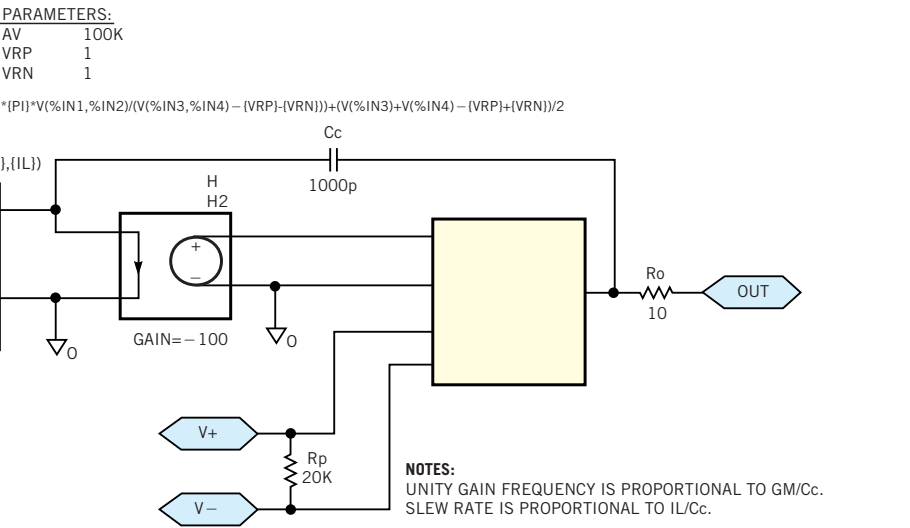
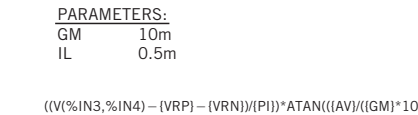
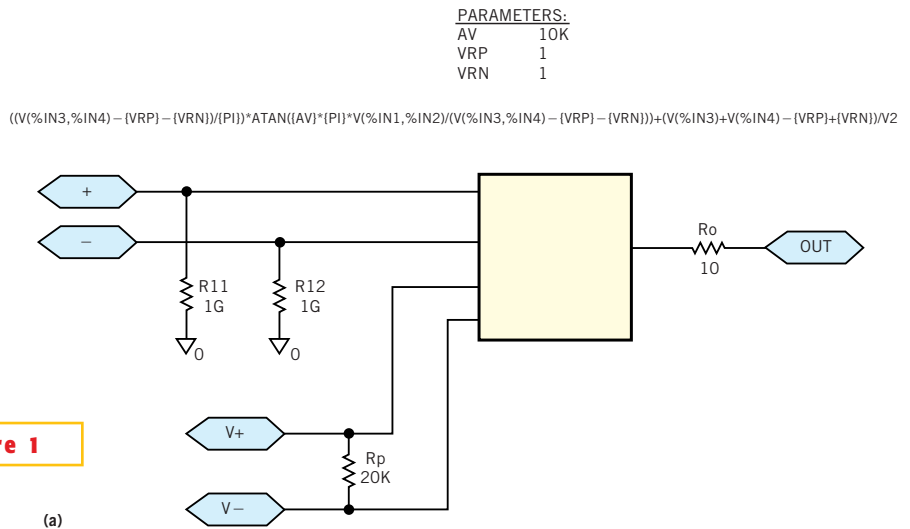


**AN EXTENSION OF MACROMODELING TECHNIQUES MAKES IT EASIER AND MORE PRACTICAL TO CREATE MODELS OF COMPLEX PARTS. THIS TECHNIQUE TAKES A MODULAR, VISUAL APPROACH THAT IS MORE INTUITIVE, ENABLING YOU TO BUILD MORE OF YOUR OWN MODELS.**

# Modular macromodeling techniques for Spice simulators

**S**PICE-BASED CIRCUIT SIMULATION has become common in the last few years as software products become more user-oriented and desktop computers become more powerful. More engineers are now able to perform more complex circuit simulations than ever before. These capabilities have the potential to provide tremendous benefits for the design and verification of electronic circuits and systems. The usefulness of Spice extends far beyond its original intended purpose of IC design.

Unfortunately, techniques for modeling electronic parts have not kept pace with other advances. Commercially available Spice programs have added greater mathematical capabil-



**A dc op-amp model uses one complex equation to describe most of the op amp's dc behavior (a). An expanded version of this model results in an ac op-amp model by breaking up the model into stages so that the compensating capacitor, Cc, can realistically model the ac characteristics (b).**

ity and flexibility. Faster computers have reduced the importance of model computational efficiency. Electronic-part functions have become more modular and complex, requiring more complex models to accurately simulate them. Users depend on manufacturers to build many of the models they use, but the manufacturers often don't provide them.

