

Создание и редактирование моделей в OrCAD 15.7 (программа Model Editor). Урок 3

Иосиф ЗЛАТИН
zlatin@pochta.ru

Цель третьего урока — рассказать о добавлении информации Smoke к моделям и начать знакомство с созданием условных графических обозначений символов компонентов.

Обработка информации Smoke, используя Model Editor

Можно использовать Model Editor для добавления информации Smoke [5] к устройствам, поддерживаемым Model Editor. Это возможно, только если на компьютере установлена программа PSpice Advanced Analysis.

Используя Model Editor, можно:

- Добавлять информацию Smoke к моделям PSpice.
- Создавать модели PSpice, основанные на шаблоне с информацией Smoke.
- Редактировать информацию Smoke для устройств, поддерживаемых Model Editor.

Добавление информации Smoke к моделям PSpice позволяет использовать модели для выполнения Advanced Analysis Smoke. Используя Model Editor, можно добавить информа-

цию Smoke только к устройствам, поддерживаемым Model Editor.

Для моделей на основе шаблона информация Smoke присутствует по умолчанию на вкладке **Smoke Parameters** (рис. 12). Можно редактировать эту информацию, используя Model Editor. Для добавления информации Smoke к модели не на основе шаблона необходимо выполнить следующие шаги.

1. В окне Model List выберем модель, для которой должна быть добавлена информация Smoke. Если модель, которую необходимо отредактировать, имеет несколько вариантов выполнения, необходимо сначала выбрать выполнение, для которого должны быть сделаны изменения.
2. В меню **Model** выберем **Add Smoke**. Откроется диалоговое окно Add Smoke, если модель использует описание .SUBCKT.

3. В диалоговом окне Add Smoke определим тип устройства.

4. Нажмем **ОК**. Наряду с окном Model Text, которое отображает описание модели в текстовом формате, появятся вкладки **Test Node Mapping** и **Parameters Smoke**.

5. На вкладке **Test Node Mapping** введем название порта, которое отображается для каждого узла. Для моделей PSpice с описанием .SUBCKT названия узла определяются портами в описании подсхемы.

6. Введем максимальные рабочие величины для параметров на вкладке **Smoke Parameters**.

7. Сохраним модель. При сохранении модели файлы .LIB и .PRP обновляются.

Когда используется Model Editor для создания модели PSpice, основанной на характеристиках устройства, информация Smoke по умолчанию не доступна. После того как добавляется информация Smoke, создается новый файл LIB_NAME.PRP.

Создание модели PSpice на основе шаблона с информацией Smoke

Шаги для создания модели на основе шаблона с информацией Smoke — точно такие же, как шаги для создания имитационной модели, основанной на характеристиках устройства.

Если на компьютере установлена программа Advanced Analysis, позиция Smoke появится по умолчанию и не потребуются выбирать **Add Smoke** в меню **Model**.

Откроем модель с информацией Smoke в Model Editor. Можно изменить максимальные рабочие значения для различных параметров Smoke на вкладке **Smoke Parameters**. Для моделей, основанных на характеристиках устройства, можно также изменить порт к узлу, отображающемуся в окне Test Node Mapping. Сохраняя модель, обновим файл .PRP с самой последней информацией.

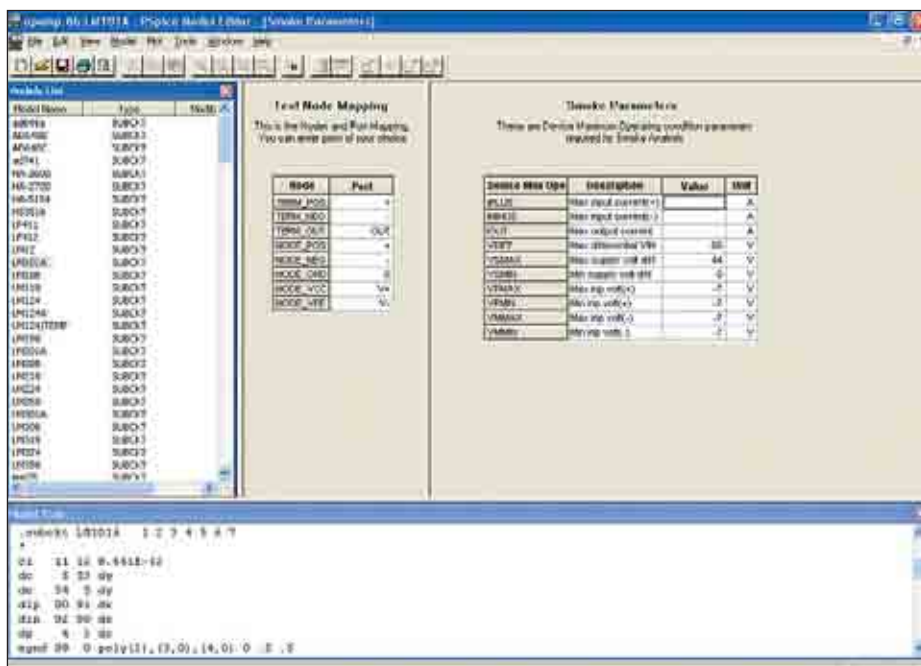


Рис. 12. Диалоговое окно программы Model Editor для моделей на основе шаблона

Пример. Добавление информации Smoke к модели диода D1

В этом примере будем применять Model Editor для добавления информации Smoke к модели диода D1, используемой в проекте однополупериодного выпрямителя, который был описан в примере «Создание модели PSpice, основанной на характеристиках устройства».

