

Altium Designer

проектирование функциональных узлов РЭС на печатных платах

.....

- *Формирование библиотек компонентной базы*
- *Проектирование электрической принципиальной схемы*
- *Схемотехническое моделирование в Altium Designer*
- *Ручная и автоматическая трассировка печатного монтажа*
- *Моделирование паразитных эффектов в печатном монтаже*
- *Проектирование функционального узла на ПЛИС*
- *Формирование конструкторской документации по ЕСКД*

Владислав Суходольский

Altium Designer

проектирование функциональных узлов РЭС на печатных платах

Рекомендовано учебно-методическим объединением вузов Российской Федерации по образованию в области радиотехники, электроники, биомедицинской техники и автоматизации в качестве учебного пособия для студентов высших учебных заведений, обучающихся по специальностям 210201 "Проектирование и технология радиоэлектронных средств" и 210202 "Проектирование и технология электронно-вычислительных средств" направления 210200 "Проектирование и технология электронных средств"

Санкт-Петербург

«БХВ-Петербург»

2010

УДК 681.3.06
ББК 32.973.26-018.2
С91

Суходольский В. Ю.

С91 Altium Designer: проектирование функциональных узлов РЭС на печатных платах. — СПб.: БХВ-Петербург, 2010. — 480 с.: ил. — (Учебное пособие)

ISBN 978-5-9775-0542-0

Книга посвящена проектированию радиоэлектронных функциональных узлов в среде Altium Designer. Описаны состав, настройка и основные приемы работы в среде Altium Designer. Подробно освещены вопросы формирования и редактирования электрической схемы, разработки печатной платы, а также трассировки печатного монтажа. Отдельно рассмотрены особенности реализации проекта на основе микросхем ПЛИС. Значительное внимание уделено схемотехническому моделированию. Приведены необходимые сведения о работе с библиотеками, взаимодействии с внешними базами данных, системе контроля версий, а также экспорте результатов. Особенность книги — изложение материала с позиций сквозного проектирования изделия, начиная от создания нового проекта и заканчивая выпуском конструкторской документации по ЕСКД.

*Для студентов вузов радиоэлектронного профиля,
а также разработчиков, использующих Altium Designer*

Рецензенты:

И. Ю. Пивоваров, к.т.н., доцент кафедры радиоэлектронных средств Санкт-Петербургского государственного электротехнического университета;
Л. М. Макаров, к.т.н., доцент, заместитель проректора по научной работе Санкт-Петербургского государственного университета телекоммуникаций им. проф. М. А. Бонч-Бруевича.

УДК 681.3.06
ББК 32.973.26-018.2

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Леонид Кочин</i>
Компьютерная верстка	<i>Натали Смрновой</i>
Корректор	<i>Наталия Першакова</i>
Дизайн серии	<i>Инны Тачиной</i>
Оформление обложки	<i>Елены Беляевой</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 30.01.10.
Формат 70×100¹/₁₆. Печать офсетная. Усл. печ. л. 38,7.
Тираж 1500 экз. Заказ №
"БХВ-Петербург", 190005, Санкт-Петербург, Измайловский пр., 29.

Санитарно-эпидемиологическое заключение на продукцию № 77.99.60.953.Д.005770.05.09 от 26.05.2009 г. выдано Федеральной службой по надзору в сфере защиты прав потребителей и благополучия человека.

Отпечатано с готовых диапозитивов
в ГУП "Типография "Наука"
199034, Санкт-Петербург, 9 линия, 12

Оглавление

ВВЕДЕНИЕ.....	1
ГЛАВА 1. СОСТАВ И БАЗОВЫЕ КОНЦЕПЦИИ ALTIUM DESIGNER.....	5
1.1. Основные функции Altium Designer	5
1.2. Проект Altium Designer	7
1.2.1. Виды проектов Altium Designer	7
1.2.2. Создание нового проекта	9
1.2.3. Включение документов в проект.....	10
1.2.4. Работа с документами проекта	12
ГЛАВА 2. БИБЛИОТЕКИ КОМПОНЕНТНОЙ БАЗЫ	15
2.1. Базовые концепции библиотечного обеспечения.....	15
2.2. Создание новой интегрированной библиотеки	16
2.2.1. Формирование схемных компонентов.....	17
2.2.2. Присоединение электрических выводов	22
2.2.3. Формирование и редактирование схемных символов интегральных микросхем	25
2.2.4. Многосекционные компоненты.....	29
2.2.5. Выводы питания и "земли"	30
2.2.6. Редактирование свойств компонента.....	32
2.3. Библиотеки топологических посадочных мест	33
2.3.1. Создание новой библиотеки посадочных мест.....	33
2.3.2. Формирование топологического посадочного места	35
Размещение контактных площадок.....	37
Черчение контуров корпуса компонента	39
Защитная маска и маска припойного трафарета	40
2.4. Присоединение моделей к схемному компоненту	41
2.4.1. Подключение модели посадочного места	41
2.4.2. Подключение SPICE-модели	44
2.4.3. Подключение моделей Signal Integrity.....	45
2.5. Компиляция интегрированной библиотеки	47
2.6. Конверсия библиотек P-CAD 200x в формат Altium Designer	48
2.7. Включение библиотек в рабочую среду Altium Designer	53
2.8. Поиск компонентов в интегрированных библиотеках.....	55

ГЛАВА 3. НАСТРОЙКА КОНФИГУРАЦИИ ГРАФИЧЕСКИХ РЕДАКТОРОВ57

3.1. Конфигурация графического редактора схем	57
3.1.1. Лист схемы по ЕСКД.....	57
3.1.2. Реквизиты документа	61
3.1.3. Заполнение основной надписи документа.....	62
3.2. Настройка конфигурации графического редактора печатной платы	65
3.2.1. Задание размеров листа	65
3.2.2. Сетки проектирования.....	67
3.2.3. Редактирование контуров заготовки.....	71
3.2.4. Слои графического редактора печатной платы	72
3.2.5. Менеджер структуры слоев.....	74
3.2.6. Настройка правил проектирования	75
3.2.7. Барьеры трассировки	76
3.2.8. Подключение бланка форматки стандартных листов	77

ГЛАВА 4. ФОРМИРОВАНИЕ И РЕДАКТИРОВАНИЕ ЭЛЕКТРИЧЕСКОЙ СХЕМЫ79

4.1. Размещение объектов на поле чертежа	80
4.1.1. Размещение схемных символов	80
4.1.2. Черчение линий электрической связи.....	85
Прокладка проводников	85
Формирование изломов проводника.....	86
Соединение и пересечение проводников.....	86
4.1.3. Имена и признаки связности цепей.....	86
4.1.4. Линии групповой связи	92
4.2. Редактирование электрической схемы	94
4.2.1. Редактирование проводников	95
Перемещение изломов проводника.....	95
Перемещение сегментов проводника.....	95
Продолжение проводника	95
Разрыв проводника	95
4.2.2. Перемещение компонентов	96
4.2.3. Копирование и вставка копий	98
4.2.4. Присвоение позиционных обозначений	98
4.2.5. Присвоение значений параметрам компонентов	100
4.2.6. Группирование цепей в классы	104
4.3. Компиляция проекта.....	107

ГЛАВА 5. ПРОЕКТИРОВАНИЕ ПЕЧАТНОЙ ПЛАТЫ109

5.1. Передача схемы в среду проектирования печатной платы.....	109
5.2. Группирование объектов в классы.....	111

5.3. Настройка правил проектирования	113
5.4. Приоритеты правил	117
5.5. Порядок применения правил	118
5.5.1. Наложение правил на объекты	119
5.5.2. Обзор объектов, подпадающих под правило	121
5.6. Размещение компонентов на печатной плате	122
5.6.1. Ручное размещение	122
5.6.2. Автоматическое размещение	126
ГЛАВА 6. ТРАССИРОВКА ПЕЧАТНОГО МОНТАЖА	131
6.1. Стандартная интерактивная трассировка	136
6.2. Разрешение конфликтов	138
6.3. Управление шириной печатного проводника	140
6.4. Разводка групповых трасс	143
6.5. Редактирование готовой разводки	145
6.6. "Тонкая" интерактивная трассировка	148
6.7. Трассировка дифференциальных пар	152
6.7.1. Объявление дифференциальной пары	155
Дифференциальные пары в редакторе электрической схемы	155
Дифференциальные пары в редакторе печатной платы	156
6.7.2. Правила трассировки дифференциальных пар	159
6.7.3. Трассировка дифференциальной пары	161
6.8. Автоматическая трассировка печатного монтажа	162
6.8.1. Стратегии и проходы автотрассировки	163
6.8.2. Правила и отчеты автотрассировщика	167
6.8.3. Автотрассировка	168
6.8.4. Отчет автотрассировщика	173
6.9. Верификация РСВ-проекта	174
ГЛАВА 7. ПРОЕКТ ПЛИС	179
7.1. Формирование нового проекта ПЛИС	184
7.2. Библиотечное обеспечение проектов ПЛИС	187
7.3. Формирование логической структуры ПЛИС	194
7.3.1. Конфигурация рабочего пространства схемного редактора	194
7.3.2. Вызов библиотечных компонентов	195
7.3.3. Линии электрической связи	197
7.4. Выбор путей реализации проекта	201
7.4.1. Создание нового файла привязок	202
7.4.2. Выбор микросхемы ПЛИС	203

7.4.3. Привязка портов.....	206
7.4.4. Формирование конфигурации проекта	208
7.5. Выполнение проекта в оболочке Devices View	210
7.5.1. Работа в режиме Live	211
7.5.2. Работа в режиме <i>Not Live</i>	216
7.6. Программирование ПЛИС	217
7.6.1. Компиляция	218
7.6.2. Синтез	218
7.6.3. Построение	219
7.6.4. Загрузка программы коммутации ПЛИС	221
7.7. Встроенные инструменты контроля и отладки.....	223

Глава 8. Объединение проекта ПЛИС

С проектом печатной платы.....227

8.1. Мастер объединения проектов	227
8.1.1. Выбор конфигурации проекта	228
8.1.2. Привязка портов логической схемы к выводам ПЛИС	229
8.1.3. Объявление имени РСВ-проекта.....	231
8.1.4. Генерация схемного листа проекта	231
8.2. Синхронизация FPGA- и РСВ-проектов.....	235
8.3. Поддержка связности при модификации проектов.....	238

Глава 9. Схемотехническое моделирование241

9.1. Моделирование аналоговых функциональных узлов	242
9.1.1. Язык моделирования	242
9.1.2. Модели компонентов.....	242
9.1.3. Подготовка схемы к моделированию	250
9.1.4. Задание на моделирование.....	250
9.1.5. Анализ цепи по постоянному току	253
9.1.6. Анализ частотной характеристики цепи.....	258
9.1.7. Параметрический анализ.....	262
9.1.8. Представление результатов моделирования	266
9.1.9. Анализ переходных процессов	273
9.1.10. Анализ функции передачи по постоянному току	281
9.1.11. Анализ нулей и полюсов передаточной функции.....	282
9.1.12. Моделирование случайных разбросов параметров компонентов	285
9.1.13. Моделирование шумовых характеристик	293
9.1.14. Моделирование вариаций температуры	295

9.2. Моделирование цифровых функциональных узлов	297
9.2.1. Язык описания моделей цифровых компонентов.....	298
9.2.2. Подключение моделей цифровых компонентов.....	305
9.2.3. Подготовка библиотеки компонентов	314
9.2.4. Подготовка схемы и моделирование	315
Синхронный формирователь одиночного импульса	316
Четырехразрядный реверсивный двоично-десятичный счетчик с дешифратором	321

ГЛАВА 10. МОДЕЛИРОВАНИЕ ПАРАЗИТНЫХ ЭФФЕКТОВ В ПЕЧАТНОМ МОНТАЖЕ.....325

10.1. Предварительные замечания	326
10.2. Присоединение IBIS-моделей к компонентам проекта.....	329
10.3. Выполнение анализа Signal Integrity	336
10.3.1. Моделирование отражений.....	343
10.3.2. Моделирование взаимных наводок.....	345

ГЛАВА 11. СВЯЗ ALTIUM DESIGNER С ВНЕШНИМИ БАЗАМИ ДАННЫХ.....349

11.1. Прямая связь с библиотеками корпоративной базы данных.....	350
11.1.1. Предварительные замечания	350
11.1.2. Создание файла связи с базой данных Database Link file	354
Поиск компонента по единственному признаку	357
Поиск компонентов по совокупности признаков	358
11.2. Связь с базой данных через файл библиотеки Database Library file	361
11.2.1. Образование DBLib-файла.....	362
11.2.2. Использование компонентов DBLib-библиотеки	367
11.2.3. Формирование базы данных из интегрированной библиотеки.....	374
11.2.4. Преобразование DBLib-структуры в интегрированную библиотеку.....	377

ГЛАВА 12. РАБОТА ALTIUM DESIGNER ПОД УПРАВЛЕНИЕМ СИСТЕМ КОНТРОЛЯ ВЕРСИЙ.....381

12.1. Активация связи с системой управления версиями	382
12.2. Управление версиями проектных документов	384
12.2.1. Активизация панели <i>Storage Manager</i>	384
12.2.2. Образование депозитария системы контроля версий	384
12.2.3. Присоединение документов к базе данных системы контроля версий	385

12.2.4. Подключение файлов к депозиторию	387
12.2.5. Работа с проектными документами	387
12.3. Библиотеки под контролем версий	389
12.3.1. Ведение библиотек	390
12.3.2. Расщепление библиотек	390
12.3.3. Образование библиотеки SVN Database Library	394
12.3.4. Редактирование таблиц базы данных из оболочки SVNDBLib	396
12.3.5. Использование библиотек разработчиком проектов.....	399
ГЛАВА 13. ЭКСПОРТ РЕЗУЛЬТАТОВ ПРОЕКТА.....	403
13.1. Экспорт в AutoCAD.....	403
13.2. Экспорт в PDF-формат Adobe Acrobat	405
13.3. Послойные распечатки проекта.....	407
13.4. Формирование распечатки видов сборки узла.....	411
13.5. Экспорт данных для сверления	412
13.6. Экспорт в формате фотоплоттера Gerber	418
13.7. Экспорт в формат ODB++.....	428
ГЛАВА 14. РАЗРАБОТКА ГРАФИЧЕСКИХ КОНСТРУКТОРСКИХ ДОКУМЕНТОВ СРЕДСТВАМИ AUTOCAD.....	433
14.1. Общая характеристика графических средств AutoCAD 2002.....	433
14.2. Нанесение размеров и предельных отклонений на чертеже	436
14.3. Создание и управление свойствами плавающих видовых окон в AutoCAD 2002.....	440
14.3.1. Создание новых плавающих окон.....	441
14.3.2. Масштабирование изображений в плавающих окнах.....	445
14.3.3. Управление видимостью объектов в плавающих окнах пространства листа	446
14.4. Подключение форматов к чертежу AutoCAD.....	449
14.5. Заполнение реквизитов документа	453
14.6. Формирование чертежей функционального узла	455
СПИСОК ЛИТЕРАТУРЫ	460
ГЛОССАРИЙ.....	461
ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ.....	468

Введение

В предлагаемой книге будут рассмотрены основные приемы проектирования радиоэлектронных функциональных узлов в среде Altium Designer 6.