An extension of macromodeling techniques helps to alleviate this problem by making it easier and more practical to create models of complex parts. This technique takes a modular, visual approach that is more intuitive, enabling you to build more of your own models. An important element of this technique is the development of easy-to-use building blocks for the macromodeling process. The construction and successful simulation of the following modeling examples used PSpice, but you can also use other Spice programs that have analog-behavioral-modeling capability. Many types of models also require analog/digital mixed-mode simulation.

### SPICE-MODELING METHODOLOGIES

All Spice programs support the most basic types of models, including diodes, transistors, JFETs, MOSFETs, and other familiar models. These models are well-defined, well-documented, and relatively easy-to-use. The user specifies the model's electrical characteristics by entering model parameters that the program uses as constants for the model equations embedded in Spice. Most commercially available varieties of Spice now include software-modeling utilities that you can use to generate model parameters from data-sheet information.

Macromodeling techniques are neces-

```

LISTING 1—SW2ON AND SW2OFF MODELS

.MODEL SW2ON VSWITCH(
RON=0.1
ROFF=1E7
VON=2.5
VOFF=1.5)

.MODEL SW2OFF VSWITCH(
RON=0.1
ROFF=1E7
VON=1.5
VOFF=2.5)
    
```

sary for devices that Spice does not directly support. A macromodel describes the observable behavior—but not necessarily the implementation—of a device. Macromodeling usually results in a less complex model, which was important in the early days of Spice when computing power was limited. A classic example of a macromodel still in frequent use today is the Boyle op-amp model (Reference 1). This model uses conventional Spice-model elements with simple math functions to model the behavior of an op amp. The Boyle model was innovative because it did not try to include all the transistors in the op amp, which would change from one op-amp design to another, but used simplified functions and parameters to allow adjustment of model behavior. The Boyle model is also an example of a macromodel that you construct using a Spice netlist, for which lines of text and node-number connections describe all model parameters. You can find examples of op-amp macromodel netlists in your Spice library.

These netlists were the only method available to Spice users for specifying circuits and macromodels until the integration of schematic entry. Only simple, difficult-to-use math functions were available in Spice until the addition of analog behavioral modeling. Schematic entry and analog behavioral modeling, enhancements that most varieties of Spice now include, make enhanced modular-macromodeling methods possible. These methods eliminate some of the limitations and difficulties of previous techniques (see sidebar “Points to remember about macromodeling”).

### ELEMENTS OF MODULAR MACROMODELING

In addition to being more visual and intuitive, schematic-based macromodels offer more flexibility. You can represent model elements by any combination of circuit elements, Spice primitives, and math functions. Model elements can be functions of voltage, current, time, temperature, or anything else that can function as an input to the element you are modeling. Most macromodels have many possible implementations.

Figure 1a shows an example of a dc op-amp model that uses modular-macromodeling techniques. This model was built in schematic form and uses one complex equation to describe most of the op amp's dc behavior. The differential input voltage and the positive and negative power voltages are external inputs to the analog-behavioral-model equation, and the open-loop gain (AV) and positive and negative output-voltage-rail limitations (VRP and VRN) are adjustable parameters that control the model's performance characteristics. The arctangent function in the equation smoothly models the

## POINTS TO REMEMBER ABOUT MACROMODELING

Macromodels use simplified simulation elements and mathematical functions to define the behavior of a simulation model.

Macromodeling techniques are the only practical method of modeling some complex parts.

Use macro functions wherever practical to reduce modeling time and make simulations run faster and converge better.

Thorough comprehension of

the part functions you want to model is the best way to understand how and where to use simplified macro functions.

Ideally, try to build a general-purpose model that is accurate for all conditions.

If a shortage of time or resources prohibits building a general-purpose model, make sure that you accurately model the functions that are critical

for your application.

Make sure to document any known model limitations—inaccuracies and functions not included, for example—so that you can later fix or change them if the limitations affect analysis results.

Always check model accuracy using simulated test circuits that apply specified conditions.

Don't expect a new model to

work perfectly the first time unless it's very simple. A few design and testing iterations may be necessary to fix all the accuracy and convergence problems.

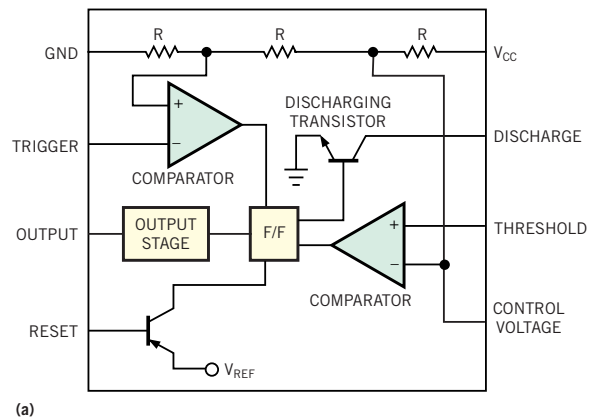
Check model behavior each time you use a model in a simulation, especially when you introduce new circuit conditions. You can discover unexpected model limitations long after you initially create a model.

voltage-rail limits while allowing high gain when the differential input voltage is close to zero (Reference 2). (For additional details about the arctangent function, see the Web version of this article at [www.ednmag.com](http://www.ednmag.com).) Although intimidating, the complexity of the arctangent function need not be a cause of great concern, because most macromodels don't require math functions of this difficulty. This dc op-amp model demonstrates excellent simulation accuracy and convergence. Once you apply this model to a model symbol, the schematic-based macromodel is ready to use in circuit simulations and as a building block for more complex macromodels.