Для добавления информации Smoke выполним следующие шаги:

1. В меню **Model** выберем **Add Smoke**. Появятся вкладки **Test Node Mapping** и **Smoke Parameters**.

Замечание: команда **Add Smoke** разрешается, только если установлена программа **Advanced Analysis**.

2. Используя вкладку **Test Node Mapping**, добавим информацию из таблицы 4.

Таблица 4. Информация, добавляемая на вкладке Test Node Mapping для диода

Узел	Порт
TERM_AN	1
NODE_AN	1
NODE_CAT	2

Можно получить название порта, открывая символ в Capture.

- а) Выберем компонент в редакторе схем Capture.
 - б) В меню **Edit** выберем **Part**.
 - в) Отобразится символ компонента. Дважды щелкнем по выводу.
 - г) Отобразится поле **Name** в диалоговом окне **Pin Properties** (рис. 13).
3. Используя вкладку **Smoke Parameters**, добавим информацию Smoke из таблицы 5. Информация Smoke доступна в справочных данных изготовителя устройства.

Таблица 5. Информация Smoke для диода

Название свойства	Значение
IF	1
VR	30
PDM	1,5
TJ	175
RJC	50
RCA	50

4. Для сохранения изменений модели диода выберем **Save** в меню **File**. После добавления информации Smoke можно использовать анализ Smoke для схемы выпрямителя.

Добавление информации Smoke к модели операционного усилителя OPA_LOCAL

В этом примере добавим информацию Smoke к модели ОУ, созданного с помощью Model Editor. Можно использовать Model Editor для добавления и редактирования информации Smoke.

1. Откроем Model Editor.
2. Откроем библиотеку LOCAL_LIB.

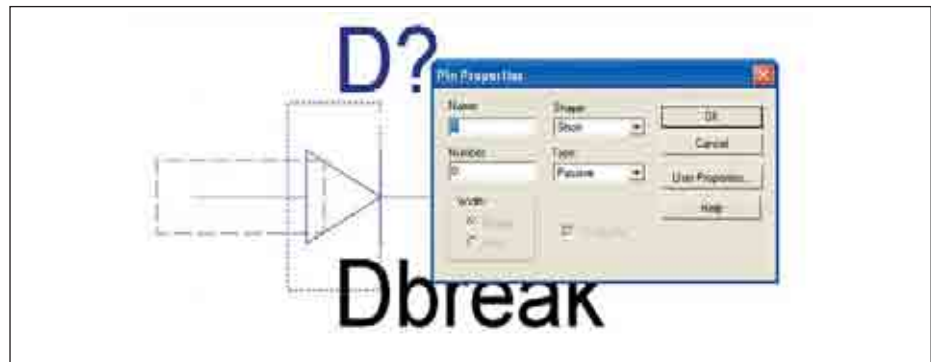


Рис. 13. Диалоговое окно Pin Properties

3. Выберем OPA_LOCAL_3. Помимо вкладок Simulation Parameters и Model Text отобразится вкладка **Smoke**.
4. Выберем вкладку **Smoke**. Отобразятся вкладки **Test Node Mapping** и **Smoke Parameters**. Используя вкладку **Smoke Parameters**, введем максимальные рабочие условия для ОУ, показанные в таблице 6.

Таблица 6. Информация Smoke для ОУ

Параметры Smoke	Значение
IPLUS	0,05
IMINUS	0,05
IOUT	0,04
VDIFF	32
VSMAX	32
VSMIN	-0,3
VPMAX	0,3
VPMIN	0,3
VMMAX	-1,5
VMMIN	0,3

5. Сохраним модель. В файл LOCAL_LIB.PRP будет записана введенная нами информация Smoke. Параметры Smoke, добавленные в этом примере, действительны для всех уровней. Даже если из библиотеки LOCAL_LIB удалится OPA_LOCAL_3, LOCAL_LIB.PRP не будет удален, потому что он содержит информацию Smoke для OPA_LOCAL_1 и OPA_LOCAL_2.

Параметры Smoke

Используя Model Editor, можно добавлять информацию Smoke к устройствам, поддерживаемым Model Editor. При добавлении информации Smoke к устройству в Model Editor отображаются следующие две вкладки (рис. 12):

- **Test Node Mapping.** На этой вкладке определяется название порта, которое должно отобразиться в предопределенном названии узла. Предопределенные оконечные названия вывода, начинающиеся с TERM, указывают текущие узлы, а те, которые начинаются с NODE, указывают узлы напряжения.
- **Smoke Parameters.** На этой вкладке определяются максимальные рабочие условия в терминах величин параметров Smoke.

Тепловые параметры, которые являются обычными для всех устройств, поддерживаемых Model Editor, перечислены ниже:

- PDM — максимально допустимая мощность рассеяния;
- TJ — максимальная температура p-n-перехода;
- RJC — тепловое сопротивление перехода;
- RCA — тепловое сопротивление между переходом и внешней средой.

