

# Пополнение библиотеки схемных элементов симулятора LTspice. Создание модели ШИМ-контроллера TL494

Валентин ВОЛОДИН

**В настоящее время любителю и профессионалу доступно большое количество программ для персонального компьютера, которые позволяют моделировать работу электронных схем. Зачастую эти программы используют различные базовые математические алгоритмы, что дает им преимущества при моделировании определенного типа электронных устройств.**

## Коротко о SPICE

В настоящее время наибольшую популярность имеют программы, использующие SPICE (Simulation Program with Integrated Circuit Emphasis) алгоритм моделирования процессов, протекающих в электронных схемах. Алгоритм SPICE, разработанный в конце 1970-х годов в университете Беркли (Калифорния), де-факто стал стандартом для разрабатываемых и уже эксплуатируемых в настоящее время коммерческих систем схемотехнического моделирования для персонального компьютера. К таким системам относятся: HSPICE (фирма MetaSoftware), PSpice (Microsim), MicroCap (Spectrum Software), Circuit Maker (The Virtual Electronics Lab), Dr. Spice, ViewSpice (Deutsch Research) и другие. Учитывая тот факт, что не все могут приобрести коммерческую систему схемотехнического моделирования, остановим свой выбор на бесплатном симуляторе LTspice IV (далее просто LTspice) от корпорации Linear Technology ([www.linear.com](http://www.linear.com)). Программа может работать под управлением ОС Windows 98, 2000, NT4.0, Me или XP. Кроме того, LTspice тестировался под Linux RedHat 8.0 с WINE версии 20030219.

LTspice был первоначально основан на симуляторе аналоговых схем SPICE 3F4/5, свободно доступной версии симулятора SPICE. С момента своего создания LTspice подвергался постоянным изменениям с целью его улучшения, устранения ошибок и расширения возможностей. Современная версия LTspice представляет собой превосходный симулятор, позволяющий моделировать аналоговые, цифровые, а также смешанные схемы.

LTspice отличается очень понятным и дружелюбным интерфейсом, позволяющим да-

же новичку очень быстро научиться работать с ним. Новые схемы можно разрабатывать с помощью встроенного редактора схем. Директивы и параметры симулятора помещаются прямо на схеме с использованием стандартного SPICE-синтаксиса. Диаграммы напряжений и токов в узлах и ветвях схемы могут быть получены простым щелчком левой кнопки мышки по соответствующей точке схемы.

LTspice имеет отличный (правда, англоязычный) встроенный помощник Help [1], который является как описанием этой программы, так и руководством для работы с ней. В Интернете существует много ресурсов, где обсуждается и поддерживается эта программа [2].

## Необходимость пополнения библиотеки моделей

Симулятор LTspice изначально предназначался для демонстрационного моделирования импульсных источников питания, использующих управляющие микросхемы производства Linear Technology. При этом LTspice демонстрирует приличную скорость моделирования и оптимальную совместимость с другими стандартами SPICE, что позволяет непосредственно использовать или легко адаптировать к нему оригинальные SPICE-модели различных производителей. Последнее обстоятельство весьма важно, так как на каком-то этапе работы с SPICE-симулятором каждый пользователь сталкивается с необходимостью пополнения библиотеки моделей. К радости пользователей симулятора LTspice можем сообщить, что проблема пополнения встроенной библиотеки новыми моделями уже в значительной степени решена поколениями энтузиастов, работающих с этой программой. Много дополнительных моделей

можно найти в файловом архиве независимой пользовательской группы программы LTspice/SwitcherCAD III [3]. Кроме того, следует посетить небольшую, но весьма информативную домашнюю страничку Андрея Кадатча [4] и скачать дополнительную библиотеку EXTRA, которая выводит объем библиотеки LTspice на уровень, сравнимый с библиотеками коммерческих симуляторов.

К сожалению, даже пополненная библиотека не гарантирует того, что в какой-то момент под рукой окажется необходимая модель. Во многих коммерческих SPICE-симуляторах присутствуют специальные утилиты, позволяющие рассчитывать параметры стандартных моделей по справочным данным реальных компонентов. Разумеется, в бесплатной программе подобная утилита отсутствует, однако это не мешает пользователям LTspice воспользоваться результатами расчета сторонней программы. Практически все производители электронных компонентов выкладывают соответствующие SPICE-модели этих компонентов на своих интернет-сайтах. Обычно эти модели выполнены в виде подсхем. Остается только корректно пополнить этими моделями существующую библиотеку. Если используется стандартная модель, то процесс пополнения простой и выполняется без особых проблем. Если же используется модель в виде подсхемы, то пополнение усложняется и не всегда может закончиться успешно. Рассмотрим оба этих варианта.

## Пополнение библиотеки схемными элементами, имеющими стандартную модель

Параметры схемных элементов, имеющих стандартные модели, расположены в подкаталоге /lib/cmp относительно корневого ка-

талога программы. Здесь можно обнаружить следующие файлы:

- **standard.bjt** — библиотека моделей биполярных транзисторов;
- **standard.cap** — библиотека моделей конденсаторов;
- **standard.dio** — библиотека моделей диодов;
- **standard.ind** — библиотека моделей индуктивностей;
- **standard.jft** — библиотека моделей полевых транзисторов с управляющим *p-n*-переходом (JFET);
- **standard.mos** — библиотека моделей полевых транзисторов с изолированным затвором (MOSFET).

Каждая строка в этих файлах, кроме **standard.cap** и **standard.ind**, представляет собой соответствующую директиву **.model**. Следовательно, пополнение этих библиотек можно осуществить простым вписыванием очередной директивы **.model** с соответствующими параметрами. Например, дополним библиотеку биполярных транзисторов моделью отечественного транзистора типа КТ315А:

```
.model kt315a NPN (Is=2.82f Xti=3 Eg=1.11 Vaf=3.32K Bf=55 Ne=2.35
Ise=554.1p Ikf=18.04m Xtb=1.5
+ Br=.6455 Nc=2 Isc=0 Ikr=0 Rc=0 Cjc=17p Vjc=.75 Mjc=.333 Fc=.5
Cje=8.55p Vje=.75 Mje=.333
+ Tr=3.36u Tf=1.02n Itf=0 Vtf=0 Xtf=0 Vceo=25 Icrating=100m
mfg=USSR)
```

Последние три параметра **Vceo**, **Icrating**, **mfg=USSR** в моделировании не участвуют, но, соответственно, позволяют вывести в строке выбора транзистора его максимальное напряжение коллектор-эмиттер, ток коллектора и наименование производителя.

После дополнения библиотечного файла транзистор КТ315А становится доступным для использования (рис. 1).

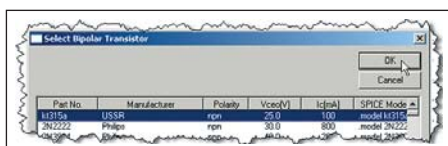


Рис. 1. Выбор биполярного транзистора КТ315А из библиотеки LTspice

## Пополнение библиотеки схемными элементами, имеющими модель в виде подсхемы

Если модель существует в виде подсхемы, то в этом случае придется решить, как минимум, две задачи. Во-первых, надо создать новый или подобрать готовый символ для будущего библиотечного элемента. Во-вторых, надо согласовать SPICE-синтаксис, используемый при описании модели, с синтаксисом LTspice.

Например, пополним библиотеку отечественным операционным усилителем типа К544УД2 (рис. 2).

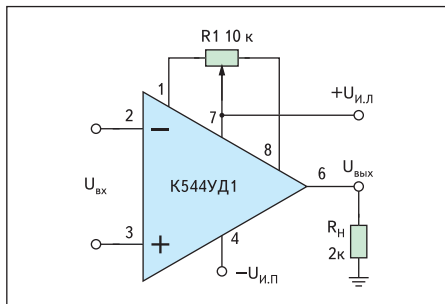


Рис. 2. Схема включения операционного усилителя К544УД2

Приведем текст подсхемы модели этого операционного усилителя:

```
* k544ud2 operational amplifier "macromodel" subcircuit
* created using Parts version 1.04 on 10/11/90 at 10:58
*
* connections:  non-inverting input
*               |
*               | |
*               | | positive power supply
*               | | |
*               | | | negative power supply
*               | | | |
*               | | | | output
*               | | | | |
*               | | | | | compensation
*               | | | | | / \
.subckt k544ud2 1 2 3 4 5 6 7
*
c1 11 12 174.6E-15
dc 5 53 dx
de 54 5 dx
dip 90 91 dx
din 92 90 dx
dp 4 3 dx
egnd 99 0 poly(2) (3,0) (4,0) 0 .5 5
fb 799 poly(5) vb vc ve vlp vln 0 127.3E6 -100E6 100E6 100E6 -100E6
ga 6 0 11 12 1.885E-3
gcm 0 6 10 99 188.5E-9
iss 3 10 dc 400.0E-6
hlim 90 0 vlim 1K
j1 11 2 10 jx
j2 12 1 10 jx
r2 6 9 100.0E3
rd1 4 11 530.5
rd2 4 12 530.5
ro1 8 5 50
ro2 7 99 25
rp 3 4 9.000E3
rss 10 99 500.0E3
vb 9 0 dc 0
vc 3 53 dc 2
ve 54 4 dc 2
vlim 7 8 dc 0
vlp 91 0 dc 20
vln 0 92 dc 20
.model dx D(Is=800.0E-18)
.model jx P(Jf=15.00E-12 Beta=17.77E-3 Vto=-1)
.ends
```

Синтаксис различных SPICE-симуляторов может иметь различия. Поэтому, если подсхема модели заимствована у другого симулятора или взята с сайта производителя электронных компонентов, необходимо внимательно просмотреть ее текст, чтобы выявить и ликвидировать синтаксические несоответствия.

Сравнив нумерацию выводов реального операционного усилителя (рис. 2) и порядок узлов в строке директивы **“.subckt k544ud2 1 2 3 4 5 6 7”**, можно заметить их несоответствие. Соответствие нумерации и порядка узлов подсхемы может потребоваться, если схемный редактор LTspice используется как генератор списка связей для сторонней программы, например, предназначенной для разводки печатной платы. В данном случае такая

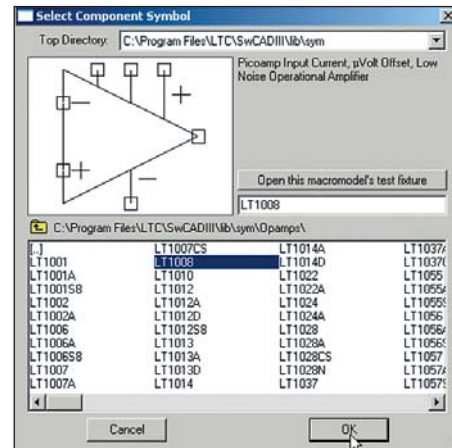


Рис. 3. Поиск подходящего символа во встроенной библиотеке программы LTspice

задача не ставится, и поэтому для нормального функционирования модели достаточно лишь обеспечить верный порядок передачи узловых потенциалов из основной схемы в подсхему, без привязки к физической реализации самой микросхемы.