For applications requiring accurate ac characteristics, you can expand the dc op-amp model to the form in Figure 1b. This model actually consists of stages similar to a real op amp (Reference 3), but the stages use equations. A transconductance amplifier with gain GM (transconductance=output current ÷ input voltage) models the differential-input stage. This stage also includes a maximum output-current-amplitude limit,

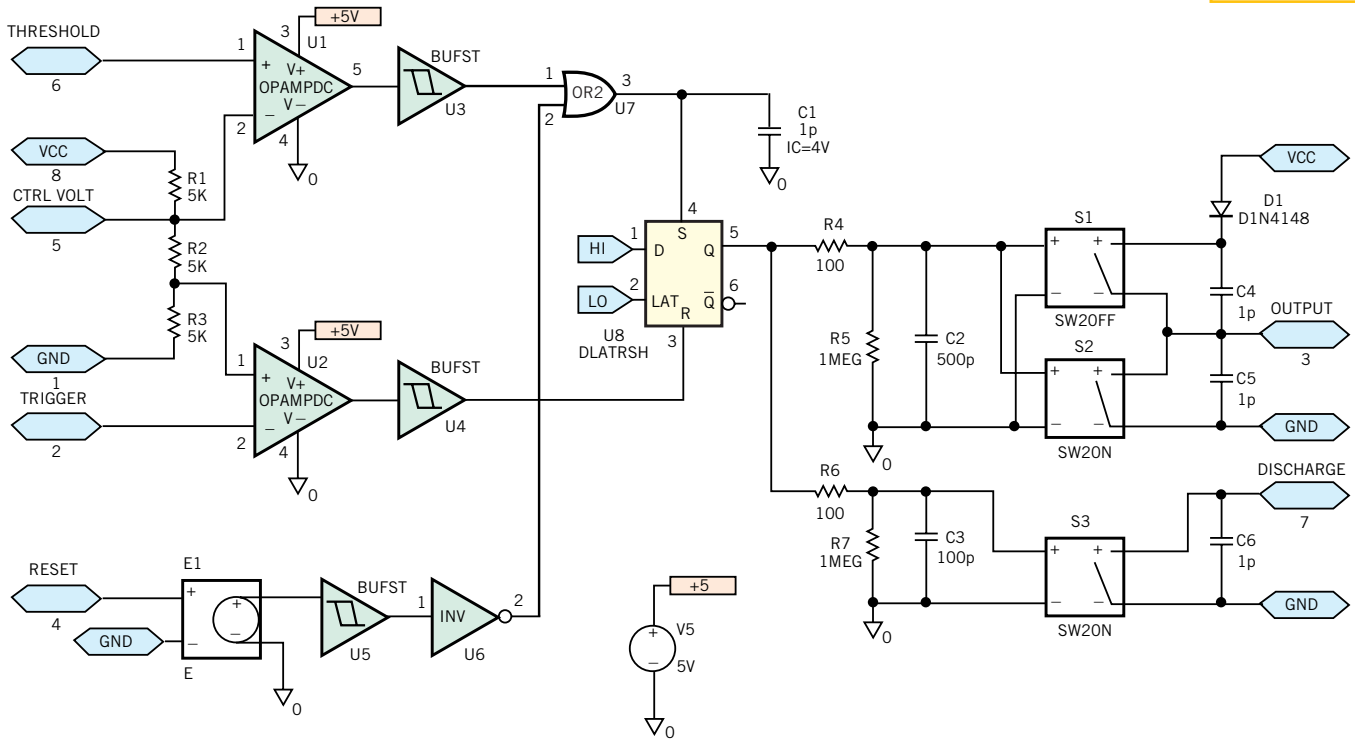
IL, to control the maximum slew rate of the model. The second stage, H2, accepts inverts, and amplifies the current output from the first stage and converts it back to a voltage. The output stage accepts and amplifies the output voltage from the second stage and limits the voltage swing using the function from the dc op-amp macromodel. All these stages are necessary, so that the compensating capacitor, Cc, can realistically model the ac characteristics. With this configuration, the ac response, specifically the unity-gain frequency, is proportional to GM/Cc, and the maximum slew rate is proportional to IL/Cc. This ac model includes most of the characteristics of an op amp that you ever need to model. Many of the omitted characteristics, such as input-offset voltage, are easy to model by adding simple elements.