Помимо параметров, перечисленных выше, параметры Smoke определяются классом устройств. Описание параметров Smoke для типов устройств, поддерживаемых Model Editor, дано автором ранее [5], а названия узлов для этих типов устройств дается в таблицах 7–14.

Для диода названия узлов приведены в таблице 7.

Таблица 7. Названия узлов для диода

Узел	Название порта
TERM_AN	Прямой токовый вывод
NODE_AN	Узел анодного напряжения
NODE_CAT	Узел катодного напряжения

Для биполярного транзистора названия узлов приведены в таблице 8.

Таблица 8. Названия узлов для биполярного транзистора

Узел	Название порта
TERM_IC	Токовый вывод коллектора
TERM_IB	Токовый вывод базы
NODE_VC	Узел коллекторного напряжения
NODE_VB	Узел базового напряжения
NODE_VE	Узел эмиттерного напряжения

Для транзистора IGBT названия узлов приведены в таблице 9.

Таблица 9. Названия узлов для транзистора IGBT

Узел	Название порта
TERM_IC	Токовый вывод коллектора
TERM_IG	Токовый вывод затвора
NODE_VC	Узел коллекторного напряжения
NODE_VG	Узел напряжения на затворе
NODE_VS	Узел напряжения истока


Для полевого транзистора с управляющим p - n -переходом  названия узлов приведены в таблице 10.

Таблица 10. Названия узлов полевого транзистора с управляющим p - n -переходом

Узел	Название порта
TERM_ID	Токовый вывод истока
TERM_IG	Токовый вывод затвора
NODE_VD	Узел напряжения истока
NODE_VG	Узел напряжения на затворе
NODE_VS	Узел напряжения истока


Для операционного усилителя  названия узлов приведены в таблице 11.

Таблица 11. Названия узлов операционного усилителя

Узел	Название порта
NODE_POS	Узел положительного источника питания (POS)
NODE_NEG	Узел отрицательного источника питания (NEG)
NODE_VCC	Узел положительного источника питания
NODE_VEE	Узел отрицательного источника питания
NODE_GND	Узел напряжения земли
TERM_POS	Положительный токовый вывод (POS)
TERM_NEG	Отрицательный токовый вывод (NEG)
TERM_OUT	Выходной токовый вывод (OUT)

Для транзистора MOSFET  названия узлов приведены в таблице 12.

Таблица 12. Названия узлов транзистора MOSFET

Узел	Название порта
TERM_ID	Токовый вывод стока
TERM_IG	Токовый вывод затвора
NODE_VD	Узел напряжения стока
NODE_VG	Узел напряжения затвора
NODE_VS	Узел напряжения истока


Для стабилизатора напряжения  названия узлов приведены в таблице 13.

Таблица 13. Названия узлов стабилизатора напряжения

Узел	Название порта
NODE_IN	Узел входного напряжения (IN)
NODE_OUT	Узел выходного напряжения (OUT)
NODE_GND	Узел напряжения земли (GND)

Для схемы Дарлингтона  названия узлов приведены в таблице 14.

Таблица 14. Названия узлов схемы Дарлингтона

Узел	Название порта
TERM_IC	Токовый вывод коллектора
TERM_IB	Токовый вывод базы
NODE_VC	Узел напряжения коллектора
NODE_VB	Узел напряжения базы
NODE_VE	Узел напряжения эмиттера

Создание УГО для моделей

В этом разделе рассмотрим создание УГО для моделей проекта. Здесь мы познакомимся с наиболее важными вопросами в создании УГО:

- Пути создания УГО для моделей.
- Как использовать инструментальные средства (использование Model Editor для создания УГО).
- Редактирование графических символов компонентов.
- Определение свойств УГО, необходимых для моделирования.
- УГО, используемые для моделирования, имеют особые характеристики:
- связь с имитационной моделью;
- список соединений;
- другие свойства, специфические для УГО — они могут включать скрытые выходы или задержку (для цифровых компонентов).

Если уже имеются описания моделей и необходимо создать УГО для них, необходимо поместить описания компонентов в библиотеки, содержащие аналогичные типы устройств. Библиотеки Model обычно имеют расширение .LIB. Однако можно использовать другое расширение, если формат файла соответствует стандартному формату библиотеки моделей.

Необходимо удостовериться, что название модели в новой библиотеке не совпадает с названиями моделей в других библиотеках.

Для запуска Model Editor:

1. В меню Windows **Пуск** выберем **Release OrCAD 15.7/PSpice Accessories/Model Editor**.
2. В меню **File** выберем **Open** или **New** и введем существующее или новое название библиотеки моделей.
3. На вкладке **Models List** выберем название модели для ее редактирования на вкладке **Spec Entry**.

Для запуска Model Editor из Capture:

1. В схематическом редакторе страницы выберем УГО, модель которого необходимо отредактировать.
2. В меню **Edit** выберем **PSpiceModel**. Программа Model Editor запустится с моделью, загруженной для редактирования.