Так как операционный усилитель является стандартным электронным узлом, то найти для него готовый символ скорее всего не составит труда. И в самом деле, требуемый символ можно позаймствовать у операционного усилителя LT1008, который расположен в подкаталоге **Opamps** встроенной библиотеки (рис. 3).

Чтобы использовать готовый символ, мы должны знать порядок, в котором он передает узлы в подсхему модели. Этот порядок можно узнать, воспользовавшись редактором символов программы LTspice, который будет далее рассмотрен подробно, а пока стараемся решить эту проблему при помощи схемного редактора. Для этого, воспользовавшись командой **Edit=>Component** или нажав клавишу **F2**, загрузим символ требуемого операционного усилителя из раздела **Opamps** встроенной библиотеки и разместим его в рабочем поле схемного редактора. Далее, воспользовавшись командой **Edit=>Draw Wire** или нажав клавишу **F3**, подключаем к выводам операционного усилителя короткие фрагменты внешних цепей и принудительно присваиваем им имена, соответствующие порядку перечисления узлов в строке директивы **.subckt** подсхемы модели операционного усилителя К544УД2 (рис. 4).

Чтобы присвоить имя цепи, нужно вызвать соответствующее меню операций, щелкнув по требуемой цепи правой кнопкой мышки. В появившемся меню выбрать режим **Label Net** («Ярлык для цепи»), что позволит получить доступ к меню **Net Name** («Имя цепи»), где в строке **ABC** присваивается требуемое имя (рис. 4).

Воспользовавшись командой **View=>SPICE Netlist**, посмотрим список связей нашей небольшой схемы (рис. 5) и убедимся, что по-

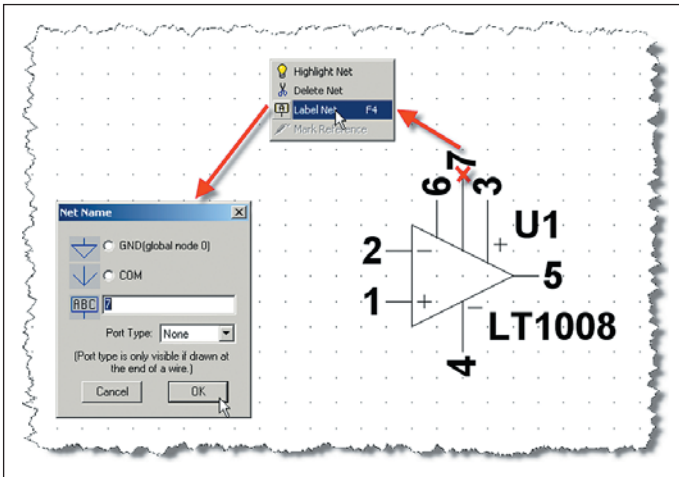


Рис. 4. Присвоение имен внешним цепям символа операционного усилителя LT1008

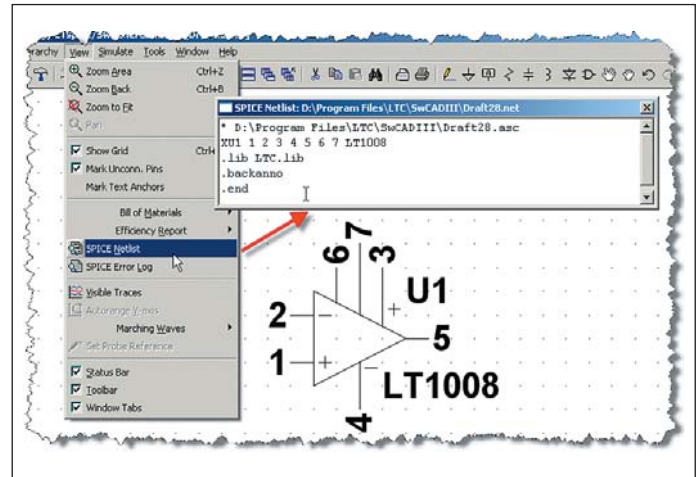


Рис. 5. Проверка порядка узлов, передаваемого символом LT1008 в подсхему модели

рядок узлов в строке “XU1 1 2 3 4 5 6 7 LT1008” соответствует порядку узлов в строке “.subckt k544ud2 1 2 3 4 5 6 7”. В противном случае необходимо соответствующим образом отредактировать строку директивы .subckt.

Программа LTspice регулярно обновляется через Интернет. Поэтому, чтобы не потерять свои библиотеки, желательно выделить их в свою пользовательскую папку. Чтобы папка была видна при выборе нового компонента, ее необходимо разместить в подкаталоге sym встроенной библиотеки. Присвоим этот папке имя Mylib. Создадим в этой папке файл r-opamp.lib и скопируем туда вышеуказанный текст подсхемы модели. Впоследствии этот файл можно пополнять другими моделями операционных усилителей отечественного производства. В начале файла, в виде комментариев, необходимо сделать небольшую информационную вставку, описывающую содержимое библиотеки. Это облегчит работу с библиотекой, а также позволит избежать каких-то досадных ошибок.

Пример информационной вставки:

```

*****
* Библиотека моделей операционных усилителей
* отечественного производства
*
*****
* Библиотека содержит:
* K544UD2
*
*****
* Внимание! Нумерация (порядок) входных узлов подсхем
* моделей всех ОУ не совпадает с реальной нумерацией выводов
* корпуса микросхем.
*****
    
```

**Примечание.** Для схемных элементов различных типов лучше создавать независимые библиотечные файлы. Например, для ШИМ-контроллеров можно создать файл pwm.lib, для полевых транзисторов с изолированным затвором — mosfet.lib и т. д.

Скопируем файл символа LT1008.asy из своего каталога ..\lib\sym\Opamps в пользовательский каталог ..\lib\sym\Mylib и переименуем его в k544ud2.asy. Однако одного пе-

реименования недостаточно. Необходимо также внести некоторые изменения и в сам файл. Для этого откроем его при помощи любого текстового редактора и изменим содержимое атрибутов SYMATR:

```

SYMATTR Value — вместо LT1008 записываем k544ud2
SYMATTR Prefix X — оставляем без изменения
SYMATTR SpiceModel — вместо LTC.lib записываем r-opamp.lib
SYMATTR Value2 — вместо LT1008 записываем k544ud2
SYMATTR Description — запишем Operational Amplifier
    
```

После этого сохраним изменения и приступим к испытанию. Для этого запустим программу LTspice в режиме схемного редактора и проверим наличие нового электронного компонента в библиотеке. Если все сделано правильно, то операционный усилитель K544UD2 можно обнаружить именно там, где он и должен быть, то есть в пользовательском каталоге Mylib (рис. 6).

Для проверки функционирования схемного элемента можно собрать какую-то тестовую схему. Пусть такой схемой будет обычный инвертирующий усилитель (рис. 7).

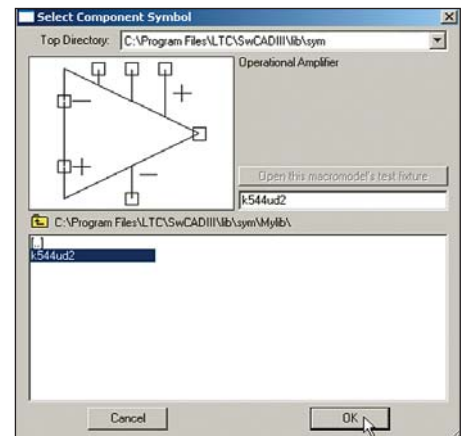


Рис. 6. Выбор операционного усилителя K544UD2 из библиотеки пользователя

Согласно результатам моделирования, модель операционного усилителя K544UD2 функциональна, и инвертирующий усилитель, собранный на нем, обеспечивает заданное усиление  $K_u = R_3/R_2 = 100к/10к = 10$ .

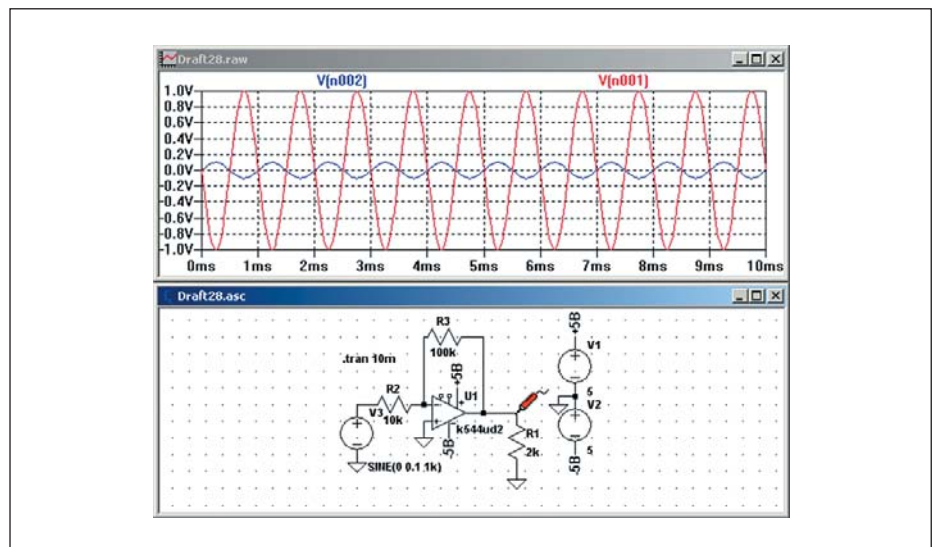


Рис. 7. Тестовая схема инвертирующего усилителя на ОУ K544UD2

### Создание модели схемного элемента с использованием иерархической схемы

Ранее мы рассмотрели случаи создания схемного элемента, когда его модель уже существует в том или ином виде. Обычно так и происходит, но иногда бывают ситуации, когда модели нет, и ее невозможно найти ни в одном из доступных информационных источников. В этом случае приходится создавать схемный элемент с нуля. Обычно создание нового элемента начинается с создания его модели.