It is critical to understand model limitations. All models display some differences in behavior between the simplified mathematical model in the computer and the real part. The most important task in modeling is to ensure that these limitations do not have significant effects on simulation accuracy. Ensuring this accuracy requires testing the model in simulated test circuits that apply data-sheet conditions that you use for measuring the real part. Testing sometimes reveals



(a)

**Figure 2**



(b)

**A block diagram of the classic 555 timer serves as a starting point for building a macromodel (a). The final macromodel reproduces the functions in the block diagram using functional building blocks, such as the dc op-amp macromodel OPAMPDC, which function as the comparators (b).**

unanticipated limitations that require model redesign.

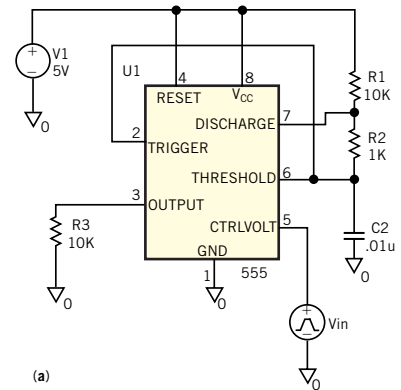
**555 TIMER MACROMODEL**

Examples of a 555 timer and a PWM modulator show you how to build models using modular macromodeling. The starting point for building a macromodel of the classic 555 timer IC is a block diagram (Figure 2a). You can reproduce the functions in the block diagram with functional building blocks and fine-tune the macromodel by testing it in simulations. In the finished 555 macromodel (Figure 2b), R1, R2, and R3 form voltage dividers that set up reference voltages for the threshold and trigger comparators. The dc op-amp macromodel, OPAMPDC, functions as the comparators. The outputs of the comparators drive a digital asynchronous latch.