Программные продукты САПР австралийской фирмы Protel Technologies, предназначенные для проведения сквозного проектирования функциональных узлов РЭС, известны с 1990-х гг. На рынок последовательно выходили версии Tango PRO, Protel 99 SE (1999), Protel DXP (2002–2004). Права на продукты Protel в 2005 г. перешли к фирме Altium Ltd., и в настоящее время версия Protel DXP [1], получившая определенное развитие, вышла на рынок в 2006 г. под именем Altium Designer 6.0. Это современная мощная "сквозная" САПР, превосходящая многие другие по эффективности.

Ее отличительная особенность — возможность проектировать функциональные узлы на микросхемах программируемой логики (ПЛИС, или FPGA в англоязычной лексике). Для этого используется как обычный схемный ввод исходных данных, так и задание логики на языке описания схем VHDL. Это позволяет модернизировать проекты, ранее разработанные на микросхемах малой и средней степени интеграции, выполнив их на ПЛИС, и создавать новые решения на ПЛИС непосредственно в среде Altium Designer. Допускается также импорт входных данных из других САПР, например P-CAD, OrCAD и PADS.

Фирма Altium Ltd, обладающая также правами на P-CAD, в 2006 г. приняла решение прекратить дальнейшую поддержку линии продуктов P-CAD, заканчивая версией 2006, полностью перейти на Altium Designer 6 и в дальнейшем развивать только этот продукт. Фирма при этом заверяет пользователей, обладающих лицензией на P-CAD, что благодаря развитым средствам конверсии форматов данных они не испытают неудобств и не потерпят ущерба при переходе с P-CAD на Altium Designer.

Показатели популярности различных САПР в нашей стране (см. таблицу), определенные в 2006 г. по результатам опроса 330 отечественных специалистов¹, занятых проектированием печатных плат, показывают, что продукты

¹ По данным с интернет-сайта www.pcad.ru.

семейства Protel пользуются определенной популярностью, но Altium Designer на момент опроса не применял ни один из 330 участников.

В последующие годы это соотношение меняется в пользу Altium Designer. Продукт интенсивно обновляется: на рынок последовательно поступили версии 6.3, 6.7 и 6.9, затем версии стали датироваться по моменту выпуска: Winter 08 и Summer 09 — соответственно, зимой 2008 и летом 2009 гг.

Наше изложение основывается на версии Altium Designer 6.9. Изменения, вошедшие в более поздние версии, не касаются состава и базового набора функций Altium Designer, поэтому читатели книги при первом знакомстве с новыми версиями пакета не будут испытывать затруднений.

Данных о приверженности зарубежных специалистов тому или иному семейству продуктов САПР автору найти не удалось, однако дискуссия на форуме сайта производителя **www.altium.com** показывает определенную озабоченность разработчиков перспективой отказа от продукта P-CAD.

Показатели популярности различных САПР у российских конструкторов

САПР	Число положительных ответов	Процент пользователей
P-CAD 2000...2006	165	47,01
OrCAD (Cadence)	41	11,68
Protel DXP (Altium)	28	7,98
Protel 99 SE	17	4,84

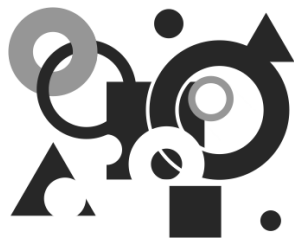
В дальнейшем можно ожидать, что возможность выполнения проектов с использованием интегральных микросхем программируемой логики (ПЛИС, они же FPGA) в интегрированной среде Altium Designer даст этому программному комплексу преимущества перед конкурентами и изменит приведенное соотношение в пользу продуктов Altium Designer.

При всех своих достоинствах Altium Designer, однако, как и другие "электронные" САПР, не дает возможности провести полный цикл сквозного проектирования изделия, включающий этап выпуска конструкторской документации по ЕСКД. Конечным результатом работы "электронных" САПР является интегральный графический образ печатной платы и файлы управляющей информации для автоматизированного производства печатных

плат. Этого достаточно для "бездокументного" производства, но принятые на практике системы ведения рабочей конструкторской документации предполагают кроме этого выпуск комплекта документов на "твердом" (бумажном) или "электронном" носителе — в виде файлов на компьютере. Проведенные в 2006 г. обновления ЕСКД устанавливают равноправный статус электронных и бумажных конструкторских документов, не отменяя при этом последние.

Для выполнения конструкторского этапа сквозного проектирования, включая выпуск документов по ЕСКД, приходится пользоваться связками САПР, одни из которых выполняют разработку "электронной" составляющей проекта радиоэлектронного функционального узла (РЭФУ), а другие — "машиностроительную" часть проекта.

В данной книге не рассматривается еще один этап проектирования — технологический, выполнение которого представляет собой самостоятельную сложную задачу и обеспечивается специализированными САПР, требующими отдельного описания.



Состав и базовые концепции Altium Designer

1.1. Основные функции Altium Designer

Рассмотрим состав интегрированной САПР радиоэлектронных функциональных узлов Altium Designer и охарактеризуем основные функции входящих в нее компонентов¹.

- ❑ Интегрирующая оболочка DXP (аббревиатура от Design eXPlorer), организующая среду проектирования и объединяющая составные части выполняемой разработки в проект.
- ❑ Базовые средства проектирования (Foundation). К ним относятся компоненты Altium Designer, обеспечивающие:
 - формирование библиотек компонентной базы;
 - просмотр и редактирование электрической схемы;
 - схемотехническое моделирование (средства PSpice и XSPICE);
 - моделирование логики на основе VHDL-описаний;
 - анализ (на этапе разработки электрической схемы) расщеплений фронтов сигналов быстродействующей логики за счет отражений волны на концах печатных проводников (средства Signal Integrity);

¹ Документ HELP-системы Altium Designer: Project Essentials. Application Note AP0129 (v. 1.1) Nov. 15, 2005.

- средства подготовки монтажного поля печатной платы, определения структуры слоев платы, правил выполнения печатного монтажа, импорта описания схемы (Netlist) из схемного редактора;
- просмотр и распечатку проекта печатной платы;
- импорт и просмотр файлов механической обработки и фотошаблонов (средства CAM File Viewer).

□ Средства проектирования печатной платы (Board Implementation):

- графический редактор печатной платы PCB Layout — размещение и редактирование объектов на печатной плате; использование библиотек компонентов; ручное, интерактивное и авторазмещение; интерактивная трассировка, трассировка дифференциальных пар и др.;
- автотрассировщик Situs — автотрассировка печатной платы;
- Signal Integrity — анализ паразитных эффектов (расщепления сигналов и наводок в печатном монтаже) на стадии проектирования печати;
- средства формирования управляющей информации для производственного оборудования — файлов в формате фотоплоттера Gerber и в обменном формате ODB++ файлы данных для сверления (NC Drill);
- редактор САМ-файлов — импорт и редактирование фотошаблонов, данных сверления и фрезерной обработки, экспорт управляющей информации;
- средства разработки аппаратной части JTAG-интерфейса программирования ПЛИС.

□ Средства проектирования устройств со встроенным интеллектом (Embedded Intelligence Implementation) включают:

- библиотеки ориентированных на реализацию в ПЛИС базовых логических элементов, генераторов, логических анализаторов, интерфейсных адаптеров и др.;
- средства реализации в ПЛИС процессорного ядра микроконтроллеров и оболочки дискретных процессоров на основе библиотеки функциональных аналогов ряда распространенных микроконтроллеров и моделей процессорного ядра;
- смешанные средства синтеза и моделирования логики ПЛИС — на основе схемного ввода, описаний логики VHDL или Verilog HDL;
- средства программно-аппаратной реализации JTAG-интерфейса программирования ПЛИС.

Третья группа средств отличает Altium Designer от большинства распространенных универсальных интегрированных САПР радиоэлектронных функциональных узлов.

Следует отметить, что Altium Designer не в состоянии выполнить полный цикл синтеза логики ПЛИС, включая упаковку логики в выбранную микросхему ПЛИС (компиляцию), формирование данных для программирования встроенной памяти и "прошивки" ПЛИС. Функции Altium Designer позволяют синтезировать и моделировать логику проекта, в результате чего формируются выходные данные в обменном формате EDIF. Эти данные передаются в специализированные САПР производителей компонентной базы ПЛИС (Vendor Tools) фирм Altera, Actel, Lattice, Sharp, Xilinx — в зависимости от выбранного для реализации проекта семейства ПЛИС. Результат работы этих специализированных САПР — информация о коммутации логических сигналов проекта на выводы выбранной ПЛИС и данные для программирования ПЛИС в аппаратуре пользователя или на программаторе. Эти данные возвращаются в Altium Designer, где выполняется окончательная проработка проекта — включение ПЛИС в состав полной электрической принципиальной схемы, размещение компонентов на монтажном поле и трассировка печатного монтажа.

1.2. Проект Altium Designer

Базовая концепция формирования данных, вырабатываемых в Altium Designer и относящихся к одному объекту того или иного вида, — проект. Он представляет собой текстовый файл-оболочку, объединяющий иерархически организованный набор документов, предполагающих воплощение их в единственном объекте.

1.2.1. Виды проектов Altium Designer

В Altium Designer поддерживается несколько видов проекта¹.

- Проект печатной платы (PCB Project). В графическом схемном редакторе на основе библиотек компонентной базы строят электрическую принципиальную схему и проводят схемотехническое моделирование. Описание

¹ Документ HELP-системы Altium Designer: Project Essentials. Application Note AP0129 (v. 1.1) Nov. 15, 2005.

схемы передается в графический редактор печатной платы, в котором строят контур печатной платы, задают число и свойства слоев печати, правила выполнения трассировки, выполняют импорт описания схемы, размещение компонентов на монтажном пространстве, трассировку печатного монтажа, формируют стандартные файлы выходных данных для изготовления печатной платы, сборки и монтажа функционального узла. Документы проекта объединяются оболочкой с именем файла *.PrjPcb.

- Проект ПЛИС (FPGA Project). Имя файла оболочки проекта — *.PrjFpg. Разработку начинают с формирования схемного или HDL-описания логики проектируемого устройства. В проект вводят файлы, в которых содержатся ограничения, связанные с выбором семейства и типа ПЛИС, определяют коммутацию входов и выходов логических сигналов на выводы выбранной ПЛИС, задают требования к рабочей частоте, подключению внешних тактовых сигналов и т. п. Заданная логика компилируется в комбинации логических ячеек нижнего уровня внутренней структуры ПЛИС. Результаты логического синтеза выдаются в виде описания в конструкциях языка обменного формата EDIF. После этого данные EDIF-формата передаются в специализированную САПР производителя выбранного семейства ПЛИС (Vendor Tools), в которой программируется коммутация логических ячеек и блоков внутренней структуры ПЛИС и вывод логических сигналов на физические контакты выбранной микросхемы. Вырабатываются управляющие файлы для программирования ПЛИС в аппаратуре пользователя или на специальном программаторе. В результате микросхема ПЛИС может быть запрограммирована и протестирована на стенде.

На этом выполнение собственно проекта ПЛИС завершается. Однако запрограммированная микросхема не может "висеть в воздухе": в любой аппаратуре пользователя ПЛИС окружена целым рядом компонентов — резисторами, конденсаторами, микросхемами более низкой степени интеграции или еще несколькими ПЛИС, электрическими соединителями и т. п. Поэтому завершающая стадия реализации такого проекта — исполнение всего устройства в виде ячейки (модуля) на печатной плате, т. е. выполнение проекта печатной платы, на которой ПЛИС выступает как обычный компонент, наравне с остальными.

- Интегрированная библиотека. Имя файла оболочки — *.LibPkg; имя файла библиотеки — *.IntLib. В редакторе библиотек формируют наборы схемных символов (УГО) электрорадиокомпонентов, определяют ссылки на модели компонента. В качестве моделей выступают топологическое посадочное место (ТПМ), формируемое в среде редактора библиотек,

модели для схемотехнического моделирования, модели для анализа целостности сигнала — расщепления фронтов логических сигналов в быстродействующих устройствах за счет отражений от концов печатного проводника, а также трехмерные геометрические модели компонентов. Файлы описаний моделей либо ссылки с указанием пути к ним включаются в структуру библиотечного пакета *.LibPkg. После этого выполняют компиляцию компонентов библиотечного пакета в единый файл интегрированной библиотеки.

- ❑ Встроенный проект (Embedded Project). Имя файла оболочки — *.PrgEmb. Это набор проектных документов, образующих программное приложение. Исходный текст программы и (или) подпрограмм формируется в конструкциях языка С или ассемблера, компилируется в объектные модули; подпрограммы связываются в исполняемый модуль в машинном коде, готовый для загрузки в выбранный процессор или микроконтроллер.
- ❑ Проект логического ядра, неоднократно повторяющегося как элемент логической структуры в других проектах ПЛИС (Core Project). Имя файла оболочки — *.PrjCor. Проект заканчивается формированием графического схемного символа и описания логики в формате EDIF.
- ❑ Скрипт-проект. Имя файла оболочки — *.PrjScr. Цель программирования в среде Altium Designer — модификация объектов в других открытых проектах. Для управления предназначен интерфейс программирования приложений API (DXP Application Programming Interface).

1.2.2. Создание нового проекта

Рассмотрим процедуру создания нового проекта на примере разработки печатной платы¹.

1. Щелчком на вкладке **System** (Система) в правом нижнем углу главного окна программы активизировать "всплывающее" контекстное (Pop-up) меню и указать в нем команду **Projects** (Проекты). Открывается плавающая панель **Projects**, в главном поле которой отображена иерархия ранее открытых проектов (рис. 1.1).
2. Вызвать команду главного меню **File** (Файл) | **New** (Новый) | **Project** (Проект) | **PCB Project** (Проект печатной платы).

¹ Документ HELP-системы Altium Designer: Getting Started with PCB Design. Tutorial TU0117 (v. 1.4) Nov. 28, 2005.

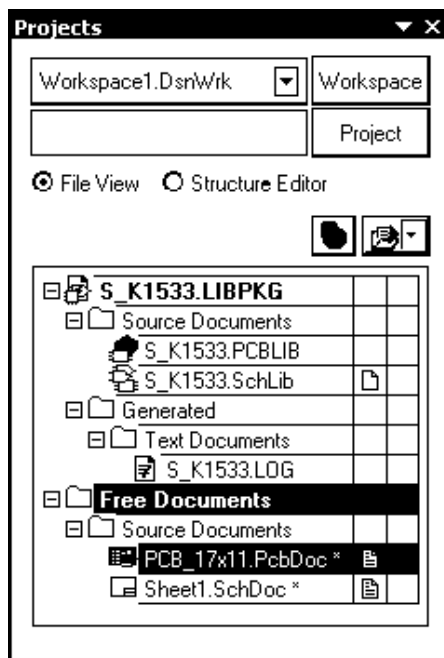


Рис. 1.1. Дерево документов панели **Projects**

- В диалоговом окне **Select Project Type** (Выбрать тип проекта) указать тип проекта — **Protel PCB** или **P-CAD PCB**. В дереве проектов в главном поле панели **Projects** появится новый узел с именем **PCB_Project1.PrjPcb**.
- Активизировать команду главного меню **File | Save As** (Сохранить как) и сохранить файл нового проекта в дисковой памяти компьютера с новым именем, например **PCB_Filter.PrjPcb**.