В качестве примера, используя средства программы LTspice, создадим модель микросхемы TL494. Микросхема TL494 является ШИМ-контроллером импульсного источника питания и включает в себя все необходимые для этого блоки. Эта микросхема, благодаря своим достаточно высоким характеристикам и низкой стоимости, очень популярна как среди любителей, так и среди профессионалов в силовой электронике. Так же как и широко известный 555-й таймер, TL494 выпускают многие мировые производители электронных компонентов, такие как Texas Instruments, ON Semiconductor, Motorola, Fairchild Semiconductor, Fujitsu, Samsung и т. д. Эта микросхема выпускается под своим оригинальным, а также под альтернативными названиями, такими как mPC494C, MB3759, KA7500B, IR3M02 и т. п. Отечественной версией TL494 является микросхема 1114EУ4.

Однако, несмотря на популярность, модель TL494 пока отсутствует в библиотеках коммерческих симуляторов и на сайтах производителей электронных компонентов.

#### Иерархические схемы

В программе LTspice предусмотрена очень полезная возможность замены отдельных фрагментов схемы схемами с более низким уровнем иерархии. Причем, в основной схеме, имеющей высокий уровень, эти иерархические схемы выглядят как абстрактные символы с выводами. В этом смысле иерархические схемы, по сути, очень похожи на подсхемы, но гораздо удобнее и нагляднее, так как пользователю приходится работать не с текстовым файлом, а с реальной схемой. Ее легко изменить и контролировать обычными средствами схемного редактора программы LTspice.

Так же как и подсхемы, иерархические схемы могут быть помещены в библиотеку для последующего использования в различных проектах. Но, в отличие от подсхем, ссылаться на иерархическую схему из основного проекта можно только с помощью символа. Файл символа должен иметь такое же имя, что и файл иерархической схемы. Например, если файл иерархической схемы имеет название tl494h.asc, то файл символа должен иметь название tl494h.asy. По этой же причине имя символа не должно содержать пробелы, а также символы, недопустимые для имени файла.

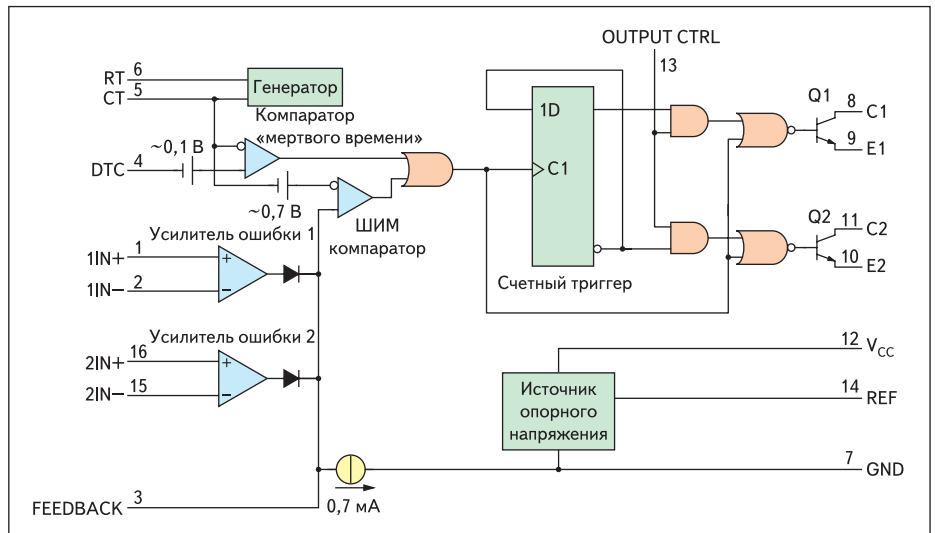


Рис. 8. Функциональная схема TL494

**Примечание.** Иерархическая схема должна располагаться в одном каталоге с основной схемой.

Имена выводов символов должны соответствовать именам соответствующих цепей иерархической схемы. Таким образом, создается связь между основной и иерархическими схемами.

Любой файл, открытый с помощью команды File=>Open, рассматривается как схема высокого уровня.

#### Создание модели TL494 в виде иерархической схемы

Справочную информацию о микросхеме TL494 можно взять, например, на сайте компаний Texas Instruments [5] или ON Semiconductor [6].

На рис. 8 изображена ее функциональная схема.

Казалось бы, что для создания корректной модели достаточно найти и продублировать принципиальную электрическую схему TL494. Благо, упрощенные версии отдельных узлов даже приводятся в документации, расположенной на сайте Texas Instruments [7]. Однако попытка создания модели таким способом скорее всего закончится неудачей. Дело в том, что, несмотря на свои маленькие размеры, микросхема имеет достаточно сложную электрическую принципиальную схему, содержащую более 50 транзисторов. Модель, содержащая такое количество активных элементов, способна в значительной мере снизить скорость симуляции или сделать ее вообще невозможной из-за проблем сходимости. Выход из данной ситуации — использование встроенных схемных элементов LTspice, которые являются функциональными аналогами отдельных узлов моделируемой микросхемы.

#### Генератор

ШИМ-контроллер TL494 работает на фиксированной частоте, которая определяется

внутренним генератором. Частота генератора программируется внешними резистором и конденсатором, подключаемыми к выводам R<sub>t</sub> и C<sub>t</sub> микросхемы. Конденсатор C<sub>t</sub> заряжается стабильным током, величина которого задается с помощью резистора R<sub>t</sub>. После того, как напряжение на конденсаторе достигает 3 В, он быстро разряжается до нуля, и процесс снова повторяется. Величину зарядного тока можно определить по формуле I<sub>зар</sub> ≈ 3,65/R<sub>t</sub>. На рис. 9 приведены графические зависимости частоты внутреннего генератора от номиналов частотозадающих элементов. Эту частоту также можно определить по формуле f<sub>ген</sub> = 1,2/(R<sub>t</sub>·C<sub>t</sub>).

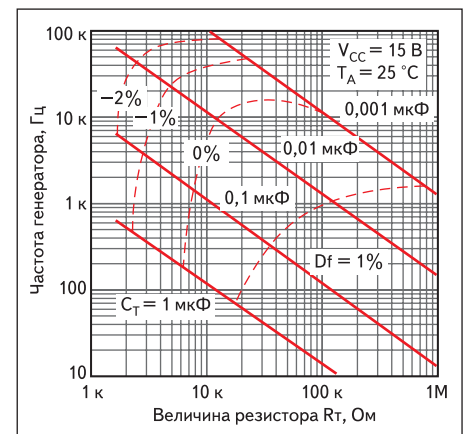


Рис. 9. Частота генератора и ее температурный дрейф в зависимости от величин R<sub>t</sub> и C<sub>t</sub>

Схема модели генератора изображена на рис. 10.

Источник напряжения V1 имитирует напряжение, приложенное к резистору R<sub>t</sub>. ИТУТ (источника тока, управляемый ток) F1 является своеобразным токовым зеркалом, которое формирует ток зарядки конденсатора C<sub>t</sub>, равный току, протекающему через источник напряжения V1 и резистор R<sub>t</sub>. Под воздействи-

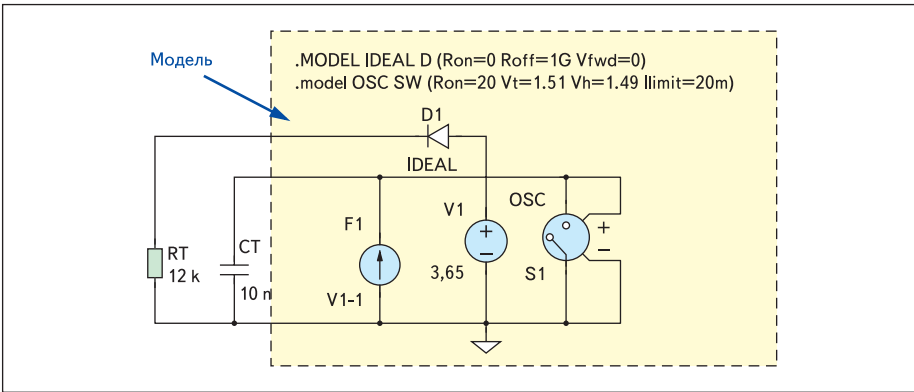


Рис. 10. Схема модели задающего генератора

ем этого тока конденсатор  $C_T$  линейно заряжается. Как только напряжение на конденсаторе достигает порогового напряжения 3 В, он разряжается с помощью ключа  $S_1$ . Диод  $D_1$  предотвращает обратный ток и позволяет напряжению на выводе  $R_T$  подниматься выше потенциала источника напряжения  $V_1$ .

При указанных на схеме номиналах резистора  $R_T = 12$  кОм и конденсатора  $C_T = 10$  мкФ рабочая частота генератора составляет 10 ГГц, что соответствует графической зависимости на рис. 9.

**Усилитель ошибки**

Регулировочные характеристики ШИМ-контроллера определяются амплитудно-частотной характеристикой (АЧХ) его усилителя ошибки. Поэтому при создании модели усилителя нужно обеспечить корректность этой характеристики. Согласно документации Texas Instruments и других производителей, АЧХ усилителя ошибки имеет частоту среза 10 Гц с последующим завалом 20 дБ/дек (рис. 11).

Подобную АЧХ можно легко получить с помощью RC-фильтра нижних частот. Схема модели усилителя ошибки с RC-фильтром нижних частот, а также результаты моделирования изображены на рис. 12.

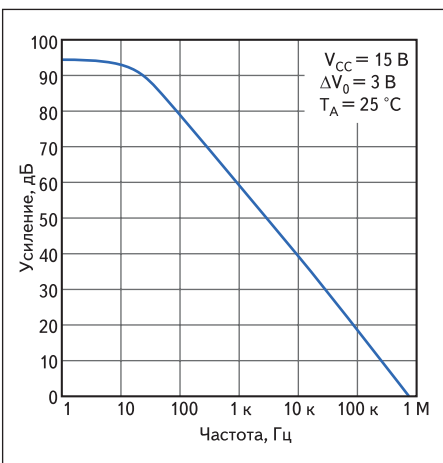


Рис. 11. Амплитудно-частотная характеристика усилителя ошибки

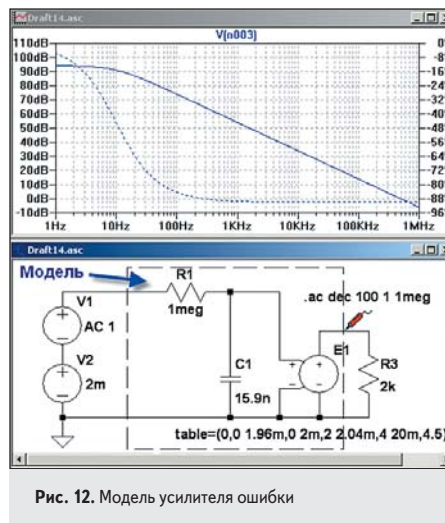


Рис. 12. Модель усилителя ошибки

Фильтр нижних частот  $R_1, C_1$  позволяет получить АЧХ, которая идеально совпадает с изображенной на рис. 11.