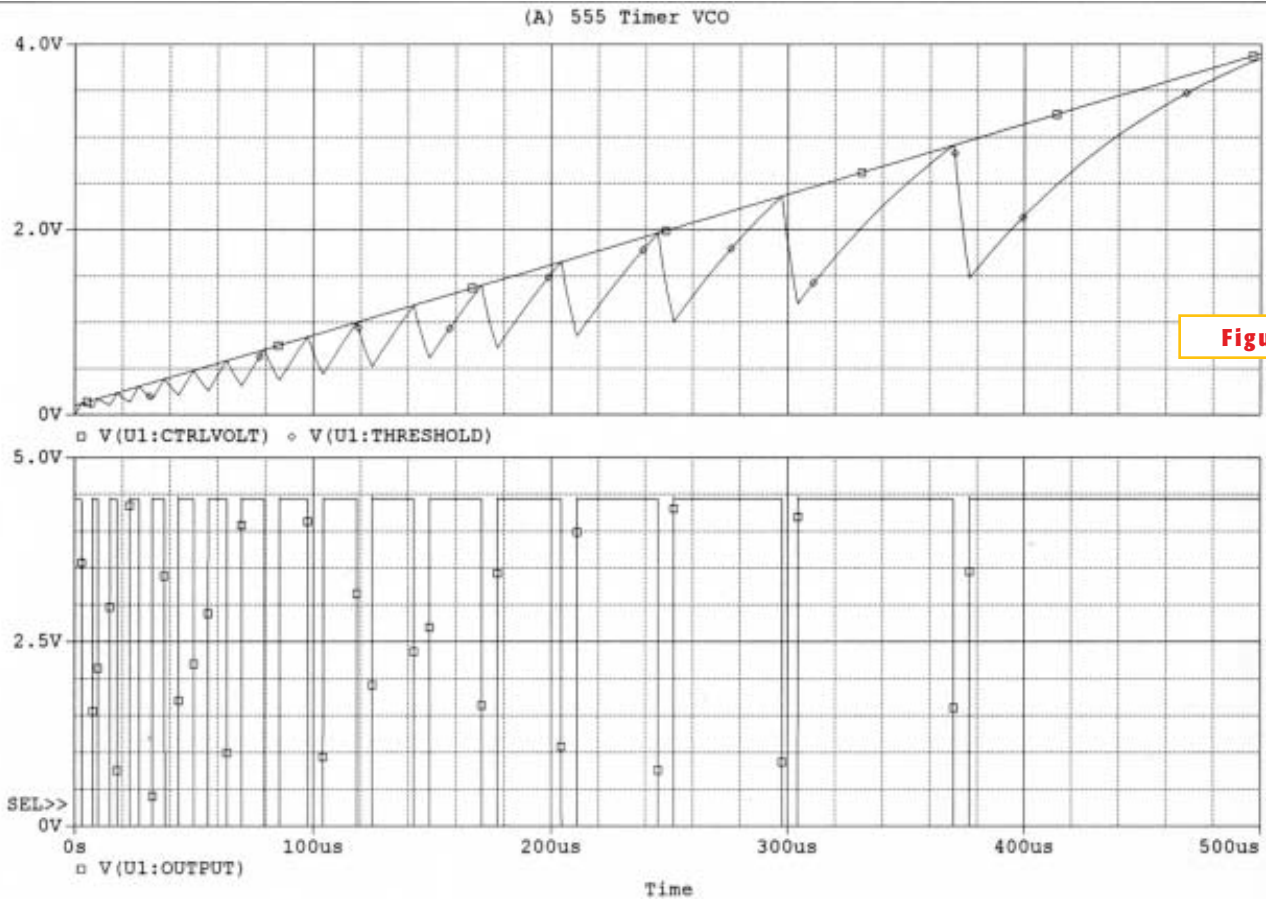
Proper grounding is an important

consideration in macromodeling. Just because most real parts have a ground pin, you should not assume that this pin always connects directly to ground, or Node 0 in Spice. In the real world, some resistance—usually small—exists between the ground reference and the ground pin of each part. Most users neglect this ground resistance in Spice simulations, but it can sometimes be critical. So, it is important to use the part's ground pin as reference for all macromodel input and output signals. This ground reference is more difficult to implement because all digital part functions in PSpice use Node 0 as ground by default. It may also be convenient to represent other types of macromodel functions with reference to Node 0. For these cases, exercise care to use functions that translate between references and keep them isolated. The OPAMPDC macromodel provides this function for input

signals because its inputs are differential and isolated from ground by 1-GΩ resistors. The voltages applied to OPAMPDC power pins affect only the output, so in the 555 macromodel, an internal 5V source referenced to Node 0 provides the V+ voltage for the op amps. Because the Reset input connects directly into a logic function, a simple voltage-con-



(a)



(b)

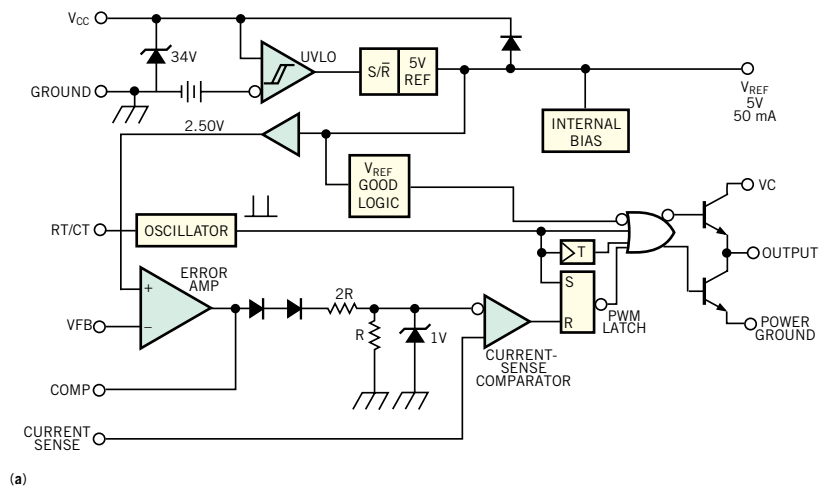
**Figure 3**

A test circuit configures the 555 timer as a voltage-controlled oscillator so that the control-voltage input determines the output frequency (a). The simulation results show the control-voltage input and the resulting threshold voltage in the top plot and the change in output frequency in the bottom plot (b).