1.2.3. Включение документов в проект

Документы, создаваемые в Altium Designer, могут быть как связаны, так и не связаны с конкретным проектом. В последнем случае они носят статус "свободных" и так же, как связанные, отображаются в плавающей панели **Projects**, в ветви **Free Documents**.

Для включения документов в проект:

- Поставить курсор на имя проекта в плавающей панели **Projects** и щелчком правой кнопки мыши вызвать контекстное меню (рис. 1.2), в котором ак-

тивизировать команду **Add New to Project** (Добавить новое к проекту) или **Add Existing to Project** (Добавить существующее к проекту).

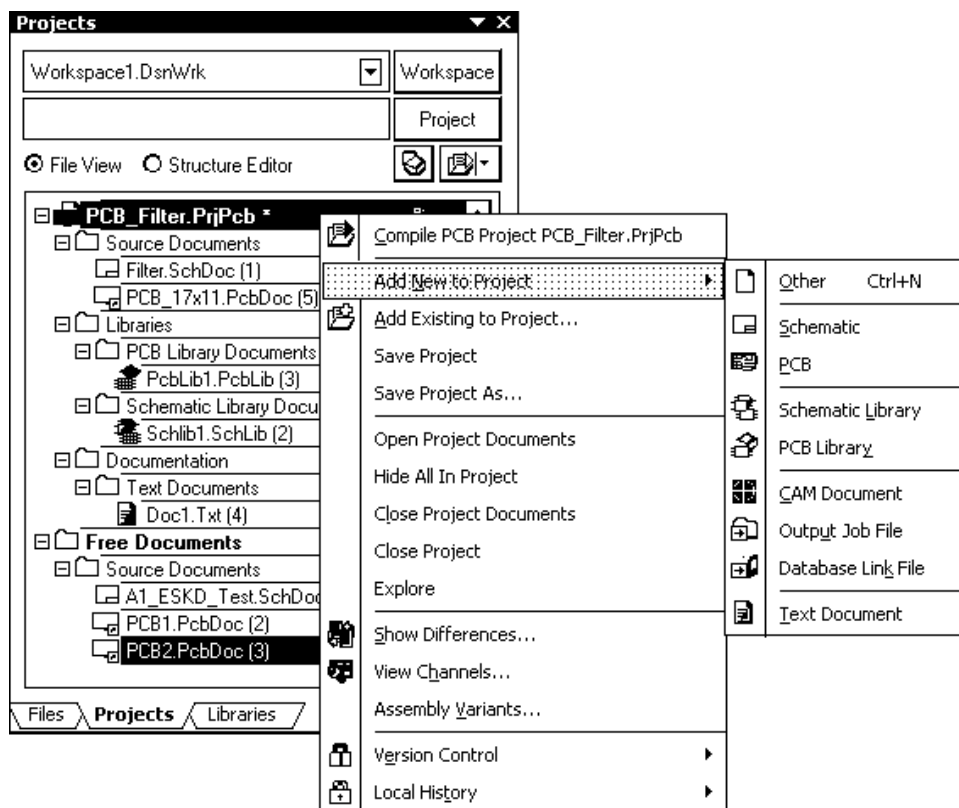


Рис. 1.2. Меню команд управления документами проекта

В первом случае открывается контекстное меню следующего, нижнего уровня с перечнем возможных типов подключаемого документа:

- **Schematic** — схемный документ;
- **PCB** — файл проекта печатной платы;
- **Schematic Library** — библиотека схемных символов;
- **PCB Library** — библиотека топологических посадочных мест;
- **CAM Document** — документ программы редактирования фотошаблонов CAMtastic;

- **Output Job File** — файл выходных данных для производства;
- **Database Link File** — файл-указатель связи с базой данных;
- **Text Document** — текстовый документ;
- **Other** — прочие документы.

При подключении существующего документа (**Add Existing to Project**) открывается стандартная процедура поиска файла. Найденный файл включается в состав документов проекта.

В обоих случаях имя файла подключенного документа включается в дерево документов проекта. При подключении графических документов (файлов схемы, печатной платы, библиотеки схемных символов или посадочных мест) одновременно с появлением имени документа в дереве проекта открывается лист соответствующего графического редактора — пустой либо занятый объектами подключенного документа.

2. Сохранить подключенные к проекту новые (пустые) документы, указывая на них по очереди курсором и активизируя каждый раз команду главного меню **File | Save As**.
3. Для исключения документа из состава проекта щелчком правой кнопки мыши на имени документа в дереве проекта вызвать контекстное меню и указать в нем команду **Remove from Project** (Исключить из проекта).

1.2.4. Работа с документами проекта

Каждый документ, подключенный к проекту, хранится в памяти компьютера под своим именем, а файл-оболочка проекта (*.PrjPcb) устанавливает связь между всеми этими документами, в результате они становятся доступными при открытии файла проекта.

Документ (как свободный, так и принадлежащий тому или иному проекту) активизируется щелчком правой кнопки мыши на имени в дереве плавающей панели **Projects**. Имя документа выделяется белым на темном фоне. Одновременно появляется доступ к контекстному меню, командами которого можно вносить изменения в данный активный документ. Связанный документ и проект, которому он принадлежит, активизируются одновременно. Если открывается свободный документ, то в дереве документов панели **Projects** становится активным узел **Free Documents** (Свободные документы).

Активизация проекта достигается тем же способом: щелчком правой кнопки мыши на имени проекта в плавающей панели **Projects**. Имя проекта выделяется белым на темном фоне.

Одновременно активизируется контекстное меню, командами которого можно вносить изменения в данный активный проект. При этом может оставаться активным документ другого, активизированного ранее, проекта.

Один и тот же документ можно подключить к неограниченному числу проектов и вызывать из каждого проекта для разработки и редактирования. При этом нужно быть осторожным: документ, отредактированный в одном проекте, при использовании в других проектах приносит с собой внесенные изменения, что не всегда приемлемо. Не помогает и простое переименование документа.

Команда **File | Save As** не решает данную задачу, поскольку при простом сохранении файла под новым именем в проекте автоматически обновляются связи, в результате чего документ с новым именем становится принадлежностью сразу всех открытых в текущий момент проектов.

Поэтому перед редактированием следует сохранить резервную копию документа командой главного меню **File | Save Copy As** (Сохранить копию как). Тогда копия документа сохраняется по указанному адресу, а исходный документ остается связанным с проектом.

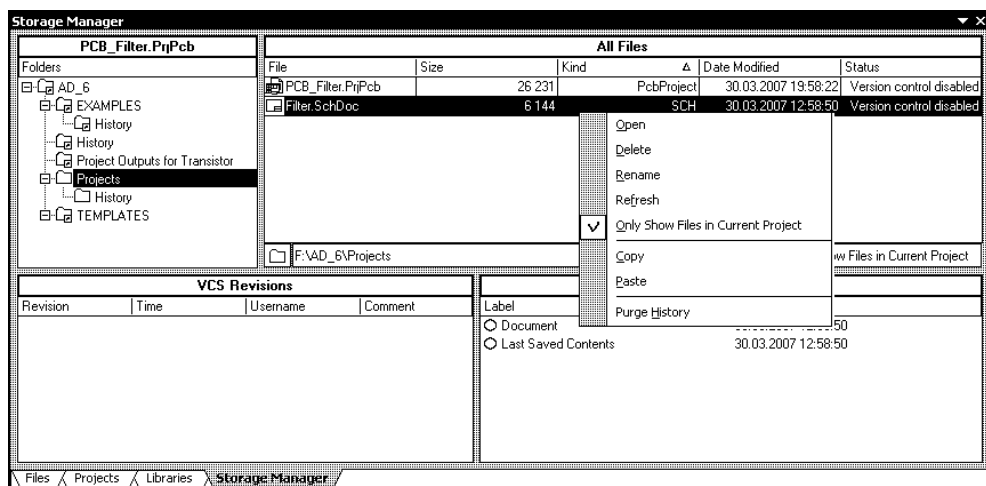
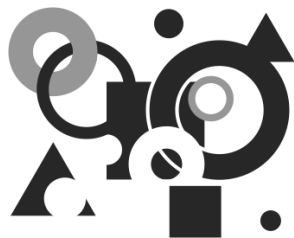


Рис. 1.3. Рабочая панель **Storage Manager**

Отметим также, что дерево документов в плавающей панели **Projects** вызывает логическую связь документов проекта и никак не отражает локализацию документов в памяти компьютера. Для выяснения физической локализации документов активного проекта следует щелчком мыши активизировать в правом нижнем углу главного окна программы команду **System | Storage Manager** (Менеджер памяти). Открывается плавающая панель **Storage Manager** (рис. 1.3), в которой отображаются пути к документам в файловой системе компьютера.

Щелчком правой кнопки мыши на имени документа вызывается контекстное меню, позволяющее выполнить с документом ряд стандартных действий, предусмотренных в ОС MS Windows: открыть документ, переименовать, обновить, удалить его, образовать копию в буфере обмена и т. д.

Дальнейшее изложение начнем с рассмотрения вопросов обеспечения проектных процедур библиотеками компонентной базы.



Библиотеки компонентной базы

2.1. Базовые концепции библиотечного обеспечения

Залог успешного выполнения проекта радиоэлектронного функционального узла — обеспеченность библиотеками компонентной базы.

Концепции библиотечного обеспечения Altium Designer¹ и САПР P-CAD 200x [2] имеют и сходства, и различия.

Как обычно, компонент представляет собой тот базовый "кирпичик", из которых составляется проект радиоэлектронного функционального узла. При выполнении разных этапов работы над проектом компонент описывается по-разному: в электрической схеме используется так называемый логический символ, при проектировании печатной платы — топологическое посадочное место (ТПМ); в схемотехническом моделировании компонент представлен SPICE- или XSPICE-моделью, при анализе целостности сигнала — IBIS-моделью своих входных и выходных цепей, при объемном проектировании функционального узла — трехмерной геометрической моделью (3D-model). Все эти представления компонента, за исключением логического символа, объединяет понятие "модель".

Наличие полной совокупности этих представлений для каждого компонента не обязательно, но в качестве отправной точки проектирования непременно требуется представление компонента его логическим символом. Это то минимальное представление, без которого невозможно начать разработку электрической принципиальной схемы. Логический символ является элементом

¹ Документ HELP-системы Altium Designer: Component, Model and Library Concepts. Article AR0104 (v. 2.0) June 07, 2006.

библиотеки схемного редактора. Он включает условно-графическое обозначение (УГО) и электрические контакты, может быть односекционным или многосекционным.

Таким образом, полное описание электронных компонентов в Altium Designer складывается из трех самостоятельных описаний:

- ❑ элемента библиотеки схемных символов *.SchLib (в отечественной терминологии — УГО);
- ❑ элемента библиотеки топологических посадочных мест *.PCBLib;
- ❑ файлов описания модели — SPICE-модели аналогового компонента, XSPICE-модели цифрового компонента и (или) IBIS-модели для анализа паразитных эффектов в печатном монтаже (целостности сигнала).

Элемент схемной библиотеки, кроме непосредственно УГО и описания электрических контактов, содержит ссылку на модель. Модель представляет собой иерархическую структуру, объединяющую ссылки на две последние составные части полного описания компонента.

Библиотечные описания символа и посадочного места могут использоваться в проектировании самостоятельно либо могут быть скомпилированы в интегрированную библиотеку. Преимущества такой библиотеки: ее компактность, возможность автоматически извлекать в проект составные части интегрированного образа в зависимости от того, ведется ли проектирование электрической схемы или печатной платы либо моделирование, а также невозможность непосредственного редактирования компонентов. Компонент можно открыть для редактирования из интегрированной библиотеки командой **Extract Sources**.

2.2. Создание новой интегрированной библиотеки

Для создания интегрированной библиотеки¹ следует:

1. Выполнить последовательность действий **File | New | Project | Integrated Library** (Интегрированная библиотека). В результате образуется библио-

¹ Документ HELP-системы Altium Designer: Creating Library Components. Tutorial TU0103 (v. 1.4) June 9, 2006.

течный пакет с именем `Integrated_Library1.LibPkg`. Этот пакет имеет статус проекта, его имя отображается в плавающей панели **Projects**.

2. Переименовать и сохранить образованный пакет в дисковой памяти. Для этого щелчком правой кнопки мыши на его имени в панели **Projects** активизировать в плавающем контекстном меню команду **Save Projects As**, ввести с клавиатуры новое имя, например `Transistor.LibPkg` (расширение имени указывать не обязательно — оно присваивается автоматически).
3. Добавить в созданный библиотечный пакет новую (пустую) библиотеку схемных символов. Для этого щелчком правой кнопки мыши на имени проекта `Transistor.LibPkg` открыть плавающее контекстное меню и указать в нем команду **Add New to Project | Schematic Library** (Библиотека схемных образов). В дереве проекта откроется новый узел — библиотека с именем `SchLib1.SchLib`, а в главном окне программы — пустой лист редактирования схемного компонента **Component_1**.
4. Вызвать в главном меню **File** команду **Save As**, переименовать и сохранить файл схемной библиотеки с именем, например `Transistor.SchLib`.
5. Активизировать в строке статуса панель-вкладку **SCH | Library**, после чего откроется пустая плавающая панель редактирования **SCH Library**.

2.2.1. Формирование схемных компонентов

Формирование нового компонента может начинаться по команде главного меню **Tools** (Инструменты) | **New Component** (Новый компонент), но в нашем случае, поскольку уже открыта панель редактирования **SCH Library** с пустым шаблоном `Component_1`, можно просто переименовать его и начать процедуру формирования логического символа схемной библиотеки, например NPN-транзистора.

1. Указать курсором имя **Component_1** и активизировать команду главного меню **Tools** | **Rename Component** (Переименовать компонент). Внести имя NPN в поле имени окна **Rename Component** и завершить переименование щелчком на кнопке **OK**.
2. Установить точку привязки графики УГО в центр окна редактирования командой главного меню **Edit** (Редактирование) | **Jump** (Прыжок) | **Origin** (В исходную точку). То же действие можно выполнить нажатием "горячих" клавиш `<J>` и `<O>`. На графическом экране окна редактирования эта точка помечается перекрестием. Вокруг нее обычно строится вся графика

УГО. В дальнейшем при вызове логических символов из библиотеки и размещении их на поле электрической схемы программа поведет символ за электрическую "горячую" точку — окончание линии электрического контакта, ближайшего к точке привязки графики.

3. Активизировать команду главного меню **Tools | Document Options** (Настройки документа) и настроить в диалоговом окне **Library Editor Workspace** (Рабочее пространство редактора библиотек) основные элементы рабочего пространства редактора библиотек (рис. 2.1):

- **Units** — систему единиц измерения (установить метрическую систему единиц);
- **Grids | Snap** — сетку захвата при построении графики;
- **Grids | Visible** — сетку, видимую на экране.

