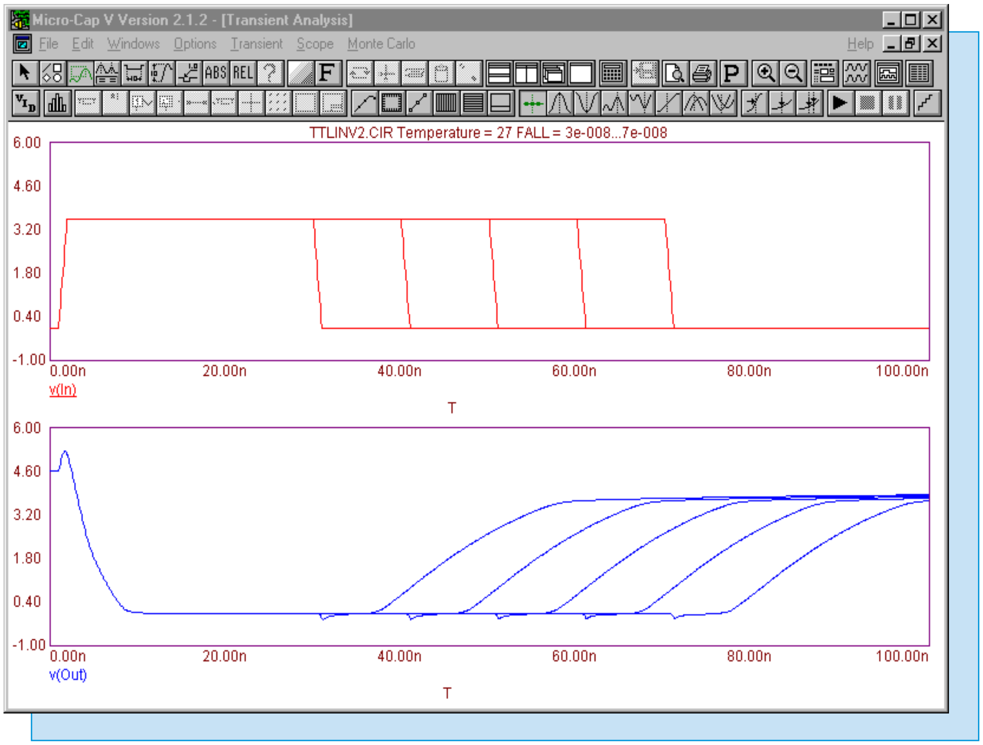


Fig. 10 - Duty Cycle Stepping Analysis

Fourier Theory and Phase

A general misconception about Fourier theory is that a Fourier analysis returns the magnitude and phase of the waveform as it would be represented by pure sine waves. However, true Fourier representation of a waveform consists of both sine and cosine waves which will give much different results, especially for phase, than if it consisted of just sine waves. The following article discusses the expressions and theory that MC5 uses for its Harm and Phase operators in DSP analysis.

The general expression for a Fourier series is as follows:

V(t) = A0 + A1*Cos(w*t) + A2*Cos(2*w*t) + ... + An*Cos(n*w*t)
B0 + B1*Sin(w*t) + B2*Sin(2*w*t) + ... + Bn*Sin(n*w*t)

where w is the fundamental frequency in radians/sec and is equal to 2*PI/T. In MC5, T will be the specified simulation time. The above equation can also be represented as:

V(t) = SUMMATION(An*Cos(n*w*t) + Bn*Sin(n*w*t)) or
V(t) = SUMMATION(Mn*Cos(n*w*t + Pn)) n=0,1,2...,n

The magnitude and phase values in the above equation are calculated by the equations:

Mn = sqrt(An*An + Bn*Bn) ;magnitude
Pn = $-\mathrm{Tan}^{-1}$(Bn/An) ;phase n=0,1,2...,n

As can be seen from the equations, both the An coefficients of the cosine terms and the Bn coefficients of the sine terms are used to calculate the magnitude and phase of the Fourier series.