Если программа Model Editor запущена из Capture, и необходимо продолжить работать с новыми моделями, то:

1. Закройте открытую библиотеку моделей.
2. Откройте новую библиотеку моделей.
3. Загрузите модель устройства или создайте новую модель.

Замечание: создание УГО заблокировано, когда Model Editor запускается из Capture.

Использование Model Editor для создания УГО

Если необходимо создать новые УГО, которые не привязаны к локальному проекту, откройте только Model Editor. Используя Model Editor, можно создать УГО в двух режимах:

- пакетном;
- диалоговом.

В пакетном режиме для создания символов для всех моделей в библиотеке используется

команда **Export to Capture Part Library** в меню **File**. При этом подходе можно рассмотреть символы только после того, как все изменения будут сделаны и сохранены в файле .olb. Необходимо открыть файл .olb в Capture и рассмотреть символы.

В диалоговом режиме создания УГО можно рассмотреть символ, присоединенный к имитационной модели до сохранения изменений в библиотеке символов. Можно также обновить и присоединить символы, выбранные к моделям в библиотеке. Для создания символов Capture в диалоговом режиме выберем в меню **File** команду **Model Import Wizard [Capture]**.

Когда символы УГО генерируются с помощью мастера Model Import Wizard, можно сделать обзор форм символа перед сохранением изменений в библиотеке УГО. Это не похоже на использование команды **Export To Capture Part Library**, где все изменения делаются в библиотеке УГО до обзора символов, при открывании их в Capture.

В Model Import Wizard определим входную библиотеку моделей и выходную библиотеку символов и нажмем **Next**. Названия моделей и названия символа, которые будут связаны с каждой моделью, включены в список. Форма символа видна справа.

Если пользователи не делают никаких изменений и выбирают кнопку **Finish**, может появиться окно сообщений. Это окно сообщений появляется, только если есть модели, для которых не найдены символы. Присоединим прямоугольные символы к таким моделям, нажав **Yes**. Если нет необходимости присоединять символы, выбирают **No**. Если выбирается **Yes**, то сгенерированная библиотека символа с использованием Model Import Wizard будет точно такой же, как библиотека .olb, сгенерированная с помощью команды **Export to Capture Part Library**.

Для использования пакетного режима:

1. Откроем только Model Editor.
2. В меню Windows **Пуск** выберем **Programs/Release OrCAD 15.7/PSpice Accessories/Model Editor**.
3. В меню **File** выберем **Export to Capture Part Library**. Появится диалоговое окно **Create Parts for Library**. Опция **Export to Capture Part Library** является доступной, только если в качестве редактора схем в диалоговом окне Options выбран Capture. Для отображения диалогового окна Options выберем **Options** в меню **Tools**.

В текстовом поле **Enter Input Model Library** определим местоположение библиотеки моделей, для которой должно быть создано УГО Capture.

Замечание: можно использовать кнопку **Browse** для определения другого местоположения файла .OLB.

Текстовое поле **Enter Output Part Library** автоматически отображает название и местоположение нового создаваемого файла

.OLB. Отображаемое название библиотеки совпадает с тем, что определено пользователем в разделе **Save Part To** диалогового окна Options.

- Нажмем **OK** для создания УГО и нажмем **OK** еще раз для очищения диалогового окна .ERR log.

Библиотека символов, созданная повторно с помощью команды **Export to Capture Part Library**, не записывается поверх содержания библиотеки символов. Только новые УГО добавляются в библиотеку. Рассмотрим библиотеку символов MYLIB.OLB, которая имеет два УГО — a1 и a2, и библиотеку моделей MYLIB.LIB, которая имеет три модели — a2, b1 и b3. Библиотека MYLIB.OLB, созданная повторно из MYLIB.LIB, не будет удалять a1 из MYLIB.OLB. Измененная библиотека MYLIB.OLB имеет четыре УГО: a1, a2, b1 и b2.

Model Import Wizard не рекомендуется использовать, если необходимо создать новую форму символа. С помощью мастера можно только связать существующую модель PSpice с существующим символом и наоборот. Новые символы можно создать в редакторе схем Capture OrCAD.

Для запуска мастера Model Import Wizard в Model Editor в меню **File** необходимо выбрать **Model Import Wizard [Capture]**.

Для того чтобы сгенерировать УГО в редакторе схем Capture:

- Выберем окно Project Manager в Capture.
- В раскрывающемся меню **Tools** выберем **Generate Part**.
- В диалоговом окне Generate Part (рис. 14) выберем флажок **Pick symbol manually**.
- В раскрывающемся списке **Netlist/source file type** выберем **PSpice Model Library**.
- Для запуска процесса генерации символа нажмем **OK**.

Для использования Model Import Wizard:

- На странице **Specify Library** мастера Import Wizard определим название и местоположение входной библиотеки моделей (.lib) и выходной библиотеки символов (.olb),

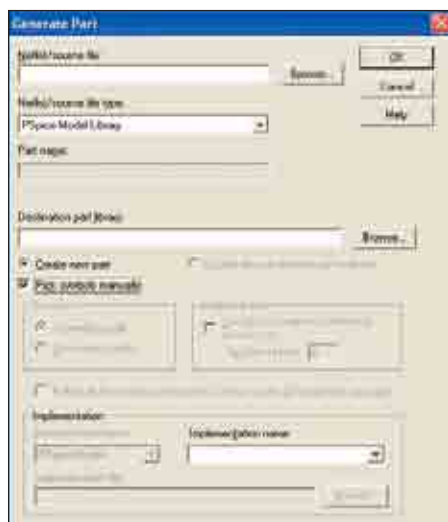


Рис. 14. Диалоговое окно Generate Part

затем, нажав кнопку **Next**, переместимся на следующий шаг.