В нашем частном случае целесообразно установить шаг сетки захвата 0,5 мм, а шаг видимой сетки 5 мм.

4. Вычертить в окне графического редактирования УГО NPN-транзистора.

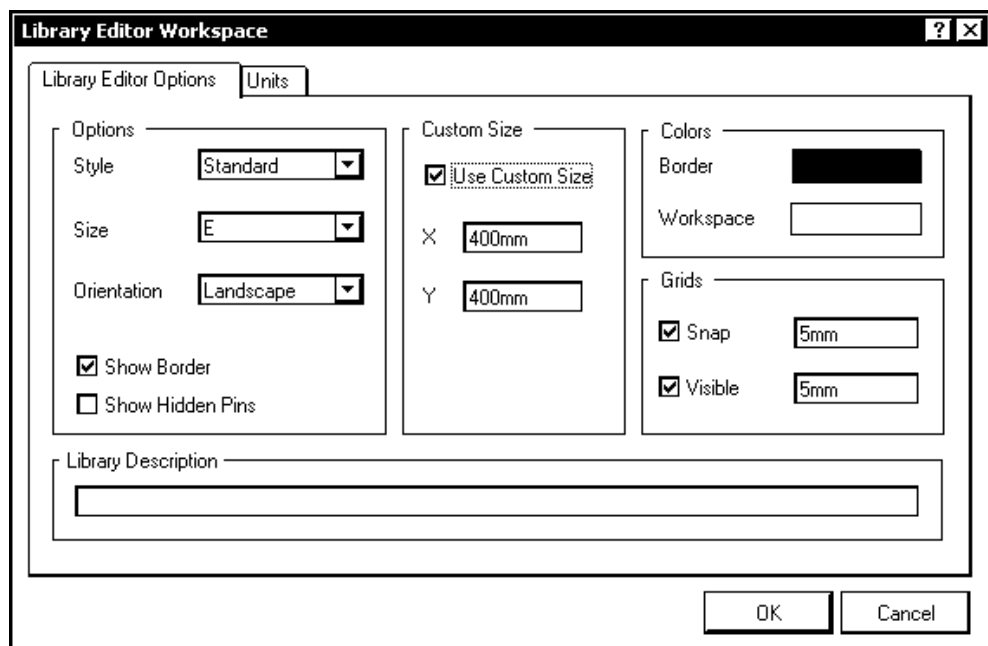


Рис. 2.1. Диалоговое окно настройки конфигурации редактора библиотек **Library Editor Workspace**

При создании УГО дискретных аналоговых компонентов по ЕСКД необходимо сформировать графику УГО и разместить электрические контакты схемного элемента (символа) в узлах сетки, единой для всех, как аналоговых, так и цифровых, компонентов. В качестве такой оптимальной сетки при проектировании электрических схем по ЕСКД целесообразно выбрать сетку с шагом 5 мм. Именно в таком модульном шаге формируются, в соответствии с ГОСТ 2.743-91 и ГОСТ 2.759-82, УГО аналоговых и цифровых интегральных микросхем. При данном подходе электрические контакты компонентов и проводники линий связи электрической принципиальной схемы оказываются в узлах единой сетки. Подобный выбор не противоречит требованиям ЕСКД к выполнению электрических схем: правила выполнения схем (ГОСТ 2.701-84, ГОСТ 2.702-75) не регламентируют расстояния между проводниками. Указывается только, что расстояние между линиями связи в электрических схемах не должно быть менее 2 мм.

Рассмотрим особенности формирования схемного символа на примере биполярного транзистора. В соответствии с ГОСТ 2.730-73 УГО биполярного транзистора формируется геометрическим построением. В окружности диаметром 12 или 14 мм под углом 60° друг к другу и симметрично относительно диаметра строятся два луча. Размер "А" между исходной точкой лучей и точкой их пересечения с окружностью составляет 9 или 11 мм и определяет все остальные размеры УГО. Линия базы транзистора строится на расстоянии 0,5А от исходной точки лучей и имеет длину, также равную А (рис. 2.2, а).

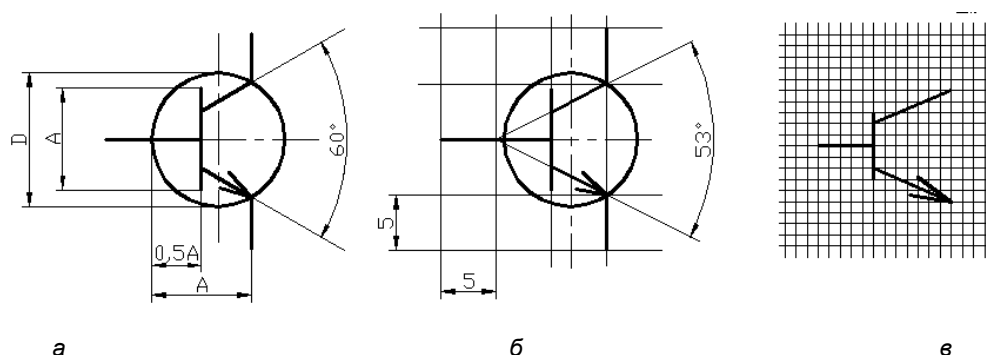


Рис. 2.2. Варианты УГО транзистора: а — по ГОСТ 2.730-73; б — в сетке с шагом 5 мм; в — в модульной сетке по ИУС № 6 1989 г.

При таком построении УГО координаты точек сопряжения элементов графики выражаются иррациональными числами. Программа, конечно, округляет

эти координаты до значения младшего разряда машинного слова, но все равно точки присоединения проводников схемы не попадают в узлы приемлемой сетки проектирования.

Изменением ИУС № 6 1989 г. к ГОСТ 2.730-73 установлен способ черчения и определения размеров УГО полупроводниковых приборов в модульной сетке (рис. 2.2, в). Это также не слишком полезно при нашем подходе (единая сетка для всех компонентов проекта). Хотя в упоминаемом "Изменении" не указано значение модульного шага, при строгом следовании приведенному в нем рисунку (см. рис. 2.2, в) затруднительно построить УГО так, чтобы оно соответствовало сформулированному требованию единой сетки для цифровых и аналоговых компонентов.

Представляется рациональным решение, изображенное на рис. 2.2, б. Линии УГО проводят через узлы 5-миллиметровой сетки, штрихи электрических контактов также имеют длину 5 или 10 мм, а окружность, изображающая корпус транзистора (согласно ГОСТ 2.730-73 с изменением ИУС № 6 1989 г. ее изображение не обязательно), смещена вправо так, что ее контур проходит через точки окончания линий эмиттера и коллектора. Угол раствора коллектора и эмиттера составляет при этом 53° . На глаз различия изображений рис. 2.2, а и б совершенно незаметны.

Порядок действий таков:

1. Выполнить команду главного меню **Place** (Разместить) | **Line** (Линию). Курсор меняет вид на перекрестие. Графический примитив Line представляет собой полилинию — все отрезки ломаной линии образуют единый объект.
2. Клавишей <Tab> активизировать функцию настройки ширины линии. Откроется диалоговое окно настройки **PolyLine** (ПолиЛиния), (рис. 2.3).

В окно вынесены следующие настройки линии:

- **Line Width** — ширина линии; курсором указывают один из вариантов:
 - ◇ **Smallest** — тончайшая, около 0,1 мм;
 - ◇ **Small** — тонкая, 0,254 мм;
 - ◇ **Medium** — средняя, около 0,5 мм;
 - ◇ **Large** — большая, около 1 мм;
- **Line Style** — стиль линии:
 - ◇ **Solid** (сплошная);
 - ◇ **Dotted** (пунктирная);
 - ◇ **Dashed** (штриховая);

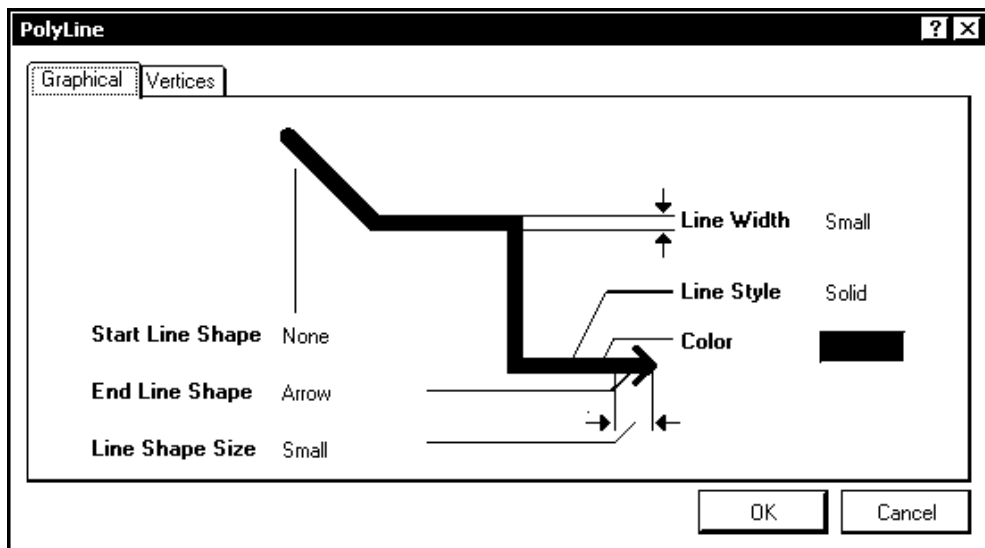


Рис. 2.3. Окно настройки параметров линии

- **Color** — цвет выбирают из палитры, открывающейся по щелчку левой кнопки мыши на цветном прямоугольнике;
- **Start Line Shape** (Фигура в начале линии) и **End Line Shape** (Фигура на конце линии) — фигуры, которыми начинается и кончается линия (стрелки, точки, квадраты и т. п.), или отсутствие фигур (**None**), а также размер этих фигур-наконечников (**Line Shape Size**).

Щелчком на кнопке **OK** завершить настройку.

3. Проложить необходимое число отрезков, образующих элементы УГО (см. рис. 2.2, б), фиксируя начало и конец каждого щелчком левой кнопки мыши. Черчение очередного отрезка завершается щелчком правой кнопки мыши или клавишей <Esc>. Перед вычерчиванием линии эмиттера клавишей <Tab> снова активизировать настройку параметров и указать стрелку на конце отрезка.
4. Выполнить команду черчения дуги **Place** (Разместить) | **Arc** (Дугу). Клавишей <Tab> вызвать диалог настройки параметров дуги (рис. 2.4).

В диалоговом окне указывают параметры:

- **Line Width** — ширина линии;
- **Color** — цвет;

- **Radius** — радиус дуги;
 - **Start Angle** — начальный угол, в абсолютных координатах;
 - **End Angle** — конечный угол, также в абсолютных координатах.
5. Указать радиус 6 мм, начальный угол 0 (ноль) и конечный угол 360°. По щелчку на кнопке **OK** вычерчивается окружность диаметром 12 мм.
 6. Переместить и зафиксировать окружность в таком положении, чтобы концы линии эмиттера и коллектора лежали на ней.

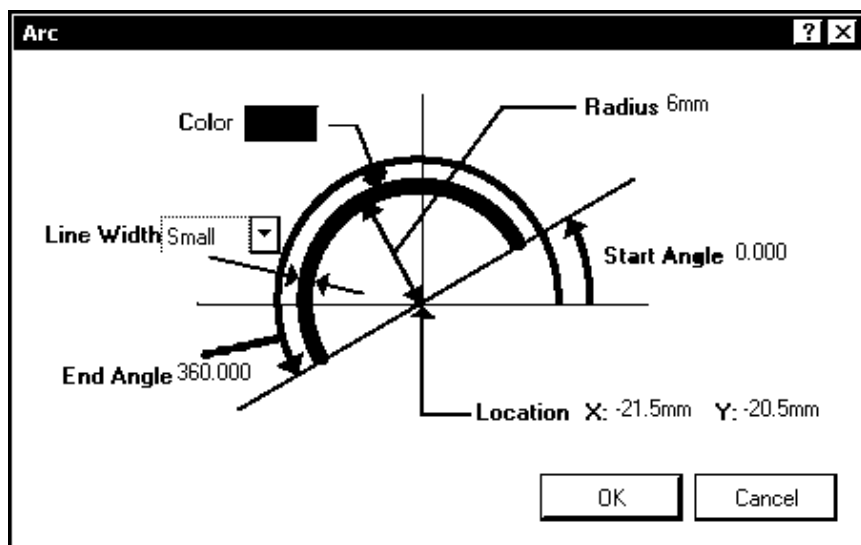


Рис. 2.4. Окно настройки параметров дуги

2.2.2. Присоединение электрических выводов

Рассмотрим порядок действий.

1. Выполнить команду главного меню **Place** (Разместить) | **Pin** (Контакт). Та же команда активизируется двукратным нажатием "горячей" клавиши <P>. На графическом экране появляется изображение электрического контакта, перемещающееся за курсором по экрану. Точка присоединения электрической связи к контакту помечена косым перекрестием. В терминах Altium Designer эта точка называется "горячим концом" (hot end). При

совмещении вывода с УГО линия вывода должна примыкать к УГО противоположным концом.

2. Перед размещением вывода клавишей <Tab> вызвать диалог редактирования его свойств. Открывается окно настройки свойств вывода **Pin Properties** (рис. 2.5).

Установить следующие свойства вывода:

- **Display Name** — логическое имя или обозначение функции вывода. В соответствии с требованиями SPICE-технологии схемотехнического моделирования последовательность ввода и обозначений выводов транзистора должна быть: коллектор – база – эмиттер. По умолчанию первому вводимому контакту дается обозначение 0 (ноль). Изменим это обозначение на 1. Обозначения следующих выводов автоматически инкрементируются;
- **Designator** — обозначение вывода в соответствии с цоколевкой, используемое при автотрассировке. По умолчанию это обозначение совпадает с обозначением Display Name и также инкрементируется. Присвоим ему также обозначение 1. Для транзистора целесообразно отключить видимость обозначений выводов, т. к. их функциональное назначение ясно из графики УГО (видимые обозначения будут у выводов микросхем);
- **Electrical Type** — электрический тип вывода. Для выводов транзистора установить тип **Passive**;
- **Description** — текстовое описание функции вывода компонента;
- **Graphical** — элементы графики вывода:
 - ◇ **Length** — длина линии вывода: задать значение 5 мм,
 - ◇ **Orientation** — ориентация линии: установить угол разворота линии вывода (против часовой стрелки, относительно положительного направления оси X). Ориентация может также меняться при установке вывода на графическом экране последовательными нажатиями клавиши <Пробел> (<Space>);
- **Symbols** — оставить без изменения признак **No Symbol** — отсутствие указателей полярности, направления передачи сигнала и т. п.

3. Щелчком на кнопке **ОК** завершить редактирование свойств вывода и установить его на УГО в окне графического редактирования компонента.