Нелинейность передаточной характеристики усилителя ошибки, а также коэффициенты усиления на низких частотах имитируются

с помощью ИТУН (источник напряжения, управляемый напряжением)  $E_1$ . Согласно общим справочным данным на ШИМ-контроллер TL494 (документация t1494.pdf и TL494-D.PDF), коэффициент передачи усилителя ошибки  $AVOL = 50\,000$  (95 дБ). Коэффициент усиления и передаточную характеристику усилителя ошибки представим в виде таблицы “table=(0,0 1.96m,0 2m,2 2.04m,4 20m,4.5)”.

Источник напряжения  $V_2$  определяет рабочую точку усилителя ошибки, а резистор  $R_3$  играет роль его нагрузки во время АС-анализа.

**Выходной каскад**

Выходной каскад контроллера TL494 представляют собой транзистор с открытыми эмиттером и коллектором. Благодаря этому, возможны два варианта включения этого транзистора, где он включен по схеме с общим эмиттером (рис. 13а) или общим коллектором (рис. 13б). Каждому из этих включений соответствует своя зависимость напряжения открытого транзистора  $V_{кэ,нас}$  от тока нагрузки.

Желательно, чтобы модель корректно имитировала эти характеристики насыщения, а также ситуации превышения максимального тока и пробоя повышенным напряжением транзистора выходного каскада.

Сначала рассмотрим модель выходного каскада, включенного по схеме с общим эмиттером, изображенную на рис. 14.

С помощью ИТУН  $G_1$  и стабилитрона  $D_1$  имитируется открытый транзистор выходного каскада. Этот эквивалентный транзистор полностью открыт, если ток нагрузки  $I_L$  не превышает максимально допустимого значения в 250 мА (для микросхемы TL494 производства Texas Instruments). При превышении максимального тока эквивалентный транзистор насыщается, и напряжение на его

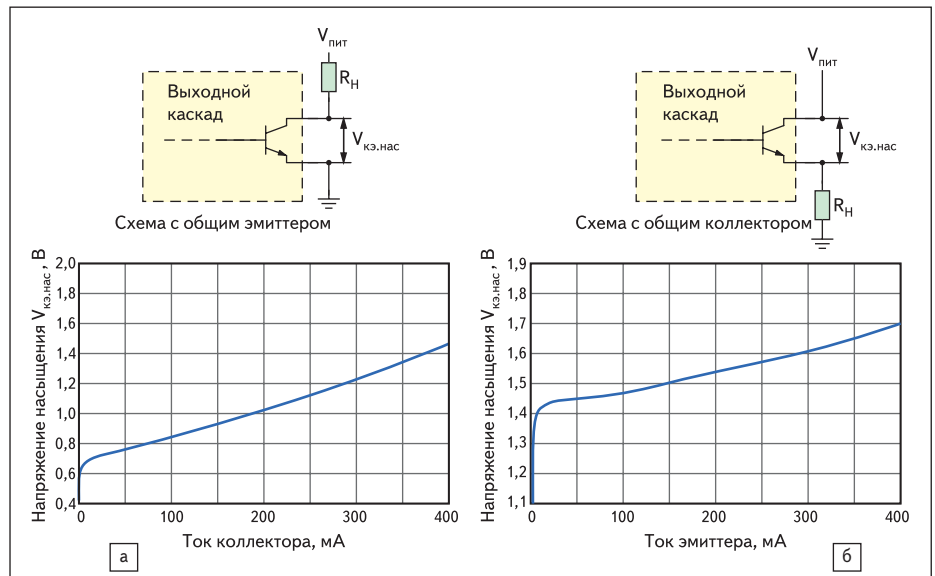


Рис. 13. Характеристики напряжения насыщения транзистора выходной схемы для включения: а) с общим эмиттером; б) с общим коллектором

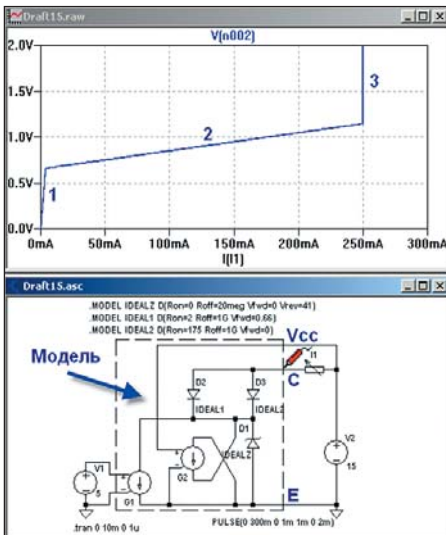


Рис. 14. Модель выходного каскада, включенного по схеме с общим эмиттером

выводах резко возрастает (участок 3 ВАХ). В области малых токов (участок 1) ВАХ эквивалентного транзистора определяется внутренним сопротивлением  $R_{on}$  диода D3, а в области средних токов (участок 2) — внутренним сопротивлением параллельно включенных диодов D2 и D3. Пробой эквивалентного транзистора при повышенном напряжении имитируется стабилитроном D1, имеющим напряжение стабилизации  $V_{rev} = 41$  В. ИТУН G2 в данном режиме не активен, так как напряжение управления всегда равно напряжению питания микросхемы.

Теперь рассмотрим модель выходного каскада, включенного по схеме с общим коллектором, изображенную на рис. 15.

В схеме с общим коллектором ИТУН G2 контролирует напряжение на открытом транзисторе выходного каскада. Коэффициент передачи G2 описывается таблицей

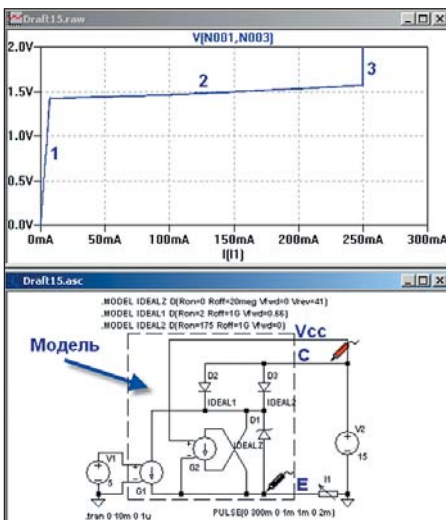


Рис. 15. Модель выходного каскада, включенного по схеме с общим коллектором

“table=(0 250m,1.42 243m,1.46 150m,1.57 0)”. Согласно этой таблице, если входное напряжение ИТУН G2 падает ниже 1,57 В, последний начинает шунтировать выход ИТУН G1. Следствием этого шунтирования будет сдвиг ВАХ открытого транзистора в сторону больших напряжений.

### Потребляемый ток и источник опорного напряжения

Ток  $I_{ss}$ , потребляемый микросхемой TL494, имеет нелинейную зависимость от питающего напряжения  $V_{cc}$ . Эта зависимость изображена на рис. 16.

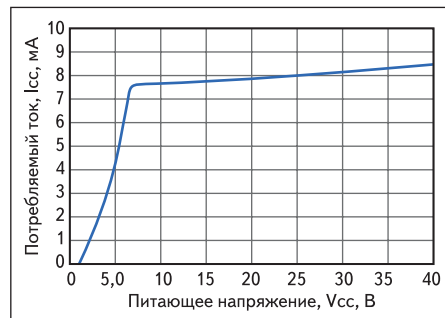


Рис. 16. Зависимость тока, потребляемого микросхемой TL494, от питающего напряжения

На рис. 17 изображена модель, которая с помощью ИТУН G1 имитирует потребляемый ток микросхемы. Коэффициент передачи ИТУН задан таблицей “table=(1 0,5 4.5m, 6.85 7.6m,40 8.4m)”. Источник опорного напряжения имитируется при помощи источника тока I1 и стабилитрона D1. Ток нагрузки источника опорного напряжения, при помощи поведенческого источника тока B1, суммируется к току, потребляемому микросхемой. В соответствии со справочными данными, ток короткого замыкания источника опорного напряжения составляет 25 мА.

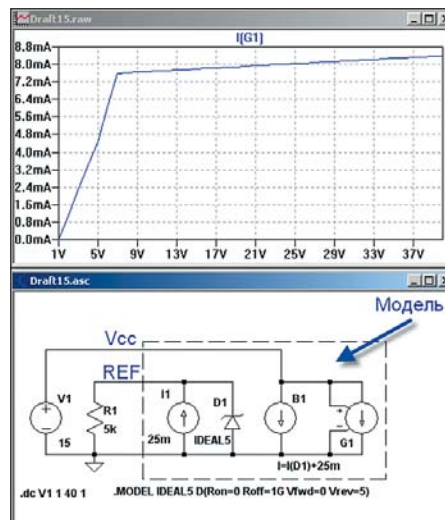


Рис. 17. Модель источника опорного напряжения и имитатора потребляемого тока

### Общая схема модели микросхемы TL494

После того, как созданы основные узлы модели, создание общей схемы модели (рис. 18) не вызывает особых затруднений.

Всем внешним цепям иерархической схемы необходимо присвоить имена. Удобнее всего использовать имена соответствующих выводов микросхемы прототипа. Чтобы присвоить имя цепи, нужно вызвать соответствующее меню операций, щелкнув по требуемой цепи правой кнопкой мышки. В появившемся меню выбрать режим Label Net («Ярлык для цепи»), что позволит получить доступ к меню Net Name («Имя цепи»), где в строке ABC указывается требуемое имя (рис. 19). Для большей наглядности можно определить направления сигнала во внешних цепях с помощью указателя Port type:

- None — без направления;
- Input — входная цепь;
- Output — выходная цепь;
- Bi-Direct — двунаправленная цепь.

Указатель Port type на процессы моделирования не влияет.

Общий провод внутри иерархической схемы лучше не подключать к «земле», а соединить явно или подключить к цепи «COM». Это позволит в дальнейшем использовать модель в схемах с плавающим, относительно «земли», питанием.

При помощи команды File=>Save As... назовем как TL494N.asc и сохраним в пользовательский каталог ..\lib\sym\Mylib полученную иерархическую схему модели TL494.