Когда нажимается кнопка **Next**, Model Import Wizard автоматически запускает процесс поиска символов для каждой из моделей в файле .lib и присоединяет символы, подбираемые к описаниям моделей. Model Import Wizard использует описания модели для обнаружения соответствующего символа для имитационной модели. Поэтому Model Import Wizard может автоматически согласовать символы только для устройств, поддерживаемых Model Editor. Соответствующие символы хранятся в заданной библиотеке символов, определенной пользователем.

- Просмотрим символы, имеющиеся в Model Import Wizard. Страница мастера **Associate/Replace Symbol** перечисляет модели в файле .lib и соответствующие названия символа. По умолчанию, все модели внесены в список.

Для отображения моделей с присоединенными символами должен быть выбран флажок **Models with symbols**. Для отображения моделей без присоединенных символов должен быть выбран флажок **Models without symbols**. В этом случае ячейка **Symbol Name** пуста.

Можно рассмотреть присоединенный символ, выбирая модель из списка моделей с присоединенными символами. Форма символа появляется справа.

На данном этапе можно закончить процесс присоединения символов и закрыть Model Import Wizard, нажав кнопку **Finish**. Когда нажимается кнопка **Finish**, появляется диалоговое окно с вопросом (рис. 15): «хотите ли вы присоединить прямоугольные символы к моделям, которые не имеют символов, прикрепленных к ним?» Это сообщение появляется, только если файл .lib имеет модели, для которых символы недоступны.

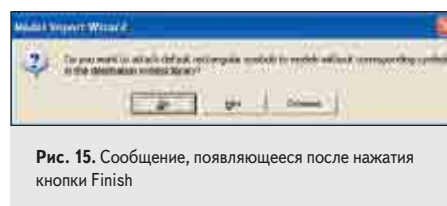


Рис. 15. Сообщение, появляющееся после нажатия кнопки Finish

В этом случае символы, имеющие прямоугольную форму, могут быть связаны с моделями при выборе **Yes**. Для закрытия мастера без прикрепления прямоугольных символов к моделям выберем **No**.

- Model Import Wizard позволяет присоединить определенные пользователем символы к моделям без символов или заменить присоединенные символы символом, выбранным пользователем.

– Для изменения предлагаемой связи модели-символа Model Import Wizard нажмем кнопку **Replace Symbol**.

– Для присоединения определенного символа к модели, для которой не могло быть найдено соответствующего символа, нажмем кнопку **Associate Symbol**.

Присоединим определенные пользователем символы к модели:

- Нажмем кнопку **Associate/Replace**.
- На странице мастера **Select Matching** определим путь к основной библиотеке. Основная библиотека может быть определена как библиотека символов (.olb), которая содержит желательный компонент. В указанной библиотеке мастер Model Import Wizard фильтрует и вносит названия символов, которые могут быть присоединены к выбранной модели, в список. Можно затем выбрать символ из списка, чтобы связать его с моделью.
- Используем кнопку **View Model Test** для отображения описания модели для выбранного режима.
- Нажмем кнопку **Save**.

Замечание: изменения, сделанные в библиотеке адресата, необратимы. После того как файл сохранен, изменения нельзя будет уничтожить нажатием кнопки. Точно так же можно связать существующий символ с любой из моделей в файле .lib.

Когда используется мастер Model Import Wizard для прикрепления символа к модели и наоборот, выполняются следующие задачи:

- Значение свойства IMPLEMENTATION TYPE — PSpice Model присоединяется к символу.
- Значение свойства IMPLEMENTATION, устанавливающее название выбранной модели, присоединяется к символу.
- Для моделей, основанных на характеристиках устройства, свойство PSpiceTEMPLATE обновляется с информацией **pin name**.
- Для моделей, базирующихся на шаблоне PSpice, генерируется вид **pspice_ink**.

Если разрешено автоматическое создание УГО для новых моделей:

- в программе Model Editor, в которой открыта библиотека, в меню **Tools** выберем **Options**.
- в рамке **Part Creation Setup** выберем **Always Create Part When Saving Model**.
- Выберем флажок **Pick symbols Manually** для обеспечения генерации символа УГО для модели с помощью Model Import Wizard.
- в рамке **Save Part To** определим название библиотеки символов для нового компонента. Выберем одно из двух:
 - Путь к библиотеке символов такой же, как и к библиотеке моделей для создания или открытия файла *.OLB, имя которого совпадает с именем открытой библиотеки моделей (*.LIB).

Например, если библиотека моделей называется MYPARTS.LIB, то Model

Editor создает библиотеку УГО по имени MYPARTS.OLB.

- Введем название библиотеки символов, определенной пользователем в текстовое поле **Library Name**.