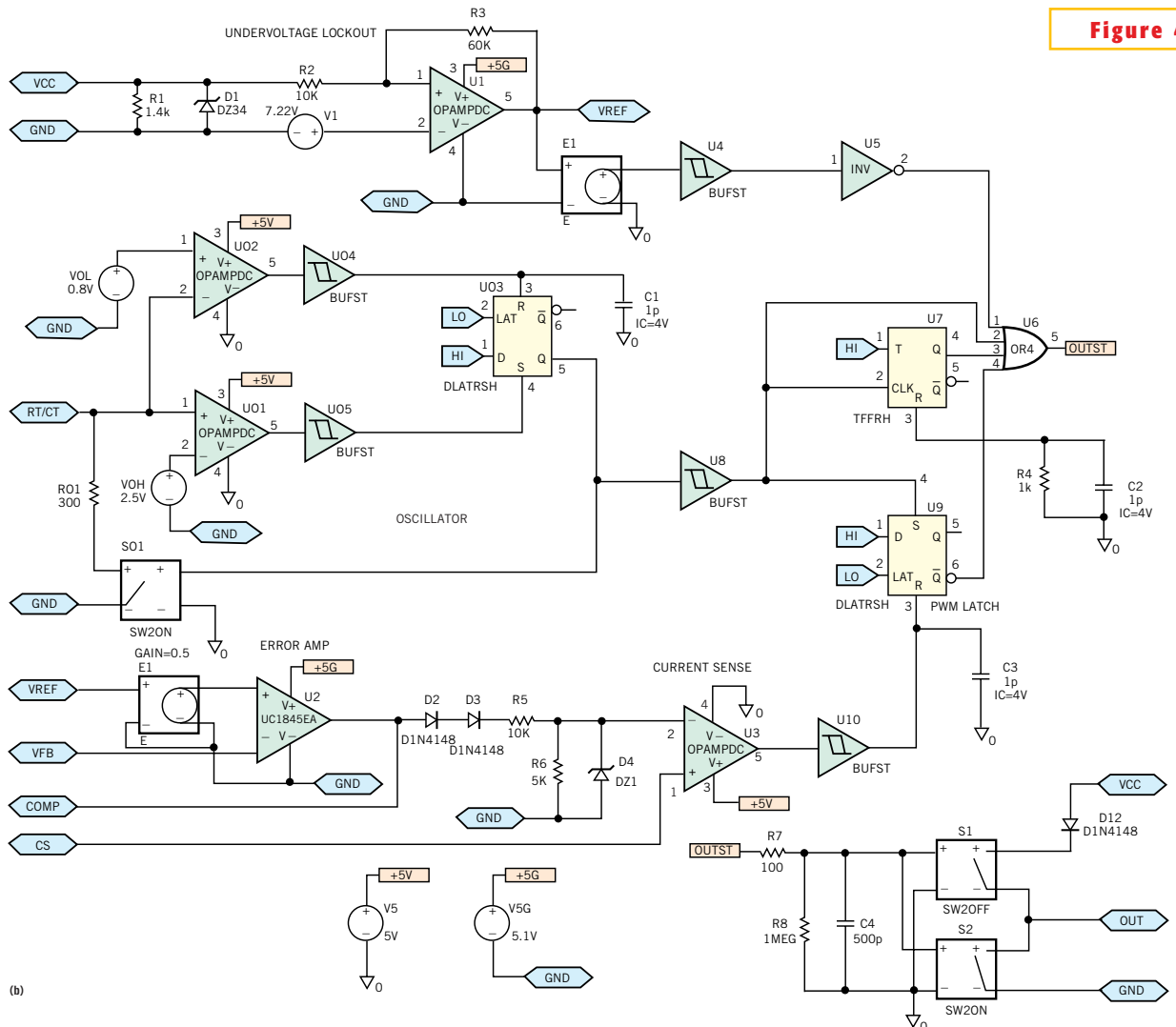
trolled voltage source, E, with a gain of 1 transforms the input reference from ground to Node 0. This method always works if you can't think of anything better.

Another technique that improves the function of macromodels is using Schmitt-trigger buffers between all analog outputs and digital inputs. The hysteresis in the Schmitt trigger prevents analog voltages that transition through the digital dead zone from even temporarily affecting the digital inputs. The BUFST Schmitt-trigger buffer, based on the 7414 gate, performs this function in the 555 macromodel.