### Редактор символов

Каждый схемный элемент имеет собственный символ, позволяющий легко идентифицировать его на электрической принципиальной схеме. Внешний вид символов схемных элементов, таких как резисторы, конденсаторы, индуктивности, диоды, транзисторы и т. п., оговаривается соответствующим стандартом на конструкторскую документацию. В настоящее время в мире существует два основных стандарта, один из которых американский, а другой — европейский. К последнему весьма близка отечественная «Единая система конструкторской документации» (ЕСКД). В библиотеках LTspice использован американский стандарт на символы схемных элементов. С этим можно смириться, так как обычно данный симулятор не используется для подготовки конструкторской документации, но при необходимости символы схемных элементов можно привести к более привычному виду.

### Окно редактора символов

В LTspice имеется специальный редактор символов, с помощью которого можно изменять существующие или создавать новые символы схемных элементов. Для запуска этого редактора необходимо активизировать команду File=>New Symbol. В результате появится окно редактора символов (рис. 20).

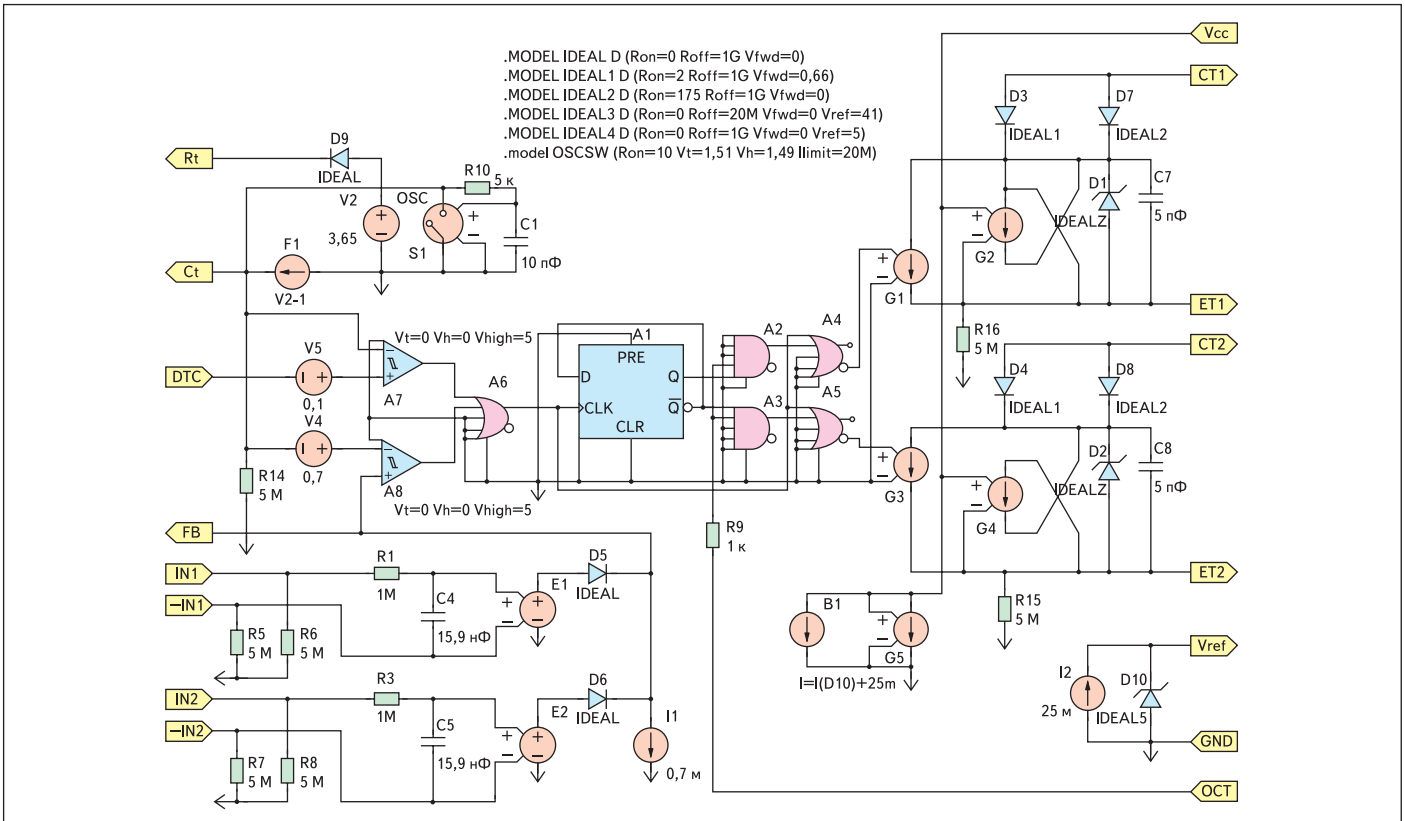


Рис. 18. Иерархическая схема модели микросхемы TL494

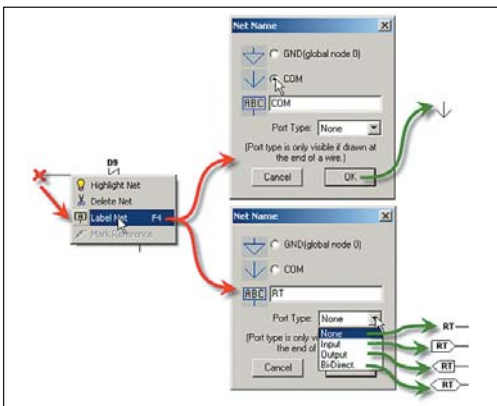


Рис. 19. Указание наименования и направления цепей

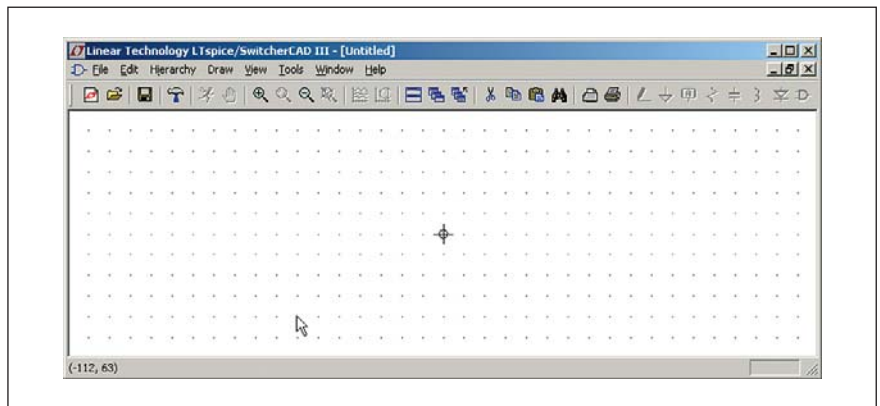


Рис. 20. Окно редактора символов

Отличительный признак этого окна — крестик с кружком в центре рабочего поля, а также синий символ логического вентиля в левой части панели команд.

**Панель команд редактора символов**

Панель меню команд очень похожа на аналогичную панель схемного редактора. В отличие от панели команд схемного редактора здесь присутствует меню Draw, но зато отсутствует меню Simulate.

Рассмотрим подробнее подпункты меню команд:

- File — подпункты этого меню аналогичны подпунктам меню File схемного редактора.
- Edit — служит для редактирования символа и содержит следующие подпункты:

- Undo (F9) — отменить предыдущее действие.
- Redo (Shift+F9) — вернуть отмененное действие.
- Attributes — атрибуты символа:
  - Edit Attributes (Ctrl+A) — редактировать отдельные атрибуты символа;
  - Attribute Window (Ctrl+W) — делать видимыми отдельные атрибуты символа.
- Add Pin/Port ('P') — добавить электрическое подключение (вывод) символу.
- Move (F7) — переместить выбранные элементы символа. При этом существующие связи разрываются.
- Drag (F8) — перетащить выбранные элементы символа. При этом существующие связи сохраняются.

- Rotate (Ctrl+R) — вращать выбранные элементы символа.
- Mirror (Ctrl+E) — отразить выбранные элементы символа.
- Delete (F5) — удалить выбранные элементы символа.
- Duplicate (F6) — копировать (дублировать) выбранные элементы символа.
- Hierarchy — посмотреть внутреннюю структуру иерархического символа.
- Draw — создавать изображения символа при помощи графических примитивов и текста:
  - Line ('L') — выбрать рисование прямой линии;
  - Rect ('R') — выбрать рисование прямоугольника;
  - Circle ('C') — выбрать рисование круга;

- Arc ('A') — выбрать рисование дуги;
- Line Style — определить стиль линии;
- Text ('T') — выбрать ввод текста.
- View — управление режимом просмотра изображения символа:
  - Zoom In (Ctrl+Z) — увеличить изображение символа;
  - Zoom to Fit — установить масштаб изображения, позволяющий максимально использовать текущий размер рабочей области редактора символов;
  - Zoom Out (Ctrl+B) — уменьшить изображение символа;
  - Anchor Points ('O') — включить/выключить видимость точек привязки графических элементов или текста;
  - Pin Table — вывести таблицу, связывающую имена выводов символа с их порядковыми номерами;
  - Status Bar — вывести/убрать строку подсказки редактора символа;
  - Toolbar — вывести/убрать панель инструментов редактора символа;
  - Windows Tabs — вывести/убрать панель вложенных окон.
- Tools — подпункты этого меню аналогичны подпунктам меню Tools схемного редактора.
- Windows — подпункты этого меню аналогичны подпунктам меню Windows схемного редактора.
- Help — подпункты этого меню аналогичны подпунктам меню Help схемного редактора.

### Панель инструментов редактора символов

Панель инструментов редактора символов полностью аналогична панели инструментов редактора схем.

### Создание изображения символа

В редакторе символов для создания изображения символа используются различные графические примитивы, такие как линия, прямоугольник, круг и дуга. Кроме этого, в изображение символа можно вносить текст, который используется для наименования выводов, а также для указания различных атрибутов символа.

Все графические и текстовые элементы символа имеют специальные точки привязки, которые отображаются в виде маленьких красных кружков, если, с помощью команды View=>Mark Object Anchors, разрешена видимость точек привязки. Именно за эти точки элементы перемещаются или перетаскиваются при помощи команд Move (переместить) и Drag (перетащить). Для улучшения ориентации в рабочем поле редактора символов нанесена сетка. Шаг сетки равен 16 внутренним единицам программы, которые не имеют определенной привязки к метрическим или дюймовым мерам длины. Например, если судить по резисторам, то этот шаг составляет примерно 2,5 мм, а если по микросхемам, то примерно 1,25 мм. Особо не вникая в эту пробле-

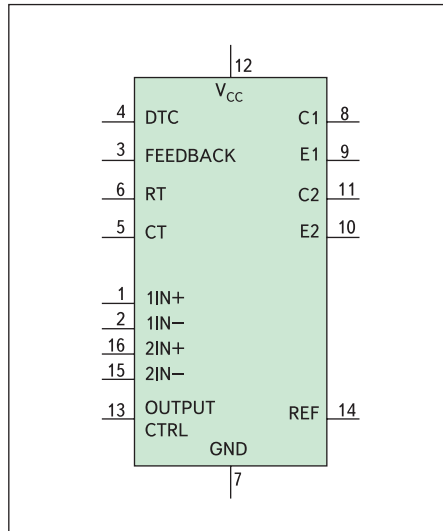


Рис. 21. Схематическое обозначение ШИМ-контроллера TL494

му, будем создавать новые символы, ориентируясь на те подходы, которые уже используются в программе LTspice.