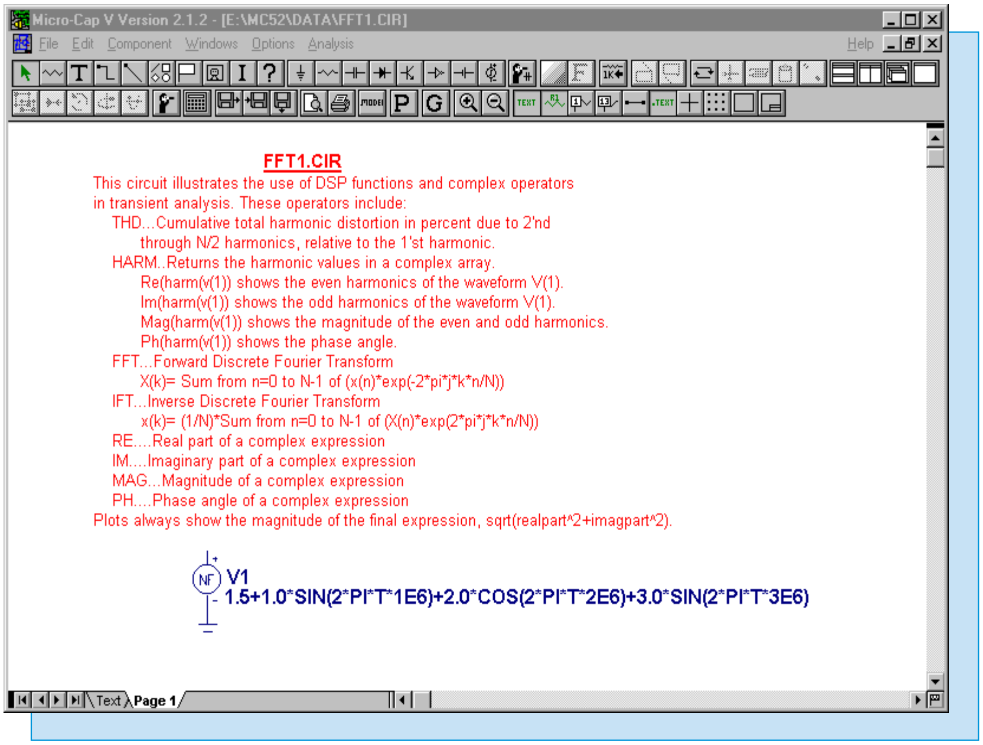
The circuit in Figure 11 consists of a single nonlinear function voltage (NFV) source. This circuit is available as the example circuit FFT1.CIR in the DATA directory. The NFV source has its VALUE attribute defined as:

1.5+1.0*SIN(2*PI*T*1E6)+2.0*COS(2*PI*T*2E6)+3.0*SIN(2*PI*T*3E6)

This equation will produce a voltage waveform which has a DC term, a pure sine term at 1MHz, a pure cosine term at 2MHz, and a pure sine term again at 3MHz.

The transient analysis of this voltage source appears in Figure 12. The three waveforms plotted were V(1), the voltage waveform, HARM(V(1)), the harmonics of the V(1) waveform, and PHASE(FFT(V(1))), the phase of the FFT spectrum of the V(1) waveform. The harmonics waveform returns the expected 1.5V at DC, 1V at 1MHz, 2V at 2MHz, and 3V at 3MHz. For the phase waveform, the sine terms return -90 degrees at 1MHz and 3MHz and the cosine term returns 0 degrees at 2MHz. The -90 degrees for the sine terms is due to the fact that the Fourier phase calculation will divide the sine wave magnitude by 0 producing an infinite value for the inverse tangent operation.

The PHASE operator was not used in conjunction with the HARM operator because the HARM operator will return the absolute value of the magnitude which will always produce positive phases. If the waveform PHASE(HARM(V(1))) was used, it would have returned the incorrect value, in terms of Fourier theory, of 90 degrees at 1MHz and 3MHz instead of -90 degrees.



Micro-Cap V Version 2.1.2 - [E:\MC52\DATA\FFT1.CIR]
File Edit Component Windows Options Analysis Help
Text
Page 1

FFT1.CIR

This circuit illustrates the use of DSP functions and complex operators in transient analysis. These operators include:

THD...Cumulative total harmonic distortion in percent due to 2'nd through N/2 harmonics, relative to the 1'st harmonic.