The DLATRSH model from the



**Figure 4**

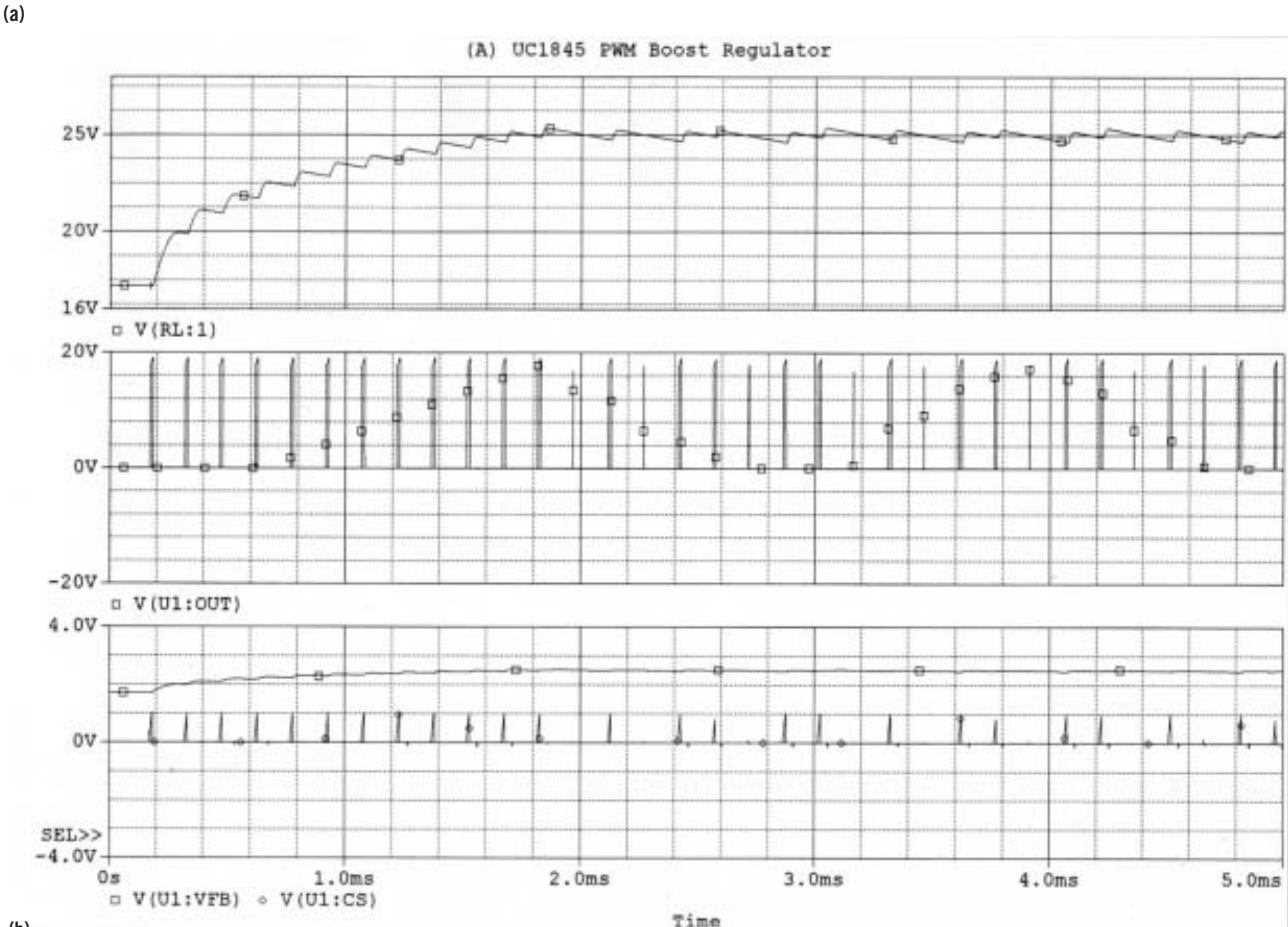
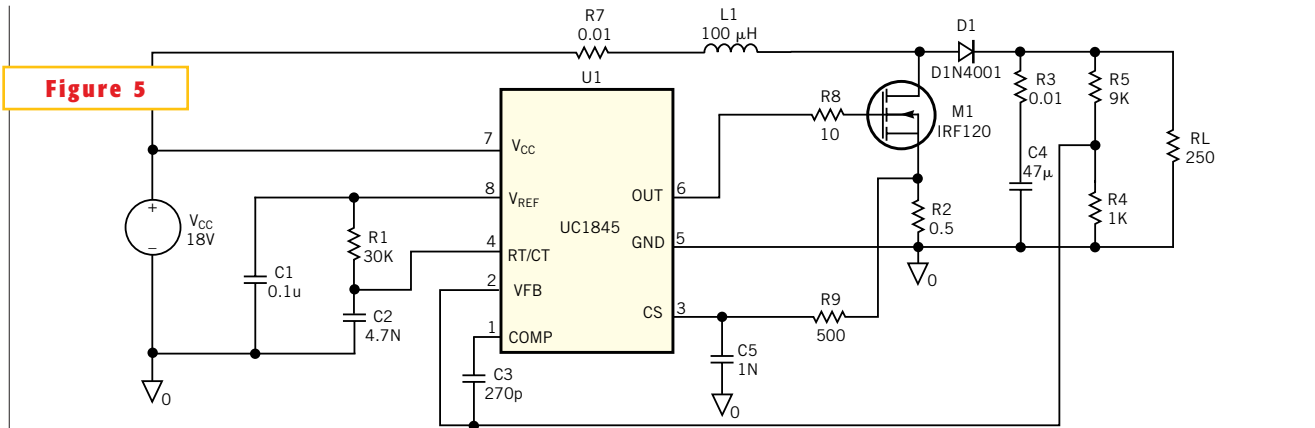


Based on the manufacturer's specifications and block diagram of the UC1845 PWM IC (a), the finished model uses most of the same concepts as in the 555 macromodel (b).

PSpice library functions as the digital asynchronous latch using the Set, S, and Reset, R, inputs, as **Figure 2b** shows. For the DLATRSH Q output, the S input overrides the R input when both are high. Because you always want the OFF state to dominate in the 555 model, the model uses negative logic. In other

words, Q=1 is defined as off. This convention, although confusing, makes the model work correctly for all input combinations. The Trigger input turns on the latch, so the output of U2 connects through BUFST to the R input (negative logic). The latch turns off either when the input reaches the threshold or when

the Reset command is active. The Reset command is active low, so U6 inverts it. U7 ORs these two functions and connects the result to the S input. C1, with an initial condition of 4V, also connects to the S input, requiring the latch to initialize in the Off state for transient simulations. Without C1, the latch will start



(b) In a boost-regulator circuit using the UC1845 macromodel, the UC1845 output-pulse width (middle plot) adjusts (b) according to the voltage- and current-sense feedback (bottom plot).

the simulation in an unknown state. Alternatively, in PSpice, you can change the DIGINITSTATE option from the default value of 2 to either 0 or 1 to set all initial latch values.