В качестве примера создадим символ ШИМ-контроллера TL494 (рис. 21), для которого ранее уже была создана модель в виде иерархической схемы.

Согласно неписанному соглашению, принятому для символов микросхем из библиотеки программы LTspice, расстояние между отдельными выводами символа составляет 4 шага сетки. Поэтому, выбрав команду Draw=>Rect, нарисуем прямоугольник шириной 16 и высотой 32 клетки. Так как курсор не привязан к узлам сетки, при этом придется хорошенько прицеливаться. Результат данных действий изображен на рис. 22.

### Добавление выводов

Далее надо указать точки, в которых символ будет подключаться к схеме. Эти точки создаются при помощи команды Edit=>Add

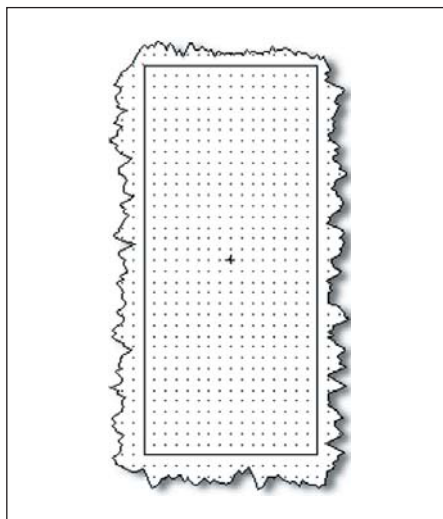


Рис. 22. Заготовка символа ШИМ-контроллера TL494

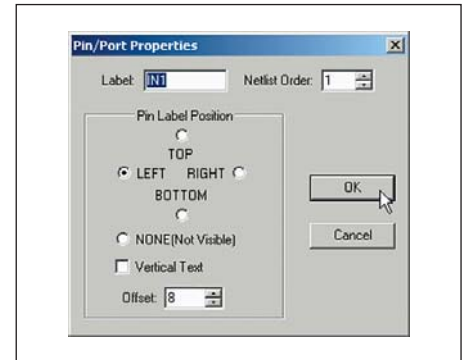


Рис. 23. Окно Pin/Port Properties, определяющее свойства вывода

Pin/Port. Результатом будет появление окна Pin/Port Properties (рис. 23). В области Pin Label Position определяется видимость (NONE) и расположение названия вывода. Вывод может быть расположен левее (LEFT), правее (RIGHT), выше (TOP) или ниже (BOTTOM) своего названия. С помощью поля настройки Offset можно определить смещение названия относительно вывода во внутренних единицах программы. По умолчанию это смещение составляет 8 единиц, но может быть установлено любое значение в диапазоне от 0 до 50.

Если символ представляет элемент, являющийся нижним уровнем иерархической схемы, то название вывода должно соответствовать названию цепи в схеме нижнего иерархического уровня. Если символ представляет стандартный SPICE-примитив, то название вывода не имеет особого значения, но порядковый номер в поле Netlist Order должен соответствовать номеру вывода соответствующего схемного элемента. Это же справедливо и для символа, представляющего библиотечный элемент в виде подсхемы. Но в этом случае номер в поле Netlist Order должен соответствовать порядковому номеру соответствующего узла в строке директивы .SUBCKT.

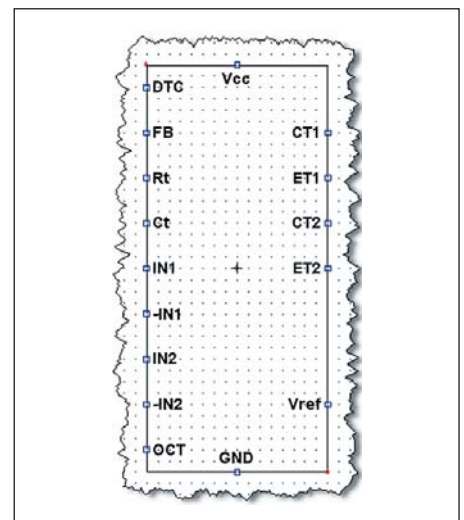


Рис. 24. Вид символа микросхемы TL494 после добавления всех выводов



Если создаваемый с нуля схемный элемент не является стандартным SPICE-примитивом, то желательно, чтобы нумерация выводов его символа соответствовала нумерации выводов оригинального электронного элемента, которую указывает производитель в виде справочных данных.

Ошибки, допущенные при добавлении выводов, можно легко исправить, если повторно вызвать окно Pin/Port Properties, щелкнув правой кнопкой мышки по соответствующему выводу символа.

На рис. 24 изображен результат, который должен быть получен после добавления всех выводов символа микросхемы TL494.

**Добавление атрибутов**

Чтобы получить доступ к атрибутам символа, необходимо с помощью команды Edit=>Attributes=>Edit Attributes вызвать окно редактирования атрибутов Symbol Attribute Editor (рис. 25). В строке Symbol Type необходимо выбрать тип символа:

- Cell — тип соответствует символам элементов, имеющих встроенную SPICE-модель или модель в виде подсхемы. Для этих символов надо указывать атрибут Prefix.
- Block — тип соответствует символам элементов, имеющих модель в виде иерархической схемы. Для этих символов не надо указывать атрибут Prefix.

Далее, в виде таблички, перечисляются все атрибуты символа:

- Prefix — этот атрибут является наиболее важным. Он определяет тип схемного элемента. Например, резистор имеет префикс R, конденсатор — C, MOSFET-транзистор — M и т. д. Если модель схемного элемента представлена подсхемой, то используется префикс X. Возможна ситуация, когда модель стандартного схемного элемента, например, MOSFET-транзистора, представлена подсхемой. В этом случае

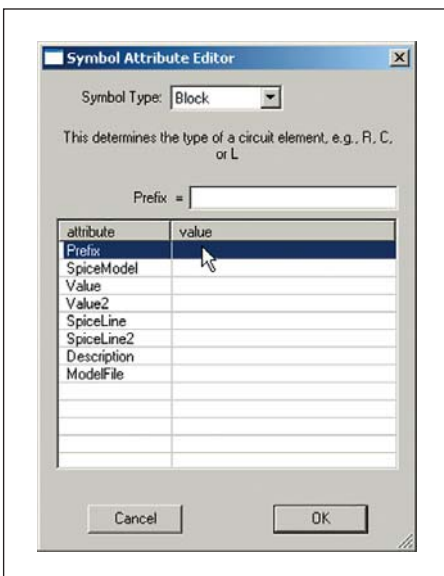


Рис. 25. Окно редактирования атрибутов символа

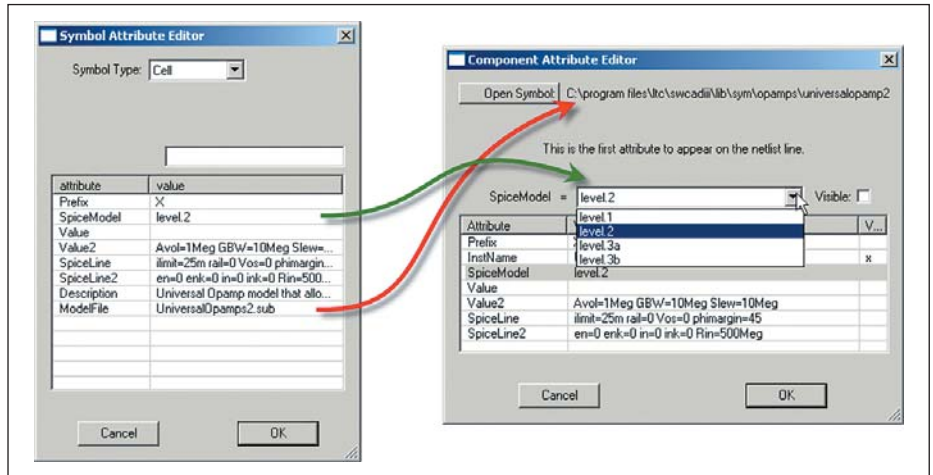


Рис. 26. Пример использования атрибута ModelFile

символу схемного элемента также присваивается префикс X.

- SpiceModel — полное или относительное имя библиотечного файла, включающего модель схемного элемента.
- Value — атрибут отображается на схеме и обозначает номинал или тип схемного элемента.
- Value2 — атрибут выводится в список соединений и должен совпадать с именем подсхемы в библиотечном файле. Если этот атрибут указан, редактирование элемента на схеме становится недоступным.
- SpiceLine — атрибут позволяет передавать параметры в подсхему.
- SpiceLine2 — атрибут позволяет передавать параметры в подсхему.
- Description — атрибут, позволяющий вставить краткое описание схемного элемента.
- ModelFile — атрибут позволяет определить файл, который будет включен в список соединений как библиотечный. Использование этого атрибута меняет назначение атрибута SpiceModel, который теперь указывает название конкретной подсхемы в библиотечном файле. В свою очередь, в окне редактирования компонента появляется строка выбора, позволяющая выбирать отдельные подсхемы библиотечного файла. Хорошим примером использования этого атрибута является пара символа “.lib/sym/Opamps/UniversalOpamp2.asy” и подсхемы “.lib/sub/UniversalOpamps2.sub” (рис. 26).

Атрибуты SpiceModel, Value, Value2, SpiceLine и SpiceLine2 формируют строку текстового описания компонента в формате SPICE списка соединений. В результате генерируется SPICE-строка, которая выглядит следующим образом:

```
<Имя> узел1 узел2 [...] <SpiceModel> <Value> <Value2>
<SpiceLine> <SpiceLine2>
```

Компонент, имеющий префикс X и определенные атрибуты Value и Value2, в списке связей представлен двумя SPICE-строками:

```
<name> node1 node2 [...] <Value2>
.lib <SpiceModel>
```

Как говорилось ранее, редактирование такого компонента недоступно из редактора схем.