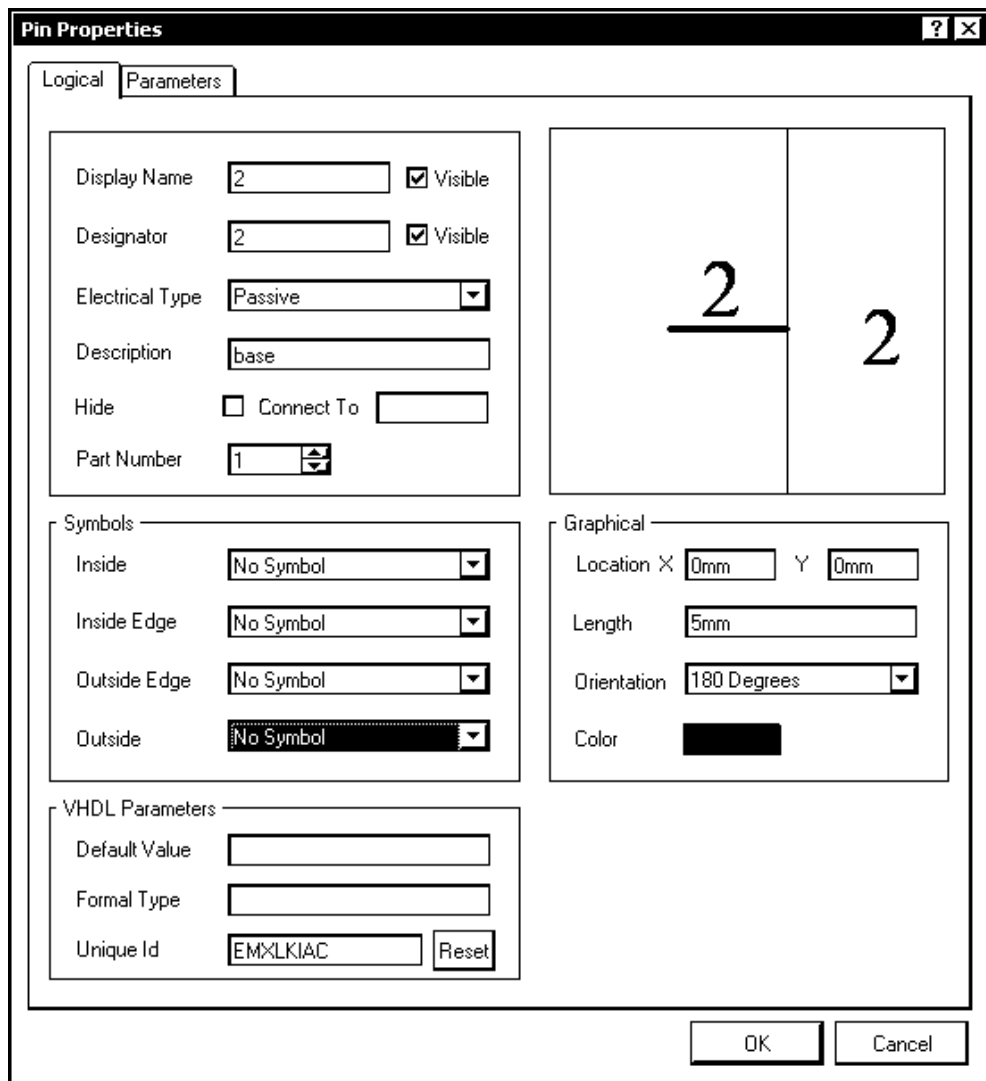
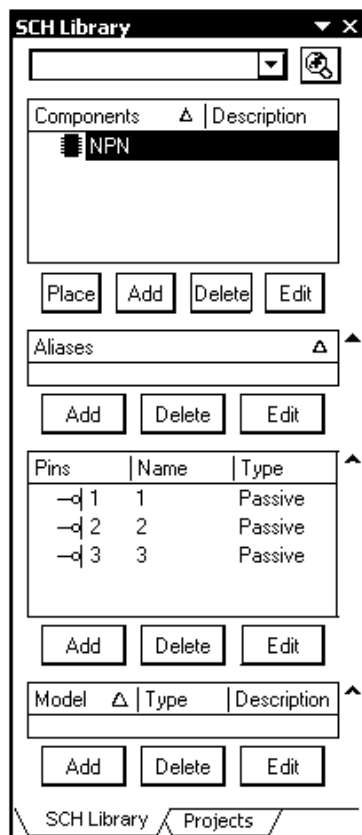
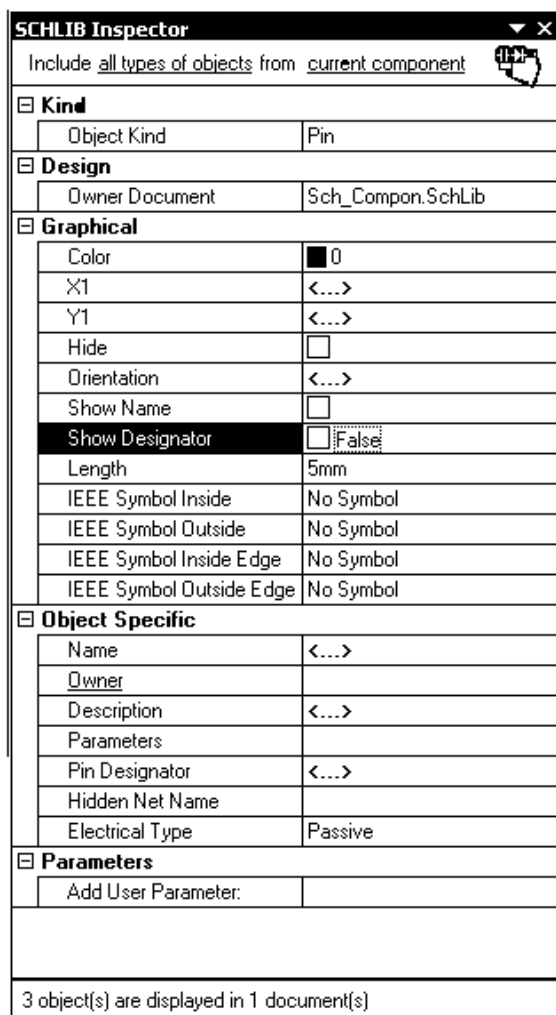


Рис. 2.5. Окно настройки свойств электрического вывода

- Повторить действия пп. 2 и 3 для следующих выводов компонента. Последовательность ввода оставшихся выводов — база, эмиттер.
- Командой главного меню **File | Save** сохранить построенный компонент в библиотеке. Результат формирования схемного символа отображается в полях плавающей панели **SCH Library** (Библиотека схемных образов) (рис. 2.6, а).



a



б

Рис. 2.6. Рабочие панели редактора схемной библиотеки:

а — панель редактирования SCH Library; б — панель "инспектора" SCHLIB Inspector

2.2.3. Формирование и редактирование схемных символов интегральных микросхем

Процедуры формирования символов цифровых и аналоговых интегральных микросхем и дискретных компонентов в принципе не отличаются. Аналогич-

но пп. 1...5 из *разд. 2.2.2* создают прямоугольный контур УГО в соответствии с требованиями ГОСТ 2.743-91 и ГОСТ 2.759-82, формируют и присоединяют к УГО выводы компонента.

Различия состоят в наличии так называемых скрытых выводов, а также меток и указателей, поясняющих функциональное назначение выводов компонента. Перечислим варианты этих обозначений, принятые в Altium Designer.

1. Обозначение электрического типа вывода (Electrical Type):

- **Input** — входной;
- **IO** — вход-выход (двунаправленный);
- **Output** — выходной;
- **Open Collector** — открытый коллектор;
- **Passive** — пассивный;
- **HiZ** — вывод с третьим, высокоимпедансным состоянием;
- **Emitter** — эмиттер;
- **Power** — силовой (выводы питания и "земли").

2. Метки, размещаемые внутри контура УГО (Inside):

- **No Symbol** — отсутствие метки;
- **Postponed Output** — выход задержанного сигнала;
- **Open Collector** — открытый коллектор;
- **HiZ** — вывод с высокоимпедансным состоянием;
- **High Current** — сильноточный выход;
- **Pulse** — импульсный сигнал;
- **Schmitt** — гистерезисный элемент (триггер Шмитта);
- **Open Collector Pull Up** — открытый коллектор, оттянутый к питанию;
- **Open Emitter** — открытый эмиттер;
- **Open Emitter Pull Up** — открытый эмиттер, оттянутый к питанию;
- **Shift Left** — сдвиг влево;
- **Open Output** — открытый выход.

3. Указатель по внутреннему краю УГО (Inside Edge):

- **No Symbol** — отсутствие указателя;
- **Clock** — вход прямого динамического тактового сигнала.

4. Указатель по внешнему краю УГО (Outside Edge):

- **No Symbol** — отсутствие указателя;
- **Dot** (колечко) — признак инверсии сигнала;
- **Active Low Input** — вход логики с активным низким уровнем логической единицы;
- **Active High Input** — вход логики с активным высоким уровнем логической единицы.

5. Указатель на выводе снаружи от контура УГО (Outside):

- **No Symbol** — отсутствие указателя;
- **Right Left Signal Flow** — передача сигнала справа налево;
- **Left Right Signal Flow** — передача сигнала слева направо;
- **Analog Signal In** — вход аналогового сигнала;
- **Digital Signal In** — вход цифрового сигнала;
- **Bidirectional Signal Flow** — двунаправленная передача сигнала;
- **Not Logic Connection** — нелогическое соединение (подключение цепей питания, "земли", пассивных времязадающих цепей и т. п.).

Для обозначения имен инверсных входов и выходов с надчеркиванием над обозначением после каждого символа пишется обратная косая черта (Backslash). Например, при вводе с клавиатуры последовательности знаков A/D\0\1\ на поле УГО появляется обозначение $\overline{AD01}$.

Altium Designer предоставляет широкие возможности индивидуального и группового редактирования свойств электрических выводов компонента.

Для индивидуального редактирования свойств нужно выделить вывод на графическом экране редактора библиотеки, щелчком правой кнопки мыши вызвать контекстное меню и в нем указать команду просмотра и редактирования свойств (**Properties**). Открывается диалоговое окно **Pin Properties** (см. рис. 2.5), в котором можно изменить свойства вывода.

Для группового редактирования следует указать курсором, удерживая нажатой клавишу <Shift>, функционально однородные выводы в поле плавающей

панели **SCH Library** (рис. 2.6, *a*) и клавишей <F11> активизировать функцию "инспектора". Открывается диалоговое окно "инспектора" схемной библиотеки **SCHLIB Inspector** (см. рис. 2.6, *б*), в котором могут быть изменены свойства группы селектированных выводов.

Рассмотрим еще один способ редактирования свойств электрических выводов.

1. В главном меню **Tools** или двойным щелчком мыши на имени компонента в плавающей панели **SCH Library** активизировать диалог редактирования свойств компонента **Component Properties**.
2. В диалоговом окне **Library Component Properties** (Свойства библиотечного компонента) кнопкой **Edit Pins** (Редактировать контакты) вызвать диалоговое окно редактора **Component Pin Editor** (рис. 2.7).
3. Указать курсором вывод и кнопкой **Edit** вызвать диалоговое окно **Pin Properties** (см. рис. 2.5). Отредактировать свойства выбранного контакта и сохранить результат.

Designator Δ	Name	Desc	Type	Owner	Show	Number	Name
1	1	collector	Passive	1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	2	base	Power	1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	3	emitter	Passive	1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Buttons: Add..., Remove..., Edit...

Buttons: OK, Cancel

Рис. 2.7. Окно настройки параметров электрических выводов

Отметим, что не все используемые в Altium Designer метки и указатели у выводов ЭРК приемлемы с точки зрения соответствия графики символов требованиям ЕСКД:

- ❑ на выводы с электрическим типом **Input**, **Output** и **IO** программа ставит специальную метку, указывающую направление сигнала. Ее наличие противоречит требованиям ГОСТ 2.743-91;
- ❑ меток внутри УГО недостаточно для формирования полного спектра изображений по ЕСКД (полный перечень этих меток в ГОСТ 2.743-91 занимает 20 страниц), в то же время часть меток и указателей, принятых в

стандартах IEEE и применяемых в Altium Designer, не соответствует по начертанию предусмотренным по ГОСТ 2.743-91;

- указатели, перечисленные ранее в пп. 4 и 5, кроме знака инверсии Dot, в изображениях по ЕСКД не применяют.

Тем не менее при разумном применении некоторых меток и указателей в сочетании с буквенно-цифровыми именами выводов (Pin Name) можно получить приемлемое изображение разнообразных аналоговых и цифровых интегральных микросхем.

2.2.4. Многосекционные компоненты

К многосекционным компонентам относятся резисторные, конденсаторные, диодные и транзисторные сборки и матрицы, электрические соединители, некоторые логические интегральные микросхемы. Схемный символ многосекционного компонента можно построить целиком, со всеми секциями в едином УГО. В этом случае создание такого символа ничем не отличается от рассмотренного ранее. Однако для формирования принципиальной электрической схемы, которая бы легко читалась и наглядно демонстрировала принцип действия функционального узла, такое изображение не всегда удобно:

- в схеме могут быть задействованы не все секции, а отображение неиспользуемых невозможно отключить, в результате схема загромождается лишними УГО;
- при едином УГО на весь компонент невозможно изобразить схему разнесенным способом, приходится тянуть к нему линии электрической связи по полю чертежа издалека, что затрудняет чтение схемы.

Поэтому целесообразно рассмотреть особенности формирования схемного символа многосекционного компонента разнесенным способом на примере логической интегральной микросхемы КР1564ЛА3, содержащей четыре двухходовых вентиля И-НЕ в одном корпусе.

1. Выполнить действия аналогично пп. 1...5 в *разд. 2.2.2*. Присвоить при этом новому библиотечному проекту имя KR1564.LibPkg, а присоединенной к нему библиотеке схемных компонентов — имя KR1564.SchLib.
2. Переименовать пустой компонент Component_1 в плавающей панели **SCH Library**, присвоив ему имя KR1564LA3, и установить точку привязки графики в центре главного окна графического редактора, аналогично пп. 1, 2 в *разд. 2.2.1*.

3. Вызвать команду главного меню **Place | Line** и вычертить квадрат УГО со стороной 15 мм.
4. Активизировать команду главного меню **Place | Pin** и вызвать клавишей <Tab> диалог настройки свойств электрического вывода (см. рис. 2.5). Назначить первому выводу:
 - обозначение **Designator** — 1, в соответствии с цоколевкой микросхемы, имя **Display Name** также задать 1;
 - электрический тип назначить **Passive**, чтобы избежать размещения меток, противоречащих требованиям ЕСКД.

Заккрыть диалог настройки и зафиксировать первый вывод в главном окне графического редактора, как показано на рис. 2.8, *а*.