Пример использования Model Editor для создания УГО в проекте

В этом разделе создадим усилитель, использующий два ОУ с внутренней компенсацией. Один из операционных усилителей — CA1458 — поддерживается PSpice в библиотеке моделей OPA.LIB, а второй используемый операционный усилитель — OPA_LOCAL — создан с помощью Model Editor. Обе модели основаны на шаблоне и обе поддерживают несколько уровней параметров моделирования.

1. Создадим в Capture новый проект FUNC_GEN.OPJ и включим LOCAL_LIB.LIB как одну из библиотек моделей.

2. Создадим схему, как показано на рис. 16. Модель CA1458 является многоуровневой моделью и по умолчанию уровень модели — 3. Свойство LEVEL для модели OPA_LOCAL устанавливается 2.

Замечание: можно также выбрать проект в папке ... \tools\pspice\tutorial\capture\modeleditor.

3. Для моделирования схемы в меню PSpice в Capture выберем **Run**.

Теперь изменим проект схемы так, что для CA1458 будет использован уровень параметров моделирования 2, а для модели OPA_LOCAL, созданной с помощью Model Editor, будет использован уровень параметров моделирования 3.

Изменим уровень параметров моделирования для CA1458 с 3 на 2:

1. Выберем УГО CA1458.
2. В меню **Edit** выберем **Properties**.
3. В диалоговом окне Property editor изменим значение свойства LEVEL с 3 на 2.
4. Нажмем **Apply** и закроем **Property Editor**.

Точно так же изменим значение свойства LEVEL для OPA_LOCAL с 2 на 3.

Можно теперь промоделировать схему, используя PSpice или PSpice Advanced Analysis.

Изменение свойств моделирования в Capture

Для всех доступных для редактирования свойств моделирования можно определить значение параметров моделирования в Capture.

1. Выберем OPA_LOCAL.
2. В меню **Edit** выберем **Properties**.
3. В диалоговом окне Property Editor выберем **New Row**.
4. В диалоговом окне Add New Row введем название параметра моделирования в текстовое поле **Name**. Введем VOS.
5. В текстовое поле **Value** добавим значение параметра моделирования $2e-3$ и нажмем **OK**.
6. Для того чтобы сделать свойство и его значение видимым, сначала выберем ячейку свойства, а затем в диалоговом окне Property Editor нажмем кнопку **Display**.
7. Определим **Display Format**. Выберем **Name** и **Value**.
8. Выберем **Apply** и закроем диалоговое окно.

Параметр моделирования и его значение появятся в Capture. При моделировании проекта будет использоваться значение параметра моделирования, определенное в Capture.

Пример создания УГО в диалоговом режиме

В этом разделе используем Model Import Wizard для создания модели, сохраняемой в личной библиотеке пользователя MYLIB. Эта библиотека состоит из четырех имитационных моделей: LM339 — модель подсхемы компаратора напряжения, INA105E — модель операционного усилителя плюс прецизионный резистор, LF442A/NS — модель операционного усилителя, построенного на полевых транзисторах JFET, и mybjt — модель биполярного транзистора.

1. Для запуска Model Import Wizard в меню **File** в Model Editor выберем **Model Import Wizard [Capture]**.
2. На странице мастера **Specify Library** определим путь к MYLIB.LIB (рис. 17).
3. Определим название библиотеки символов, в которой должны быть сохранены УГО,

сгенерированные Model Import Wizard. Можно либо определить название существующего файла .olb, либо создать новый. Когда определяется название существующего файла, необходимо также указать, должны ли символы создаваться для всех моделей в библиотеке моделей, заменяя существующие символы, или символы УГО должны создаваться только для моделей без символов.

- Название и местоположение библиотек символов заполняются по умолчанию. Для текущего примера примите название по умолчанию mylib.olb и нажмите **Next (Далее)**.

- Название символа, присоединенное к модели mybjt, внесено в список в столбец **Symbol Name**, рядом можно наблюдать форму символа (рис. 18).

4. Теперь выполним один из следующих шагов:

а) Закроем мастер, не присоединяя символы остальных трех моделей в библиотеке mylib.lib.

б) Закроем мастер, после того как Model Import Wizard присоединит прямоугольные символы к остальным моделям в mylib.lib.

в) Используем Model Import Wizard для выбора и присоединения существующей формы символа к одной или ко всем моделям в mylib.lib.

Для выполнения первых двух команд нажмем кнопку **Finish**. В появившемся окне сообщений выберем **No**, чтобы просто закрыть библиотеку, и **Yes**, чтобы прикрепить прямоугольные символы, а затем закрыть библиотеку.

Теперь будем использовать Model Import Wizard для присоединения существующих определенных пользователем символов к модели LF442A/NS. LF442A — маломощный двойной операционный усилитель, загруженный с сайта компании National Semiconductor. Загрузим символ из библиотеки oramp.olb, которая содержит символы для различных типов ОУ.

1. На странице **Associate/Replace Model Import Wizard** выберем LF442A/NS и нажмем **Associate Symbol**.

2. На странице мастера **Select Matching** определим название и местоположение oramp.olb. Все символы, которые подходят к описанию модели, перечислены в списке **Matching Symbols**.