Также двумя SPICE-строками в списке соединений представлен компонент, имеющий префикс X и определенный атрибут ModelFile:

```
<name> node1 node2 [...] <SpiceModel> <Value> <Value2> <SpiceLine>
<SpiceLine2>
.lib <ModelFile>
```

Редактирование такого компонента доступно из редактора схем.

На рис. 27 показано окно редактирования атрибутов Symbol Attribute Editor, заполненное для символа схемного элемента TL494, модель которого выполнена в виде иерархической схемы.

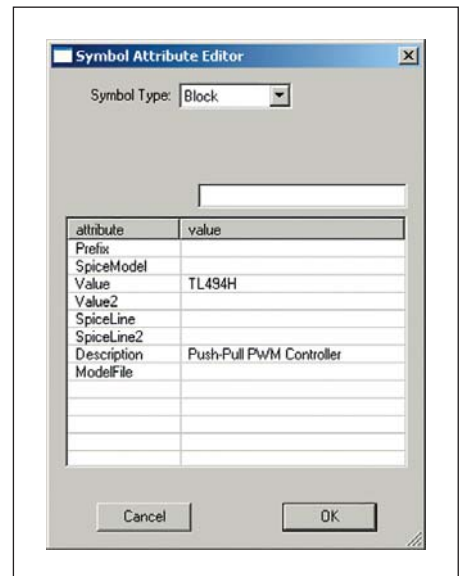


Рис. 27. Окно редактирования атрибутов, заполненное для символа TL494

**Видимость атрибутов**

Пользователь может сам выбирать те атрибуты, которые должны быть видимыми на изображении символа. Видимость атрибутов устанавливается при помощи окна Attribute Windows to Add, которое вызывается командой Edit=>Attributes=>Attribute Window (рис. 28).

Выбрав требуемый атрибут из списка, необходимо нажать кнопку ОК. После этого атрибут можно разместить в рабочем поле редактора символов. После размещения атрибута можно редактировать его текст, изменять

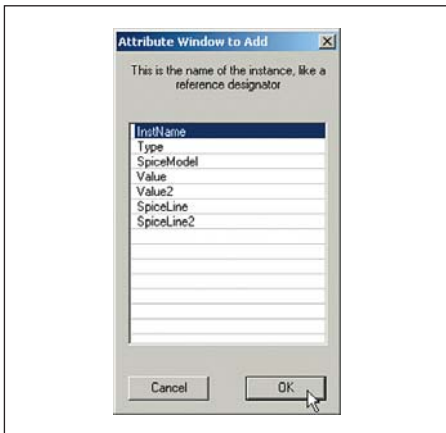


Рис. 28. Окно видимости атрибутов символа

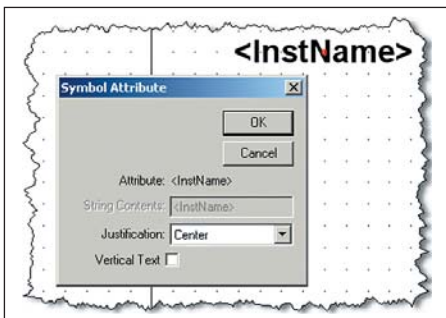


Рис. 29. Меню редактирования текста атрибута

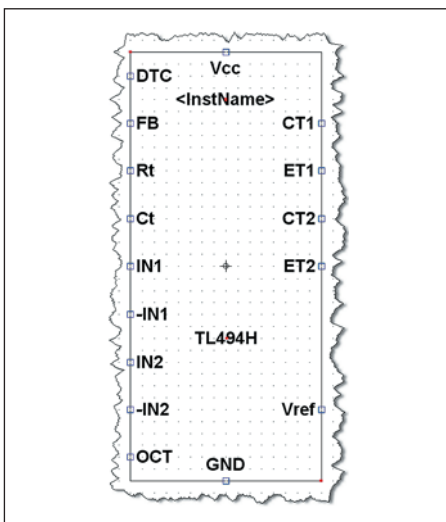


Рис. 30. Вид символа микросхемы TL494 после добавления атрибутов

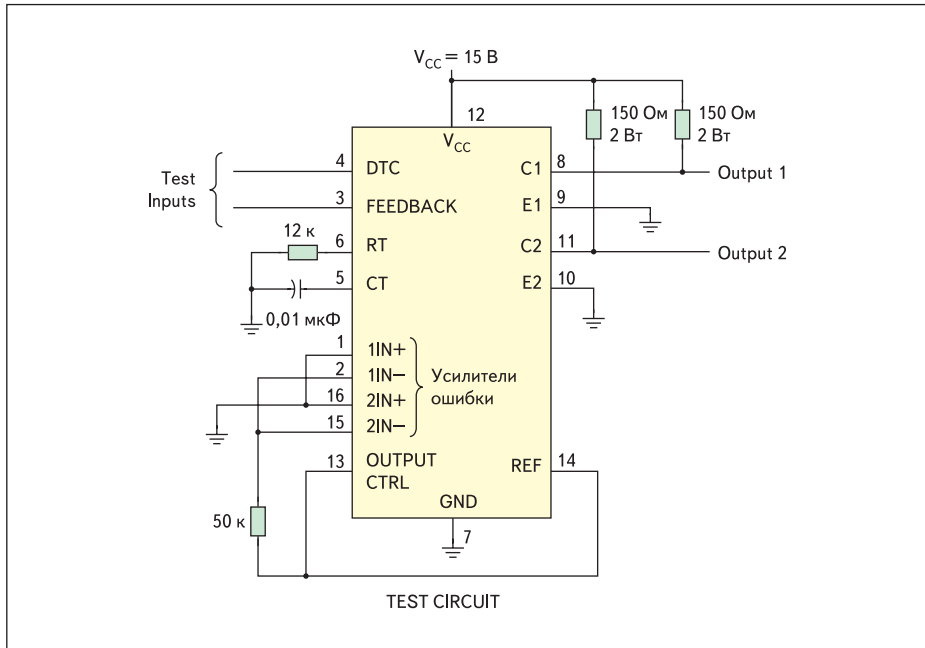


Рис. 31. Схема проверки ШИМ-контроллера TL494, рекомендуемая производителем

выравнивание и направление написания. Все это позволяет меню Symbol Attribute (рис. 29), которое вызывается щелчком правой кнопки мышки по тексту атрибута.

На рис. 30 показан внешний вид символа микросхемы TL494 после добавления атрибутов.

При помощи команды File=>Save As... назовем как TL494H.asy и сохраним в пользовательский каталог ..\lib\sym\ Mylib полученный символ.

**Проверка схемного элемента, созданного в виде иерархической схемы**

После того как созданы и размещены в пользовательском каталоге символ и модель схемного элемента, можно приступить к его проверке. Для проверки можно использовать тестовую схему из документации производителя (рис. 31).

Для этого запустим программу LTspice/SwitcherCAD III в режиме схемного редактора и проверим наличие нового электронного компонента в библиотеке. Если все сделано правильно, то ШИМ-контроллер TL494H можно обнаружить именно там, где он и должен быть, то есть в пользовательском каталоге Mylib (рис. 32).

Разместив символ в рабочем поле схемного редактора, щелкнем по нему, чтобы вызвать окно настройки Navigate/Edit Schematic Block. В верхней части окна расположены две кнопки — Open Symbol и Open Schematic, которые позволяют получить доступ к символу или схеме иерархического схемного блока. Ниже, в поле Instance Name необходимо указать позиционное обозначение схемного элемента. По умолчанию там, как для под-

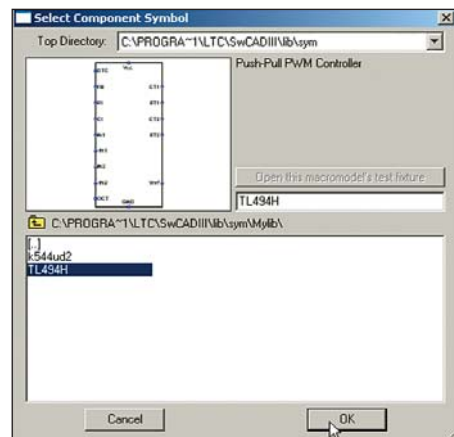


Рис. 32. Выбор ШИМ-контроллера TL494H из библиотеки пользователя

схемы, написано X1. Далее, в поле PARAMS можно перечислить параметры, которые необходимо передать в иерархическую схему. Окна выбора Visible позволяют включать и выключать видимость информации, введенной в соответствующем поле.

Иерархическому схемному элементу, в отличие от встроенных, можно присваивать позиционное обозначение с любым префиксом. Поэтому впишем в поле Instance Name более

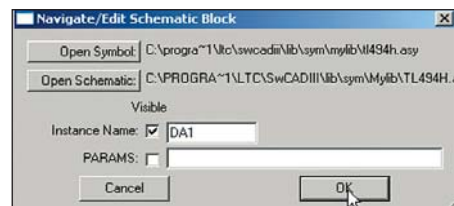


Рис. 33. Окно настройки иерархического схемного блока

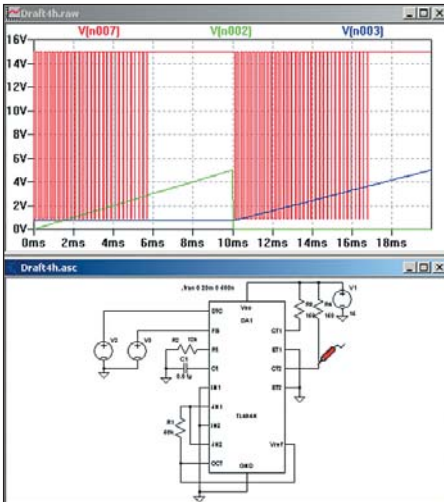


Рис. 34. Тестовая проверка модели микросхемы TL494, созданной в виде иерархической схемы

привычное наименование — DA1 и нажмем кнопку ОК (рис. 33).

Далее, в уже знакомой манере, создадим схему тестирования микросхемы TL494 и запустим процесс моделирования (рис. 34).

**Создание модели схемного элемента с использованием подсхемы**

Несмотря на многие положительные качества, такие как наглядность и удобство редактирования, иерархические схемы имеют и некоторые недостатки. Например, невозможно использовать модели в виде иерархической схемы на других SPICE-симуляторах, а также следует держать файл иерархической схемы в одном каталоге с основной схемой.