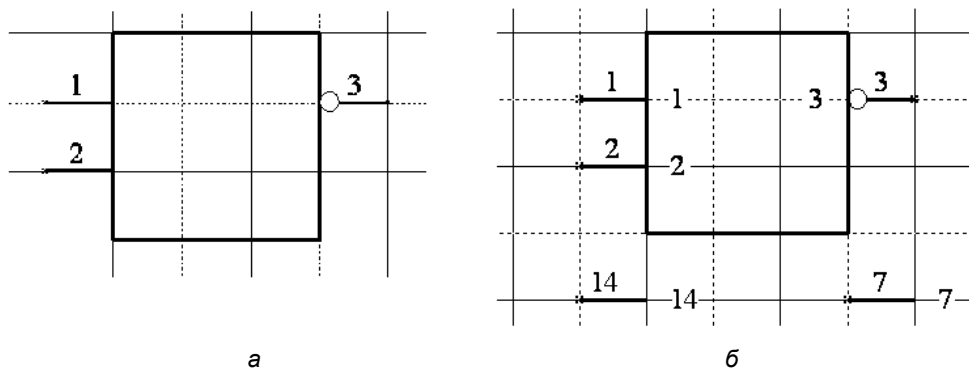


Рис. 2.8. Варианты выполнения УГО цифровых компонентов:
а — двухвходовая логическая секция;
б — логическая секция со скрытыми выводами питания и "земли"

5. Продолжить установку следующих выводов. Перед установкой третьего, выходного, снова активизировать клавишей <Tab> диалог настройки, развернуть вывод на 180° и установить значок инверсии выходного сигнала **Outside Edge** — Dot.

2.2.5. Выводы питания и "земли"

Чаще всего это выводы цифровых интегральных схем, подключаемые к цепям питания и "общему" проводу ("земле"). При традиционном ручном способе черчения электрических схем эти выводы на схеме не показывались, а

необходимость их присоединения к цепям питания и "общему" указывалась в текстовых технических требованиях на поле чертежа.

В P-CAD, Altium Designer и других САПР существуют два подхода к формированию и использованию этих выводов при выполнении электрической схемы.

При первом подходе в компонент вводят специальную секцию с выводами питания и "общего". В терминологии P-CAD такие компоненты называются гетерогенными. Цепи питания и "общего" подводят к выводам этих секций так же, как сигнальные цепи к выводам логических секций.

При втором подходе выводы питания и "общего" делают скрытыми. Они объединяются программой в цепь с назначенным именем (в P-CAD эти цепи имеют статус "глобальных"), и разработчику остается только обеспечить вывод этой цепи на контакт соединителя или лепесток для подачи питания на плату.

Рассмотрим принятую в Altium Dsigner 6 процедуру формирования скрытых выводов на примере выводов питания и "общего" нашей интегральной микросхемы KR1564LA3.

1. Выполнить команду главного меню **View (Вид) | Show Hidden Pins** (Показать скрытые выводы). По этой команде делаются видимыми все скрытые выводы компонента, а также скрытые имена и другие обозначения у видимых выводов.
2. Указать курсором в плавающей панели **SCH Library** на значок компонента KR1564LA3 (всего компонента, а не отдельной секции). В главном графическом окне редактора библиотек отобразится УГО первой секции компонента.
3. Активизировать команду главного меню **Place | Pin** и вызвать клавишей <Tab> диалоговое окно предварительного редактирования свойств вывода.
4. В диалоговом окне **Pin Properties** (см. рис. 2.5):
 - обозначениям **Display Name** и **Designator** присвоить цоколевочный номер вывода 7;
 - признаку **Electrical Type** установить значение **Power** (Силовой);
 - составить описание (**Description**): Ground_pin (Контакт "земли");
 - указать номер секции (**Part Number**) 0 (ноль): при таком обозначении выводы питания и "земли" присоединятся к каждой секции компонента;

- активизировать признак **Hide** — скрыть вывод на УГО;
- в поле **Connect to** (Соединить с) указать имя цепи **GND** — "земля".

Щелчком кнопки **ОК** завершить редактирование и зафиксировать вывод в ближайшем к контуру УГО узле сетки проектирования.

5. Повторить те же действия, установив при этом описание **Supply_pin** (Вывод питания), цоколевочный номер вывода 14 и имя цепи питания **VCC**. Конечный результат показан на рис. 2.8, б.
6. Селектируя по очереди секции компонента в плавающей панели **SCH Library**, убедиться, что у каждой секции появились выводы 7 и 14. Если это так, снять активность команды главного меню **View | Show Hidden Pins**.
7. Командой главного меню **File | Save** сохранить компонент с подключенными скрытыми выводами.

Видим, что, в отличие от P-CAD 200х, в котором скрытые выводы не включались в состав схемного символа (достаточно было присвоить им электрический тип **Power** в таблице описания контактов), в Altium Designer любой вывод может быть скрытым. Кроме того, нужно явно обозначить имя цепи, к которой подключаются скрытые выводы.

2.2.6. Редактирование свойств компонента

Порядок действий таков.

1. Двойным щелчком мыши на имени компонента в плавающей панели **SCH Library** или командой главного меню **Tools | Component Properties** (Свойства компонента) активизировать диалог редактирования свойств компонента. Откроется диалоговое окно **Library Component Properties** (рис. 2.9).
2. В поле **Default Designator** (Обозначение по умолчанию) внести буквенный префикс позиционного обозначения по ГОСТ 2.710-81. Для транзистора это VT. Добавить в строку с префиксом вопросительный знак. При составлении электрической схемы на его место в позиционное обозначение программой вносится порядковый номер компонента, в пределах группы функционально однородных.
3. В поле **Comment** вписать строку комментария, в нашем случае **NPN**.
4. В поле **Description** ввести описание компонента, по которому его можно будет отыскать при активизации поисковой системы. У нас описание может быть, например, **Transistor NPN Generic**.

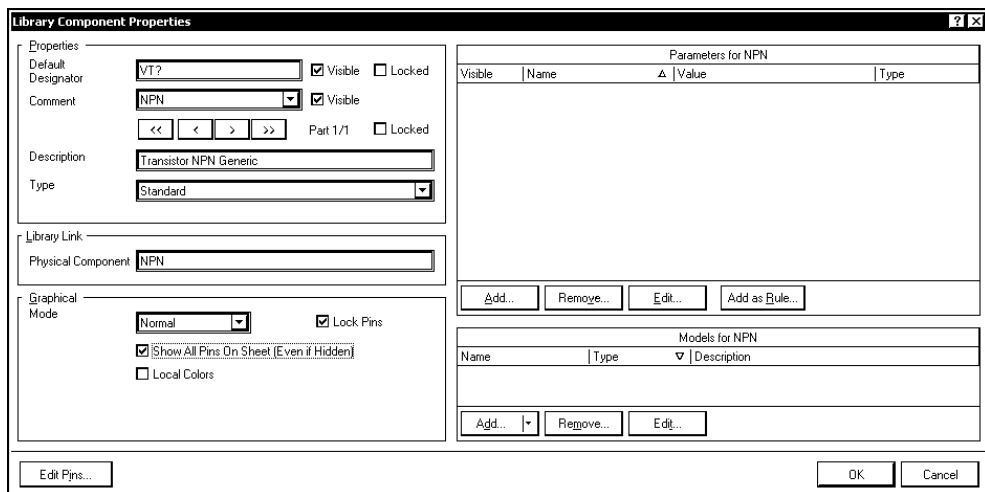


Рис. 2.9. Окно редактирования свойств
библиотечного компонента

Остальные поля в правой половине окна на рис. 2.9 остаются пока свободными. Они будут заполняться при присоединении к компоненту моделей.

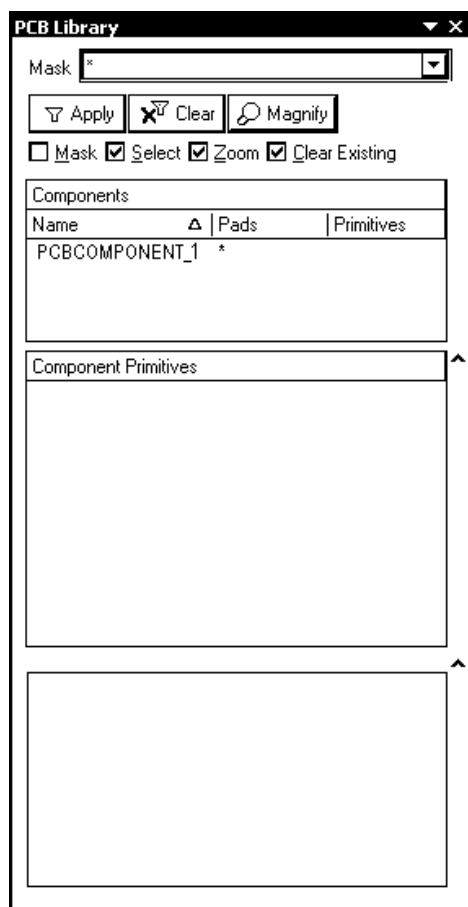
2.3. Библиотеки топологических посадочных мест

2.3.1. Создание новой библиотеки посадочных мест

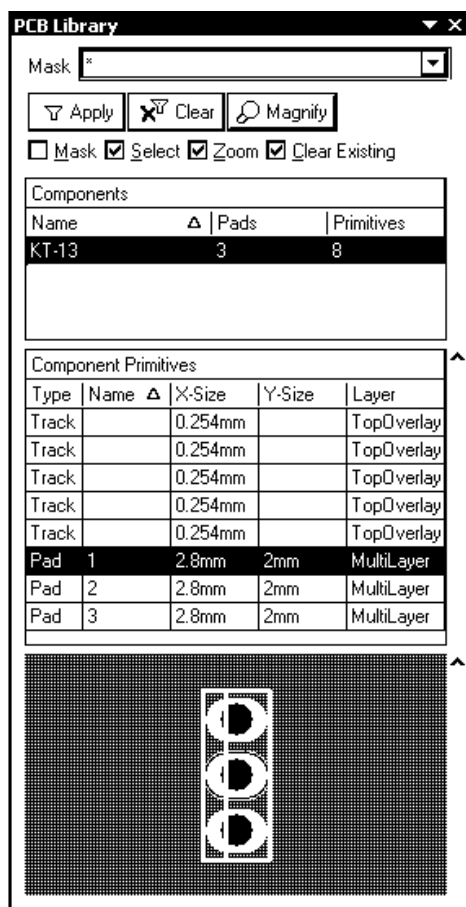
Процедуры формирования новой библиотеки топологических посадочных мест (ТПМ) компонентов и библиотеки схемных символов идентичны:

1. Активизировать команду главного меню **File | New | Library | PCB Library** (Библиотека ТПМ). В главном окне Altium Designer открывается пустое рабочее пространство редактора PCB-библиотек с именем новой библиотеки PcbLib1.PcbLib. Одновременно имя библиотеки появляется в поле проектов плавающей панели **Projects** как свободный документ.
2. Переименовать новую библиотеку: вызвать команду главного меню **File | Save As** и указать имя библиотеки, например TPM.PcbLib.

- Щелчком мыши на вкладке **PCB** в строке статуса и выбором в выпадающем меню команды **PCB Library** активизировать плавающую панель редактирования ТПМ (рис. 2.10, а). В поле **Components** этой панели представлено имя нового, пустого компонента **PCBCOMPONENT_1**.
- Вывести курсор в свободное поле главного окна и щелкнуть левой кнопкой мыши, затем несколько раз нажать клавишу <Page Up> ("горячая" клавиша увеличения масштаба изображения), пока на экране не станет видна сетка.



а



б

Рис. 2.10. Панель редактирования топологического посадочного места (ТПМ):
 а — до начала формирования ТПМ; б — после формирования ТПМ

В результате выполнения этих шагов рабочее пространство оказывается подготовленным для формирования посадочных мест компонентов. По умолчанию графический редактор имеет структуру слоев, обеспечивающую проектирование двухсторонней печатной платы:

- **Top Layer** — верхний слой печатных проводников;
- **Bottom Layer** — нижний слой печатных проводников;
- **Mechanical 1** — первый "механический" слой (для изображения радиаторов, других элементов сборки);
- **Top Overlay** — слой шелкографии;
- **Keep-Out Layer** — слой барьеров трассировки;
- **Multi-Layer** — "мультислой" — слой контактных площадок.

2.3.2. Формирование топологического посадочного места

Формирование посадочного места компонента заключается в определении его конструкции и размещении в рабочем пространстве редактора **PCB Component Editor** (Редактор компонентов печатной платы) контактных площадок для электрического подсоединения, а также в вычерчивании линий контура компонента. Линии контура располагаются обычно в слое шелкографии **Top Overlay**. Контактные площадки для монтажа компонентов со штыревыми выводами формируют в слое **Multi-Layer**, а для поверхностно монтируемых компонентов — в верхнем сигнальном слое **Top Layer**.

1. Перед началом формирования ТПМ следует задать систему единиц измерения и выполнить настройку сеток проектирования. Для этого командой главного меню **Tools | Library Options** (Настройки библиотеки) открыть диалоговое окно настроек параметров печатной платы **Board Options** (рис. 2.11).

Процедура настройки сеток подробно рассматривается в *разд. 3.2*. При формировании ТПМ отечественных компонентов следует установить метрическую систему единиц измерения и настроить три сетки:

- **Component Grid** — сетка размещения компонентов. Значение шага сетки по горизонтали и по вертикали выбирают из выпадающего списка или вводят с клавиатуры;

- **Snap Grid** — сетка захвата. Назначается для размещения графических объектов — линий, вырезов и т. п. Шаг данной сетки также выбирают из выпадающего списка или вводят с клавиатуры (значение должно находиться в дробном соотношении с шагом сетки компонентов);
- **Visible Grids** — две, лучше всего кратные, так называемые видимые сетки. Эти сетки используются только для визуального контроля действий при размещении объектов, их выравнивании и выполнении других операций графического редактирования.

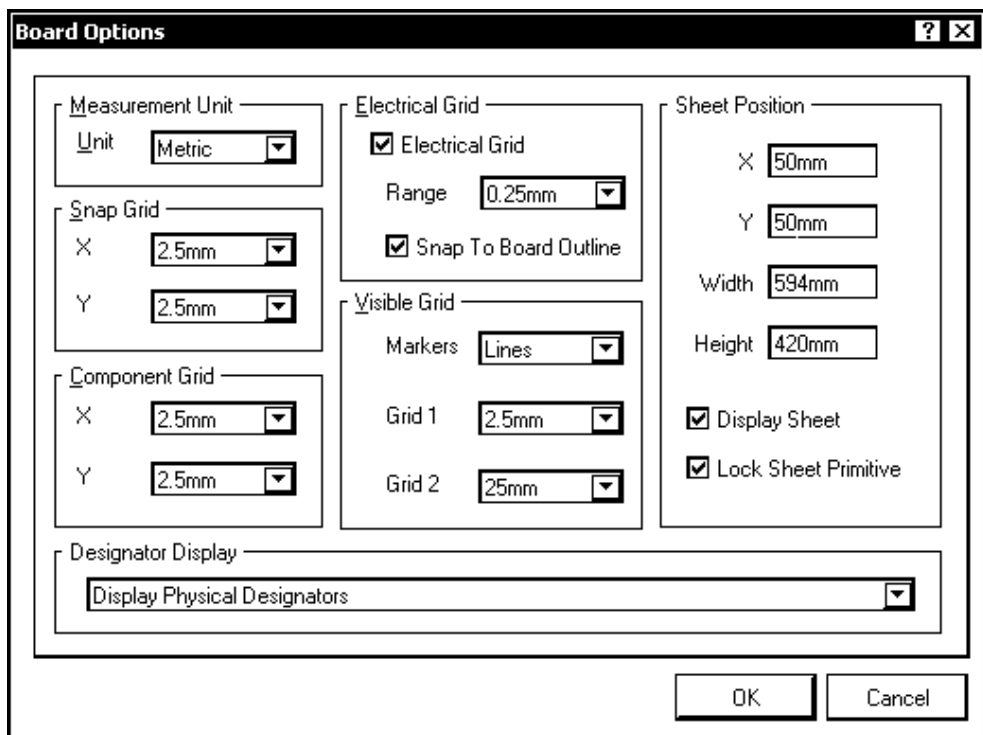


Рис. 2.11. Окно настройки конфигурации графического редактора ТПМ

2. Далее нужно переименовать открытое в панели **PCB Library** пустое посадочное место PCBCOMPONENT_1. Для определенности будем считать, что формируем посадочное место для отечественного транзистора КТ315. Выберем имя этого посадочного места по наименованию корпуса данного транзистора (КТ-13). Для присвоения нового имени следует двойным

щелчком мыши на имени **PCBCOMPONENT_1** в панели **PCB Library** активизировать окно переименования компонента, указать имя **KT-13** и ввести описание (Description) — **Footprint KT-13**.