The Q output of the latch controls the Output and Discharge outputs of the macromodel. R4 to R7 and C2 to C3 are optional for providing some adjustable delay. SW2ON and SW2OFF are voltage-controlled-switch models set to perform digital-switching functions with the parameters in **Listing 1**. As a result, SW2ON switches on when its input is greater than 2V, and SW2OFF switches off when its input is greater than 2V. For the macromodel's output pin, S1 and S2 connect in a totem-pole configuration between VCC and Ground so that one switch is always on, and the other is always off. This configuration simulates the output driver with reasonable accuracy while making the model, both conceptually and computationally as simple as possible. This configuration also converts from using the internal Node 0 reference to the external Ground reference. S1 and S2 invert the Q output of the latch, converting it back from negative logic. D1 reduces the high output voltage to better match the actual performance of the 555 timer. For the macromodel Discharge pin, S3 functions as an open-collector NPN transistor. S3 switches on when the latch is off. (Q is high.) C4 to C6 are optional for improving transient-simulation convergence.

**Figure 3a** shows one possible test circuit for the 555 macromodel. This example circuit configures the 555 as a voltage-controlled oscillator so that the control-voltage input determines the output frequency. The simulation results in **Figure 3b** show the control-voltage input and the resulting threshold voltage in the top plot and the change in output frequency in the bottom plot.

This model also provided for the successful simulation of several other applications, including some simulations that did not work correctly with the 555 models included with PSpice.

#### PULSE-WIDTH-MODULATOR MACROMODEL

A more complicated example is the UC1845 pulse-width modulator. This macromodel is based on the manufac-

turer's specification and block diagram (**Figure 4a**). The finished UC1845 macromodel in **Figure 4b** uses most of the concepts in the 555 macromodel, including the same type of output stage. The model takes care of the ground-to-Node-0 translations for all logic inputs. Initialization capacitors on the reset inputs of all the latches ensure that simulations begin in a known digital state. This model uses positive logic. The TF-

### THE ACCURACY OF VREF IS MUCH EASIER TO CONTROL IN A SIMULATION THAN IN THE REAL WORLD BY SETTING VRP OF THE OPAMPDC MODEL TO 0.1V AND SETTING THE V5G SOURCE TO 5.1V.

FRH model provides the toggle latch.

The undervoltage-lockout function implements the required hysteresis of the UC1845 using V1, R2, and R3. The value of R3 determines the amount of positive feedback and, along with V1, adjusts the voltage-good threshold to 8.4V when going in the positive direction and 7.6V when going in the negative direction. (For additional details about calculating hysteresis, see the Web version of this article at [www.ednmag.com](http://www.ednmag.com).) VREF is 5V only when the undervoltage-lockout circuit determined the input voltage is to be good. VREF also feeds into the logic as a power-good indication. The accuracy of VREF is much easier to control in a simulation than in the real world by setting VRP of the OPAMPDC model to 0.1V and setting the V5G source to 5.1V. R1 sets the operating supply current according to the specification. The parameters BV=34 and IBV=100u easily model the 34V limiting zener diode D1.

The UC1845 oscillator is based on the oscillator functional description and block diagram borrowed from the UCC1802 specification. The SW2ON model functions as the discharge switch. The oscillator dead-time specs for the UC1845 require adjustment of the dis-

charge resistance, RO1, to approximately 300Ω. The model adjusts VOL, the minimum oscillator voltage reference, so that the oscillator amplitude (VOH-VOL) meets the UC1845 spec of 1.7V. Testing of the oscillator shows that it meets the UC1845 frequency and dead-time requirements.

The error-amp and current-sense macromodels closely follow the block diagram. E2 with gain=0.5 produces the error-amp reference voltage of 2.5V from VREF. The error amp uses the OPAMPAC model with parameters GM=7m and AV=3.3K to match the UC1845 specification plot with a dc gain of 70 dB and unity-gain frequency of 1 MHz. The model of the 1V current-sense reference zener diode, D4, uses the parameters BV=1 and IBV=100u.

**Figure 5a** shows a boost-regulator circuit using the UC1845 macromodel. The regulator pumps charge into capacitor C4 by controlling the turn-on pulses of MOSFET M1. The R4 and R5 voltage dividers set the regulation voltage to 25V as the top plot of **Figure 5b** shows. The UC1845 output-pulse width (middle plot) adjusts according to the voltage and current-sense feedback (bottom plot). □

#### REFERENCES

1. Boyle, GR, BM Cohn, DO Pederson, and JE Solomon, "Macromodeling of Integrated Circuit Operational Amplifiers," *IEEE Journal of Solid-State Circuits*, December 1974.
2. Gustus, Jeff, "Trig function animates Spice comparator," *EDN*, May 27, 1993.
3. Frederiksen, Thomas M, *Intuitive IC Op Amps*, 1984.

#### AUTHOR'S BIOGRAPHY

Ray Kendall is a lead senior electronic engineer at Intuitive Research and Technology Corp (Huntsville, AL), where his duties include electronic simulation, analysis, and modeling for the Research, Development, and Engineering Center of the US Army Aviation and Missile Command. He has worked on the Multiple Launch Rocket System, the Patriot Missile System, and the Space Shuttle Solid Rocket Booster. He holds a BSEE from the University of Mississippi (University) and an MSE from the University of Alabama—Huntsville.