**Преобразование иерархической схемы в библиотечную подсхему**

Рассмотрим процесс преобразования иерархической схемы в соответствующую ей подсхему. Подобное преобразование можно произвести с помощью схемного редактора программы LTspice. Для этого загрузим в его окно иерархическую схему контроллера TL494H (рис. 18), созданную на предыдущих этапах. Далее можно сгенерировать список связей, но перед этим стоит сбросить флажок напротив опции Default Devices, расположенной на вкладке Netlist Options окна Control Panel, которое вызывается командой Tools=>Control Panel. Это позволит исключить ненужные строки определения моделей по умолчанию. После этого, при помощи команды View=>SPICE Netlist, вызовем окно просмотра списка связей иерархической схемы (рис. 35).

Теперь необходимо выделить весь текст списка связей, с помощью комбинации клавиш Ctrl-C скопировать его в буфер обмена, а затем вставить в окно любого текстового редактора, например в блокнот. Чтобы пре-

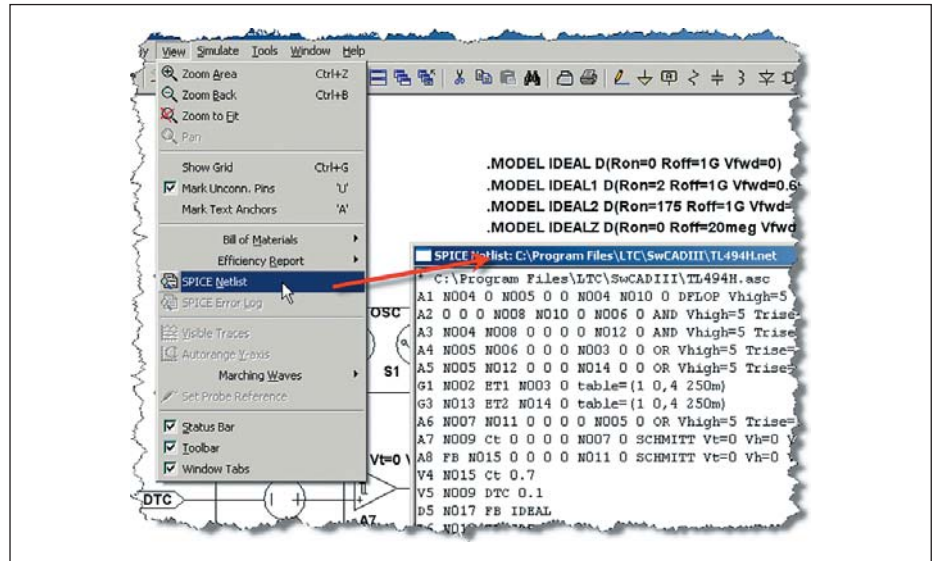


Рис. 35. Вызов списка связей иерархической схемы TL494

вратить полученный список связей в подсхему, необходимо вставить в него соответствующие директивы. В начале списка вставляется строка:

```
.subckt tl494 IN1 -IN1 IN2 -IN2 FB DTC Vref OCT CT1 ET1 CT2 ET2
Ct Rt GND Vcc
```

Порядок перечисления узлов должен соответствовать порядку, в котором эти узлы передаются в подсхему. Напомним, что он устанавливается для каждого вывода символа в поле Netlist Order (рис. 23) при его описании.

В конце списка удаляются строки с директивами .backanno и .end, а вместо них вставляется директива конца подсхемы .ENDS.

В результате должен получиться следующий список соединений:

```
*
.subckt tl494 IN1 -IN1 IN2 -IN2 FB DTC Vref OCT CT1 ET1 CT2 ET2
Ct Rt GND Vcc
A1 N005 0 N006 0 0 N005 N011 0 DFLOP Vhigh=5 Trise=50n Rout=30
A2 0 0 0 N009 N011 0 N007 0 AND Vhigh=5 Trise=50n Rout=30
A3 N005 N009 0 0 0 N013 0 AND Vhigh=5 Trise=50n Rout=30
A4 N006 N007 0 0 0 N004 0 0 OR Vhigh=5 Trise=300n Rout=30
A5 N006 N013 0 0 0 N015 0 0 OR Vhigh=5 Trise=300n Rout=30
G1 N002 ET1 N004 0 table=(1 0,4 250m)
G3 N014 ET2 N015 0 table=(1 0,4 250m)
A6 N008 N012 0 0 0 N006 0 OR Vhigh=5 Trise=50n Rout=30
A7 N010 Ct 0 0 0 N008 0 SCHMITT Vt=0 Vh=0 Vhigh=5
A8 FB N016 0 0 0 N012 0 SCHMITT Vt=0 Vh=0 Vhigh=5
V4 N016 Ct 0,7
V5 N010 DTC 0,1
D5 N018 FB IDEAL
D6 N020 FB IDEAL
R1 IN1 N017 1meg
R3 IN2 N019 1meg
C4 N017 -IN1 15,9n
C5 N019 -IN2 15,9n
D1 ET1 N002 IDEALZ
E1 N018 0 N017 -IN1 table=(0,0 1,96m,0 2m,2 2,04m,4 20m,4,5)
E2 N020 0 N019 -IN2 table=(0,0 1,96m,0 2m,2 2,04m,4 20m,4,5)
R5 -IN1 0 5meg
R6 IN1 0 5meg
R7 -IN2 0 5meg
R8 IN2 0 5meg
I1 FB 0 0,7m
V2 N001 0 3,65
F1 0 Ct V2 -1
S1 0 Ct N003 0 OSC
D3 CT1 N002 IDEAL1
D7 CT1 N002 IDEAL2
```

```
G2 ET1 N002 Vcc ET1 table=(0 250m,1,42 243m,1,46 150m,1,57 0)
D2 ET2 N014 IDEALZ
D4 CT2 N014 IDEAL1
D8 CT2 N014 IDEAL2
G4 ET2 N014 Vcc ET2 table=(0 250m,1,42 243m,1,46 150m,1,57 0)
C7 N002 ET1 5p
C8 N014 ET2 5p
D9 N001 Rt IDEAL
R14 Ct 0 5meg
R15 ET2 0 5meg
R16 ET1 0 5meg
G5 Vcc 0 Vcc 0 TABLE=(1 0,5 4,5m,6,85 7,6m,40 8,4m)
I2 0 Vref 25m
D10 0 Vref IDEAL5
B1 Vcc 0 I=(D10)+25m
R9 N009 OCT 1k
R10 Ct N003 5k
C1 N003 0 10p
.MODEL IDEALZ D(Ron=0 Roff=20meg Vfwd=0 Vrev=41)
.MODEL IDEAL D(Ron=0 Roff=1G Vfwd=0)
.model OSC SW(Ron=10 Vt=1,51 Vh=1,49 Ilimit=20m)
.MODEL IDEAL1 D(Ron=2 Roff=1G Vfwd=0,66)
.MODEL IDEAL2 D(Ron=175 Roff=1G Vfwd=0)
.MODEL IDEAL5 D(Ron=0 Roff=1G Vfwd=0 Vrev=5)
.ends tl494
```

Сохраним подсхему в пользовательскую директорию .lib\sym\Mylib под именем tl494.lib.

**Редактирование символа схемного элемента**

Загрузим в редактор символов созданный ранее символ TL494H.asy и с помощью команды Edit=>Attributes=>Edit Attributes или комбинации клавиш Ctrl-A вызовем окно редактирования атрибутов символа. Далее отредактируем его так, как это показано на рис. 36.

После этого сохраним символ в пользовательскую директорию .lib\sym\Mylib под именем tl494.asy.

**Проверка схемного элемента, созданного в виде подсхемы**

Для проверки схемного элемента, созданного в виде подсхемы, можно использовать тестовую схему (рис. 34), которая ранее уже использовалась для проверки схемного элемента, созданного в виде иерархической схемы. Для этого достаточно заменить схемный элемент TL494H на вновь созданный TL494. Так как схемный элемент создан в виде под-

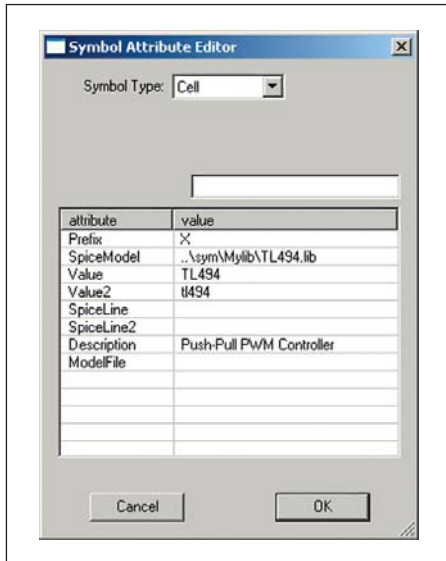


Рис. 36. Вид отредактированного окна атрибутов символа TL494

схемы, то редактирование его параметров в окне схемного редактора становится недо-

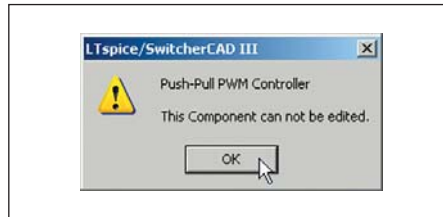


Рис. 37. Окно предупреждения о невозможности редактирования

ступным. В этом можно убедиться, щелкнув по символу элемента правой кнопкой мышки (рис. 37).

Далее запустим процесс моделирования (рис. 38).

### Литература

1. <http://ltspice.linear.com/software/scad3.pdf>
2. <http://groups.yahoo.com/group/LTspice>,  
<http://valvol.flyboard.ru/forum4.html>,  
<http://www.intactaudio.com/forum/viewforum.php?f=15>
3. <http://groups.yahoo.com/group/LTspice>
4. <http://forest2.homeip.net/Electronics/extra.rar>
5. <http://focus.ti.com/lit/ds/symlink/tl494.pdf>
6. <http://www.onsemi.com/pub/Collateral/TL494-D.PDF>
7. <http://focus.ti.com/lit/an/slva001d/slva001d.pdf>

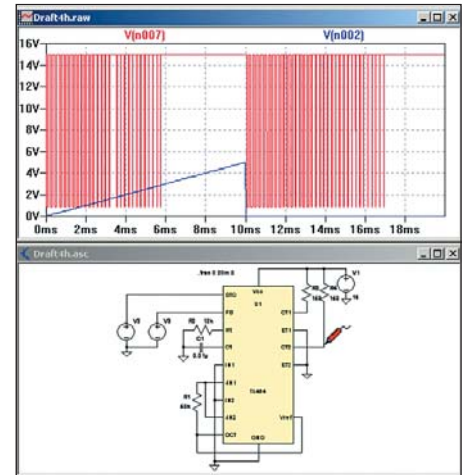


Рис. 38. Тестовая проверка модели микросхемы TL494, созданной в виде подсхемы