3. В списке **Matching symbols** выберем LM158 и нажмем **Next**.

4. До запуска отображения выводов модели для обозначения выводов символа нажмем кнопку **View Model Text**. В другом окне появится описание модели LF442A/NS (рис. 19).

5. Отобразим выходы модели и названия выходов символа (табл. 15).

6. Для присоединения символа к LF442A/NS нажмем кнопку **Save Symbol**. Название символа, присоединенного к модели, по-

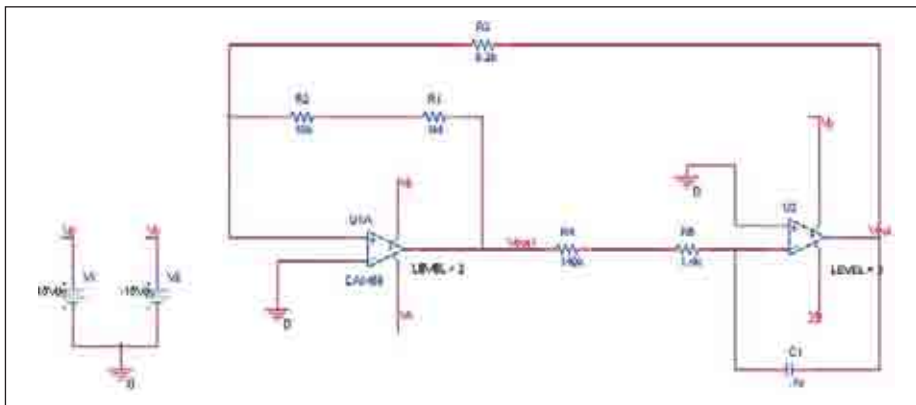


Рис. 16. Схема примера

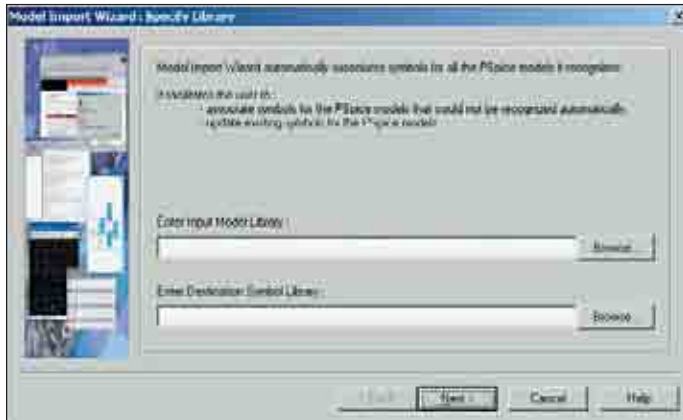


Рис. 17. Диалоговое окно Model Import Wizard

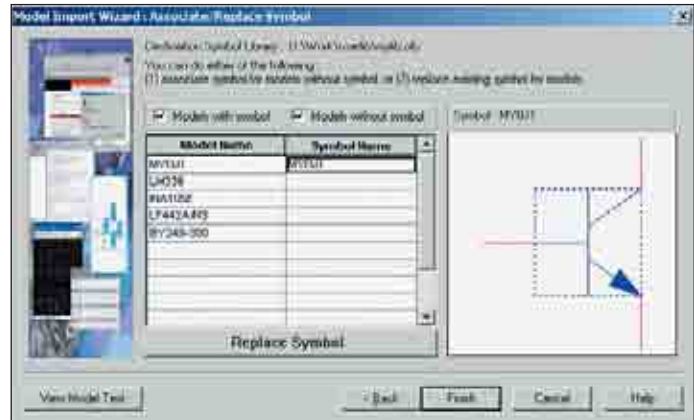


Рис. 18. Диалоговое окно Associate/Replace Symbol

Таблица 15. Отображение выводов модели и названия выводов символа

Выводы модели	Отображение выводов символа
1	+
2	-
99	V+
50	V-
28	OUT

явится в списке **Symbol Name**. Точно так же можно присоединить модели для других двух символов.

Замечание: можно использовать *Import Model Wizard* для замены символа, присоединенного к модели, на другой символ. Для этого выберем модель, к которой присоединен символ, и нажмем кнопку *Replace Symbol*.

Базирование новых УГО на пользовательских установках компонентов

Если Model Editor используется для автоматического генерирования УГО для описаний моделей, и необходимо, чтобы новые УГО базировались на пользовательском графическом стандарте (это предпочтительнее, чем предлагаемые по умолчанию УГО PSpice), то можно заменить базовые УГО, устанавливая собственный набор УГО.

Замечание: если используется собственный набор УГО, Model Editor всегда проверяет пользовательскую библиотеку символов сначала для УГО, которые соответствуют описанию модели. Если ни одно УГО не найдено, Model Editor использует УГО PSpice.

Создание пользовательских установок УГО для автоматической генерации УГО

Создадим библиотеку символов с пользовательскими УГО. Убедимся, что название этих УГО соответствует типам устройств, показанным в таблицах 16 и 17.