- Затем необходимо установить точку привязки графики в центр графического листа редактора ТПМ. Для этого воспользоваться "горячими" клавишами — последовательно нажать <J> и <R>.

Отметим, что при размещении ТПМ компонентов на поле печатной платы программа захватывает ТПМ за точку привязки (Reference Point) и устанавливает их этой точкой в узлы сетки **Component Grid**. Поэтому представляется целесообразным установить такой шаг сетки **Component Grid**, чтобы в ее узлы попадали и контактные площадки компонентов. В дальнейшем, при формировании библиотек ТПМ будем размещать в точке привязки первую, "ключевую" контактную площадку (КП). Изменить положение точки привязки можно по команде главного меню **Edit | Set Reference** (Установить точку привязки).

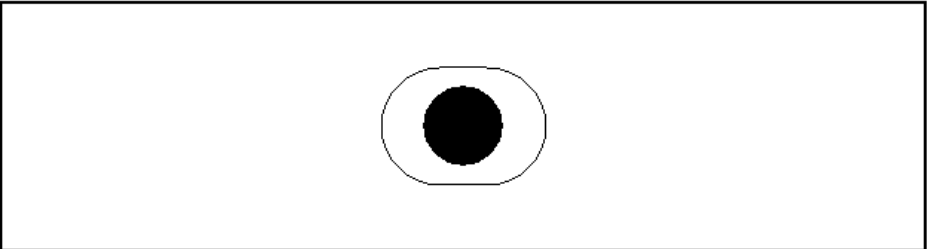
Размещение контактных площадок

Рассмотрим последовательность действий.

- Двойным нажатием "горячей" клавиши <P> или щелчком мыши в главном меню активизировать команду **Place** (Разместить) | **Pad** (Контактную площадку). Перед тем как фиксировать плавающую за курсором контактную площадку на поле графического редактора, вызвать клавишей <Tab> диалог редактирования ее параметров. Откроется диалоговое окно свойств КП, **Pad** (рис. 2.12).
- В поле **Location** (Расположение) указать координаты КП на поле графического редактора. Поскольку мы условились, что первая КП ставится в точку привязки с координатами (0, 0), не изменяем эти координаты, а по окончании редактирования свойств зафиксируем КП на месте щелчком левой кнопки мыши.
- В поле **Hole Information** (Информация об отверстии) установить вид и размер отверстия КП. Возможны следующие варианты формы отверстия:
 - Round** — круглое;
 - Square** — квадратное;
 - Slot** — щель.

Выбираем круглое отверстие диаметром (Hole Size) 1,4 мм. Такой диаметр превышает на 0,4 мм ширину ленточного вывода транзисторов в корпусе КТ-13 (требование ГОСТ 10317-79, обусловленное необходимостью свободной установки выводов компонентов в монтажные отверстия КП).

Pad [?] [X]



Top Layer / Bottom Layer / Multi-Layer

Location

X: 12.5mm

Y: 12.5mm

Rotation: 0.000

Hole Information

Hole Size: 1.4mm

☒ Round

☐ Square

☐ Slot

Properties

Designator: 1

Layer: Multi-Layer

Net: No Net

Electrical Type: Load

Testpoint: ☐ Top ☐ Bottom

Plated: ☒

Locked: ☐

Size and Shape

☒ Simple ☐ Top-Middle-Bottom ☐ Full Stack

X-Size	Y-Size	Shape
2.8mm	2.0mm	Round

Edit Full Pad Layer Definition...

Paste Mask Expansion

☒ Expansion value from rules

☐ Specify expansion value: 0mm

Solder Mask Expansions

☒ Expansion value from rules

☐ Specify expansion value: 0.102mm

☐ Force complete tenting on top

☐ Force complete tenting on bottom

OK Cancel

Рис. 2.12. Окно настройки параметров контактной площадки

4. В поле **Size and Shape** (Размер и Форма) выбрать форму и размеры площадки металлизации КП. Возможны следующие варианты формы:
 - **Round** — круглая, а при указании разных размеров по осям X , Y — овальная форма;
 - **Rectangular** — прямоугольная форма;
 - **Octagonal** — восьмиугольник.Для нашего ТПМ КТ-13 устанавливаем овальную форму с размерами **X-Size** 2,8 мм и **Y-Size** 2,0 мм.
5. В поле **Properties** (Свойства) назначить цоколевочное обозначение КП (**Designator**).

Сохраним принятый для схемного символа порядок обозначений выводов и обозначим вывод коллектора цифрой 1. Кроме цифровых, возможны буквенные (**Alphabetic**) и смешанные буквенно-цифровые обозначения выводов (**Alphanumeric**).
6. Зафиксировать сформированную КП коллектора на поле графического редактора щелчком мыши или клавишей <Enter>. После фиксации на экране первой сформированной КП за курсором начинает перемещаться следующая.
7. Зафиксировать вторую и третью КП (выводы базы и эмиттера) в ближайших узлах сетки с шагом 2,5 мм сверху и снизу от КП коллектора. Обозначения **Designator** при этом автоматически инкрементируются.
8. Завершить установку КП щелчком правой кнопки мыши или клавишей <Esc>. Сохранить полученный результат командой **File | Save**.

Черчение контуров корпуса компонента

1. Активизировать слой графического редактора **Top Overlay**, предназначенный для изображения контуров компонента в проекции, соответствующей способу установки его на плату.
2. Выполнить команду главного меню **Place | Line**. Клавишей <Tab> вызвать диалог настройки параметров линии. Установить тип линии **Solid** — сплошная, толщину линии **Small** (0,254 мм).
3. Клавишей <G> активизировать настройку сеток и выбрать активную сетку захвата **Snap Grid** с шагом 0,25 мм.

4. Вычертить прямоугольник размерами $3 \times 7,5$ мм с центром симметрии в точке расположения КП коллектора (средняя КП из трех). Затем прочертить линию фаски с отступом на 1 мм от левого края контура. Информация о геометрии контуров сформированного ТПМ и контактных площадках отображается в полях плавающей панели **PCB Library** (см. рис. 2.10, б).
5. Сохранить окончательный результат формирования ТПМ компонента командой **File | Save**.

Защитная маска и маска припойного трафарета

В структуру контактной площадки входят две маски: **Solder Mask** (дословно — припойная маска) — маска для защиты печатных проводников от воздействия влаги и перетекания припоя при монтаже и **Paste Mask** (дословно — пастовая маска) — маска, по которой выполняется трафарет для нанесения припойной пасты перед пайкой поверхностно-монтируемых (SMD) компонентов. Маски формируются на верхней и нижней сторонах платы. При настройке параметров КП есть возможность указать величину отступа краев отверстий в каждой из масок от края площадок металлизации на наружных слоях. Настройка выполняется в полях диалогового окна (рис. 2.12):

1. В поле **Paste Mask Expansion** (Расширение окон пастового трафарета) выбрать одну из опций: **Expansion value from rules** (Значение расширения из правил) либо **Specify expansion value** (Указать значение расширения). В первом случае значение отступа маски берется из правил, установленных для всего проекта в конфигурации редактора печатной платы командой **Design** (Разработка) | **Rules** (Правила). Во втором случае значение отступа назначает конструктор.
2. В поле **Solder Mask Expansion** (Расширение окон припойной маски) установить значения расширения окон в защитной маске (по умолчанию — 0,102 мм).

Здесь также можно указать полное перекрытие защитной маской площадок металлизации КП на наружных слоях — опции **Force complete tenting on top** и **force complete tenting on bottom**. Включение или отключение видимости масок управляется из диалогового окна, активизируемого командой главного меню **Tools | Layers & Colors** (Слои и Цвета). Для включения видимости каждой из четырех масок активизируется опция **Show** (Пока-

зять). В нижней части главного окна программы при этом становится активной вкладка с именем соответствующего слоя.

3. Командой главного меню **File | Save As** сохранить библиотеку TPM.PcbLib с созданным ТПМ на жестком диске компьютера.
4. Для включения сформированной РСВ-библиотеки с единственным пока компонентом в дерево библиотечного проекта Transistor.LibPkg щелчком правой кнопки мыши на имени проекта вызвать плавающее контекстное меню и активизировать в нем команду **Add Existing to Project**. Откроется стандартный диалог поиска библиотечного файла. Кнопкой **Open** (или **Открыть**, в зависимости от локализации ОС) найденный библиотечный файл включают в дерево проекта.

2.4. Присоединение моделей к схемному компоненту

Присоединение моделей делает компонент пригодным к применению в сквозном проектировании функциональных узлов. При присоединении модели в среде редактора библиотеки схемных элементов (**Schematic Library Editor**) она оказывается связанной с компонентом, но ее данные не включаются в состав схемного компонента. Это означает, что связываемые модели должны быть доступны при формировании библиотек и при помещении компонента на лист схемы. Поиск моделей для подключения к компоненту происходит в следующей очередности:

1. В библиотеках, включенных в дерево текущего проекта.
2. В РСВ-библиотеках (но не интегрированных библиотеках), входящих в список библиотек, включенных в рабочую среду текущего проекта функционального узла.
3. В любых библиотеках, путь к которым определен командой главного меню **Project | Project Options** (Настройки проекта).

2.4.1. Подключение модели посадочного места

Будем считать, что топологическое посадочное место (ТПМ) для разрабатываемого нами компонента схемной библиотеки — NPN-транзистора КТ315 — сформировано и сохранено в РСВ-библиотеке TPM.PcbLib под именем КТ-13 (название корпуса транзистора КТ315).

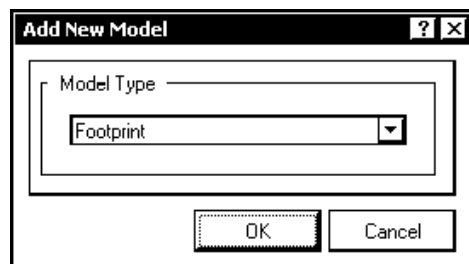


Рис. 2.13. Инициализация присоединения модели к схемному компоненту

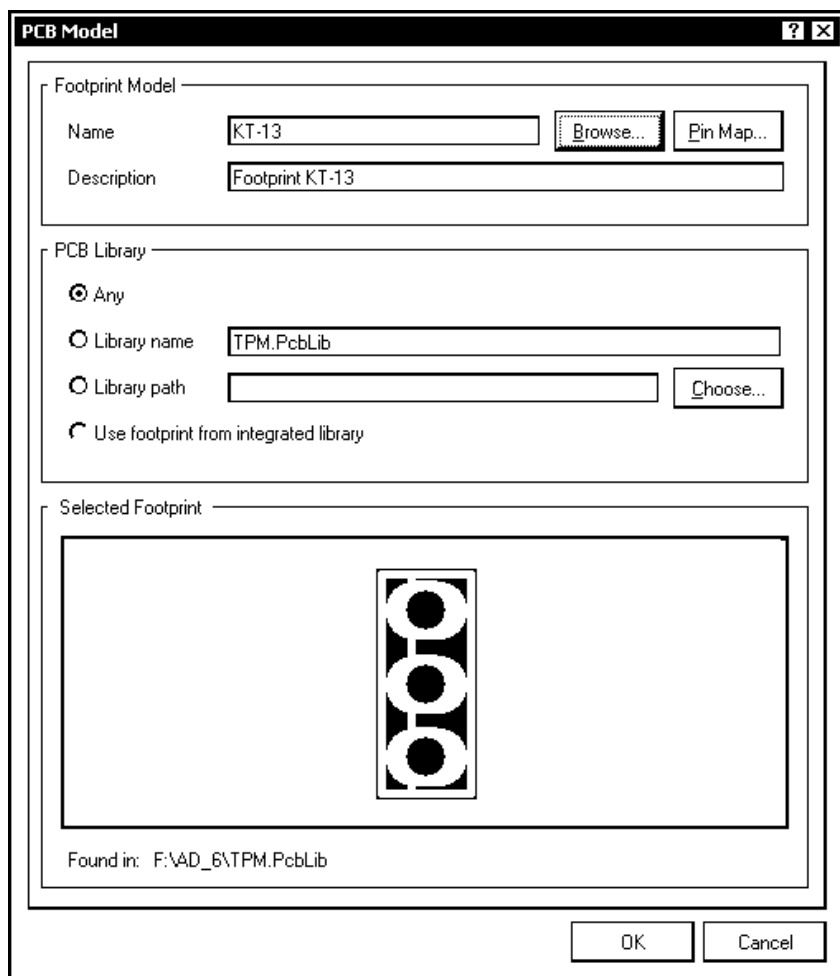


Рис. 2.14. Окно присоединения ТПМ

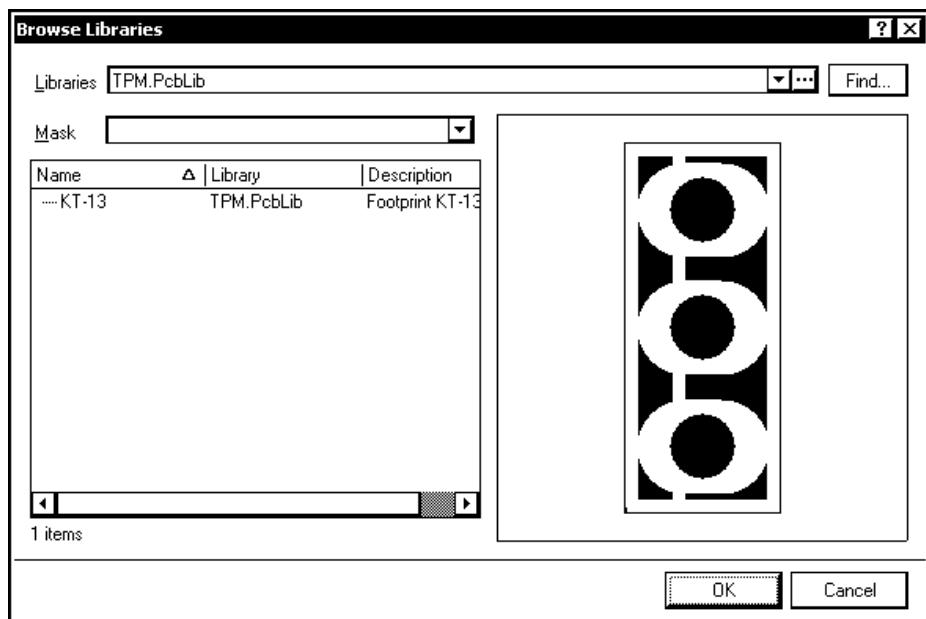


Рис. 2.15. Окно просмотра библиотек ТПМ

- Щелчком мыши на кнопке **Add** (Добавить) в поле моделей плавающей панели **SCH Library** (см. рис. 2.6, а) активизировать процедуру присоединения модели. Открывается диалоговое окно **Add New Model**, в котором следует указать тип модели — **Footprint** (рис. 2.13).
- Щелчком на кнопке **OK** открыть диалоговое окно **PCB Model** (Модель компонента печатной платы) с пустыми полями (рис. 2.14).
- Кнопкой **Browse** (Просмотр) открыть поиск PCB-библиотеки, содержащей интересующую нас топологическую модель. Открывается диалоговое окно просмотра библиотек **Browse Libraries** (рис. 2.15). Поскольку библиотека TPM.PcbLib, содержащая ТПМ KT-13, включена в дерево проекта Transistor.LibPkg, поиск сразу обнаруживает его.
- В полях окна на рис. 2.15 отображается имя библиотеки, имя ТПМ и его описание, а также графическое изображение контура и контактных площадок. Кнопкой **OK** окно закрывается, а данные из его полей передаются в окно **PCB Model** (см. рис. 2.14).
- Завершить процедуру подключения ТПМ кнопкой **OK** в окне рис. 2.13. В поле моделей плавающей панели **SCH Library** отображаются сведения о подключенной модели ТПМ — его имя, тип и описание.