HARM..Returns the harmonic values in a complex array.
Re(harm(v(1)) shows the even harmonics of the waveform V(1).
Im(harm(v(1)) shows the odd harmonics of the waveform V(1).
Mag(harm(v(1)) shows the magnitude of the even and odd harmonics.
Ph(harm(v(1)) shows the phase angle.

FFT...Forward Discrete Fourier Transform
X(k)= Sum from n=0 to N-1 of (x(n)*exp(-2*pi*j*k*n/N))

IFT...Inverse Discrete Fourier Transform
x(k)= (1/N)*Sum from n=0 to N-1 of (X(n)*exp(2*pi*j*k*n/N))

RE....Real part of a complex expression
IM....Imaginary part of a complex expression
MAG...Magnitude of a complex expression
PH....Phase angle of a complex expression

Plots always show the magnitude of the final expression, sqrt(realpart^2+imagpart^2).

V1
1.5+1.0*SIN(2*PI*T*1E6)+2.0*COS(2*PI*T*2E6)+3.0*SIN(2*PI*T*3E6)

Fig. 11 - Fourier Analysis Circuit

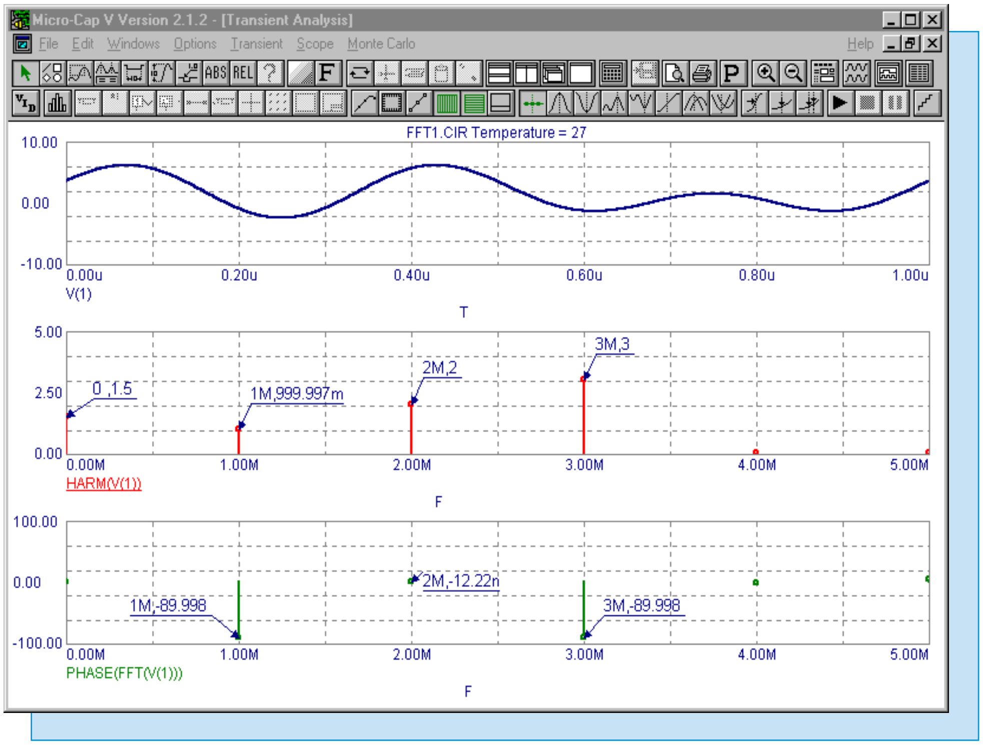


Fig. 12 - Fourier Analysis Harmonics and Phase Simulation Results

Product Sheet

Latest Version numbers

Micro-Cap V	Version 2.1.2
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	$3495.00
• Micro-Cap V LAN (single seat)	$3595.00
• Upgrade from MC5 Ver 2 to MC5 Ver 2 LAN	$100.00
• Upgrade from MC5 Ver 1 to MC5 Ver 2	$500.00
• Upgrade from MC5 Ver 1 to MC5 Ver 2 LAN	$600.00
• Upgrade from MC4 to MC5	$1000.00
• Upgrade from MC4 to MC5 LAN	$1100.00

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