Для применения пользовательских УГО:

1. В программе Model Editor, в которой открыта библиотека, в диалоговом окне

```

* LF442A Dual Low Power JFET Input Operational Amplifier
* ////////////////////////////////////////////////////
*
* Connections:  Non-inverting input
*                Inverting input
*                Positive power supply
*                Negative power supply
*                Output
*
* .SUBCKT LF442A/NS 1 2 99 50 28
*
* Features:
    
```

Рис. 19. Описание модели LF442A/NS

Таблица 16. Название стандартных пользовательских символов моделей PSpice

Тип устройства	Название символа	Тип устройства	Название символа
Биполярный транзистор LPNP	LPNP	Магнитный сердечник	CORE
Биполярный транзистор PNP	PNP	Транзистор MOSFET с л-каналом	NMOS
Биполярный транзистор NPN	NPN	Транзистор MOSFET с р-каналом	PMOS
Конденсатор ¹	CAP	Операционный усилитель с пятью выводами	OPAMP5
Схема Дарлингтона с л-каналом	DARNPN	Операционный усилитель с семью выводами	OPAMP7
Схема Дарлингтона с р-каналом	DARPNP	Резистор ¹	RES
Диод	DIOD	Переключатель, управляемый напряжением ¹	VSWITCH
Транзистор GaAsFET ¹	GASFET	Линия передачи	TRN
Транзистор IGBT с л-каналом	NIGBT	Компаратор напряжения	VCOMP
Индуктивность ¹	IND	Компаратор напряжения с шестью выводами	VCOMP6
Транзистор JFET с л-каналом	NJF	Источник опорного напряжения	VREF
Транзистор JFET с р-каналом	PJF	Стабилизатор напряжения	VREG

¹ Не применяются в Model Editor

Таблица 17. Названия символов для моделей на основе шаблона

Тип устройства	Название символа	Тип устройства	Название символа
Биполярный транзистор LPNP		Магнитный сердечник	AACORE
Биполярный транзистор PNP	AANPN3	Транзистор MOSFET с л-каналом	AANMOSFET3
Биполярный транзистор NPN	AAPNP3	Транзистор MOSFET с р-каналом	AAPMOSFET3
Конденсатор ¹	CAP	Операционный усилитель с пятью выводами	AA5_PIN_OPAMP
Схема Дарлингтона с л-каналом	AADARNPN3	Операционный усилитель с семью выводами	AA7_PIN_OPAMP
Схема Дарлингтона с р-каналом	AADARPNP3	Резистор ¹	
Диод	AADIODE	Переключатель, управляемый напряжением ¹	
Транзистор GaAsFET ¹		Линия передачи	
Транзистор IGBT с л-каналом	AANIGBT3	Компаратор напряжения	
Индуктивность ¹		Компаратор напряжения с шестью выводами	
Транзистор JFET с л-каналом	AANCHANNEL3	Источник опорного напряжения	
Транзистор JFET с р-каналом	AANPHANNEL3	Стабилизатор напряжения	AAVREG

¹ Не применяются в Model Editor

Примечание: для каждого пользовательского УГО устанавливается свойство IMPLEMENTATION

- Options** выберем **Part Creation Setup** и разрешим автоматическое создание УГО.
2. В рамке **Base Parts On** введем название существующей библиотеки УГО (*.OLB), которое содержит пользовательский компонент.
 3. Нажмем **ОК**. ■

Литература

1. Разевиг В. Д. Система проектирования OrCAD 9.2. М.: Солон-Р, 2001.
2. Златин И. Моделирование на функциональном уровне в OrCAD 9.2 // Компоненты и технологии. 2003. №№ 3, 4.
3. Златин И. В Монте-Карло с OrCAD 9.2 // Компоненты и технологии. 2003. № 5.
4. Златин И. Графический анализ результатов моделирования в OrCAD 9.2 // Компоненты и технологии. 2003. № 7.
5. Златин И. Расширенный анализ (Advanced Analysis) и режим анализа Smoke в PSD 15.0 и OrCAD 10.0 // Компоненты и технологии. 2004. № 4.
6. Златин И. Advanced Analysis и режим анализа Sensitivity в PSD 15.0 и OrCAD 10.0 // Компоненты и технологии. 2004. № 5.
7. Златин И. Advanced Analysis и режим анализа Optimizer в PSD 15.0 и OrCAD 10.0 // Компоненты и технологии. 2004. № 6, 8.
8. Златин И. Advanced Analysis и режим анализа Monte Carlo в PSD 15.0 и OrCAD 10.0 // Компоненты и технологии. 2005. № 9.
9. Златин И. Advanced Analysis и режим анализа Troubleshooting в PSD 15.0 и OrCAD 10.0 // Компоненты и технологии. 2006. № 1.
10. Златин И. Программа Advanced Analysis и режим анализа Parametric Plotter в OrCAD 10.5 // Компоненты и технологии. 2006. № 2.
11. Златин И. OrCAD 10.5 для начинающих пользователей // Компоненты и технологии. 2006. № 3, 4.
12. Златин И., Хамзин Н. Программа Transformer Designer (разработчик трансформаторов) в OrCAD 10.5 // Компоненты и технологии. 2006. № 5–8.
13. Златин И. Создание и редактирование моделей в OrCAD 15.7 (программа Model Editor) // Компоненты и технологии. 2007. № 6, 7.