2.4.2. Подключение SPICE-модели

В библиотеках Altium Designer SPICE-модели компонентов обычно хранятся в составе интегрированных библиотек. Нас же интересует процедура присоединения "сторонних" моделей, специально разработанных для отечественной компонентной базы. Эти модели представляют собой отдельные текстовые файлы, составленные по правилам SPICE-технологии моделирования. Они могут храниться в пользовательских каталогах файловой системы компьютера. Файлы моделей дискретных компонентов имеют имена *.mdl, а файлы макромоделей — *.ckt. В целях упрощения процедуры поиска, а также для объединения в дальнейшем компонентов схемной библиотеки и моделей в интегрированную библиотеку, начнем процедуру подключения SPICE-моделей с включения их в состав дерева библиотечного проекта.

1. Щелчком правой кнопки мыши на имени библиотечного проекта в плавающей панели **Projects** вызвать контекстное меню и активизировать в нем команду присоединения к проекту файла SPICE-модели. Откроется стандартный диалог поиска файлов в памяти компьютера. Найденный файл KT315A.mdl будет включен в дерево библиотечного проекта Transistor.LibPkg.
2. Кнопкой **Add** в поле моделей плавающей панели **SCH Library** запустить процедуру поиска и присоединения SPICE-модели к компоненту схемной библиотеки. В диалоговом окне **Add New Model** (см. рис. 2.13) указать категорию модели **Simulation** (Моделирование). Откроется диалоговое окно поиска модели **Sim Model** (рис. 2.16).
3. В поле **Model Kind** выбрать вид модели — Transistor.
4. В поле **Model Sub-Kind** указать разновидность транзистора (подтип) — в нашем случае BJT (биполярный транзистор).
5. В поле имени модели, **Model Name**, указать имя KT315A. Если ссылка на файл модели присутствует в дереве проекта Transistor.LibPkg, то в полях-вкладках в нижней части окна рис. 2.15 отображаются сведения о подключаемой модели:
 - **Netlist Template** — строка формальных параметров;
 - **Netlist Preview** — образец текстовой строки SPICE-описания транзистора в задании на моделирование;
 - **Model File** — текст файла описания SPICE-модели.

Щелчком на кнопке **OK** закончить операцию присоединения модели.

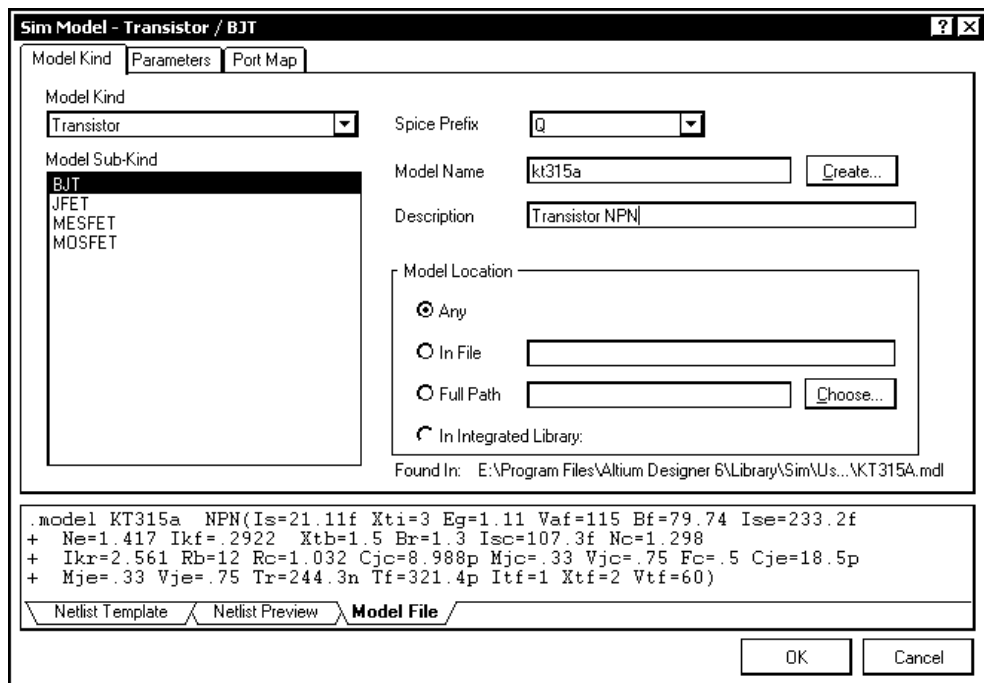


Рис. 2.16. Диалоговое окно присоединения SPICE-модели

2.4.3. Подключение моделей Signal Integrity

Для моделирования паразитных эффектов в печатном монтаже необходимо включить в описание компонентов схемной библиотеки модели входных и выходных буферов интегральных микросхем, электрических выводов других компонентов. Модель зависит от типа, схемотехники и технологии изготовления компонента. Обычно задаются значения активного входного/выходного сопротивления и входной (выходной) емкости. Модели бывают внешние, поставляемые производителями компонентов, и встроенные, параметры которых задаются непосредственно в среде редактора библиотек.

Присоединение моделей **Signal Integrity** (Целостность сигнала) осуществляется так же, как моделей TPII и SPICE-моделей.

1. Активизировать в поле моделей плавающего окна **SCH Library** и указать в диалоговом окне (см. рис. 2.13) вид модели — **Signal Integrity**. Открывается диалоговое окно **Signal Integrity Model** (рис. 2.17).

- Для подключения встроенной модели сформированного нами NPN-транзистора (его выводов) указать в поле **Model Name** имя NPN, в поле описания **Description** ввести описание Generic NPN, а в поле **Type** выбрать тип модели BJT — биполярный транзистор.
- Щелчком на кнопке **ОК** завершить настройку.

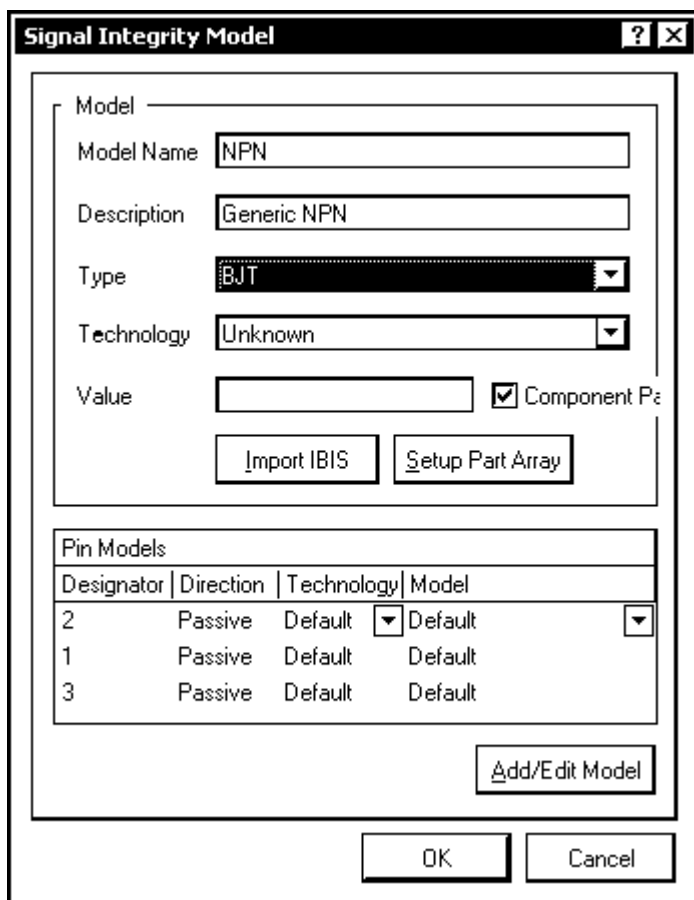


Рис. 2.17. Диалоговое окно присоединения IBIS-модели

Сведения о присоединенной модели отображаются одновременно в полях моделей плавающей панели **SCH Library** и главного окна редактора библиотек **Schematic Library Editor** (рис. 2.18).

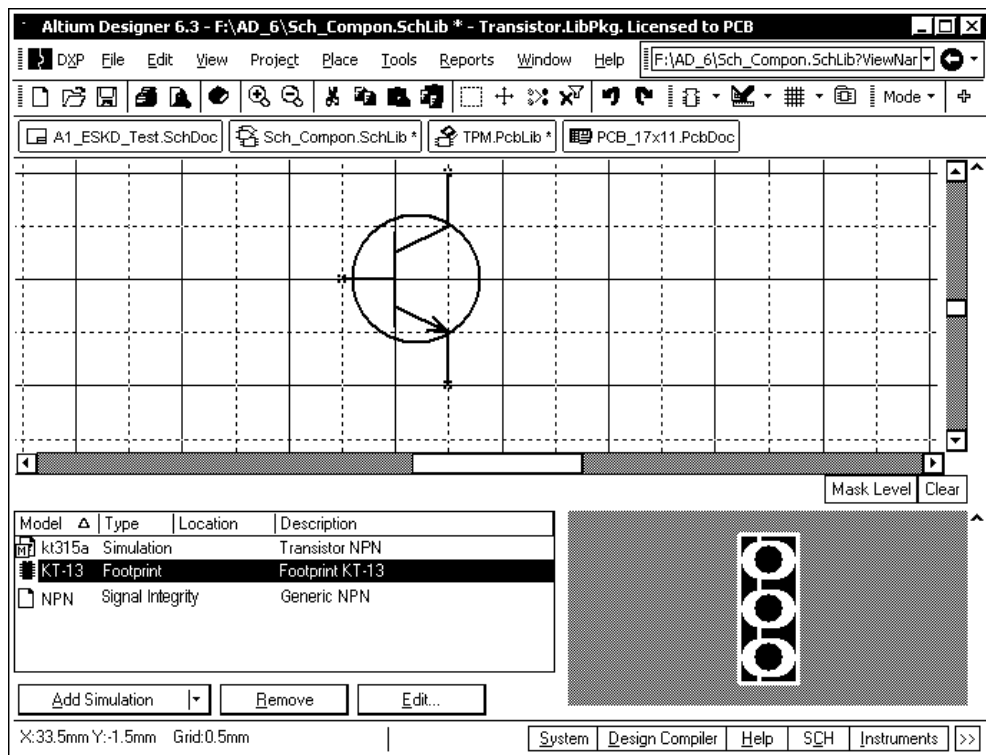


Рис. 2.18. Главное окно редактора библиотек Altium Designer

В главном поле располагается схемное УГО компонента, слева внизу — список ссылок на модели и кнопки управления функциями редактирования, в правом нижнем поле отображается ТПМ компонента.

2.5. Компиляция интегрированной библиотеки

В результате рассмотренных процедур получается библиотечный пакет (проект), иерархическое дерево которого объединяет библиотеку схемных символов, ТПМ и другие модели компонентов. Однако, как уже говорилось, защита компонента от случайных повреждений, компактность библиотеки и наибольшие удобства использования достигаются, когда эти составляющие объединяются в интегрированную библиотеку. Рассмотрим процедуру фор-

мирования такой библиотеки на примере разработанной библиотеки транзисторов (пока что всего с одним NPN-транзистором).

1. Щелчком правой кнопки мыши на имени библиотечного проекта **Transistor.LibPkg** в плавающей панели **Projects** активизировать контекстное меню и указать в нем команду **Compile Integrated Library Transistor.LibPkg** (Компилировать интегрированную библиотеку Transistor.LibPkg).
2. Если компиляция проходит без ошибок, формируется интегрированная библиотека **Transistor.IntLib**. Библиотека автоматически сохраняется в подкаталоге ...\\Project Outputs for Transistor\\Transistor.IntLib того каталога, в котором находятся файлы-источники — библиотека схемных компонентов и библиотека ТПМ.

Файлы описания моделей также копируются и компилируются в интегрированную библиотеку. Новая интегрированная библиотека включается в состав библиотек, подключенных к рабочему пространству Altium Designer, и ее компоненты могут в дальнейшем использоваться в сквозном проектировании радиоэлектронных функциональных узлов.

2.6. Конверсия библиотек P-CAD 200x в формат Altium Designer

При отказе от использования P-CAD и переходе к проектированию в среде Altium Designer необходимо обеспечить по возможности безболезненное преобразование данных от одного формата к другому, чтобы можно было поддерживать и развивать ранее выполненные проекты. Altium Designer предоставляет средства такой конверсии, называемые **Import Wizard** (Мастер, или дословно — волшебник импорта).

Проекты и библиотеки, подлежащие конверсии, необходимо преобразовать в текстовый формат ACCEL ASCII в среде P-CAD 200x. Преобразование библиотечных наборов выполняется в среде **Library Executive** (Программный модуль работы с библиотеками) САПР P-CAD 200x.

1. Активизировать команду главного меню **Library** (Библиотека) | **Translate** (Конвертировать). Открывается одноименное диалоговое окно (рис. 2.19).
2. В поле **Source Format** (Формат источника) выбрать формат **P-CAD Binary** (Бинарный формат P-CAD).

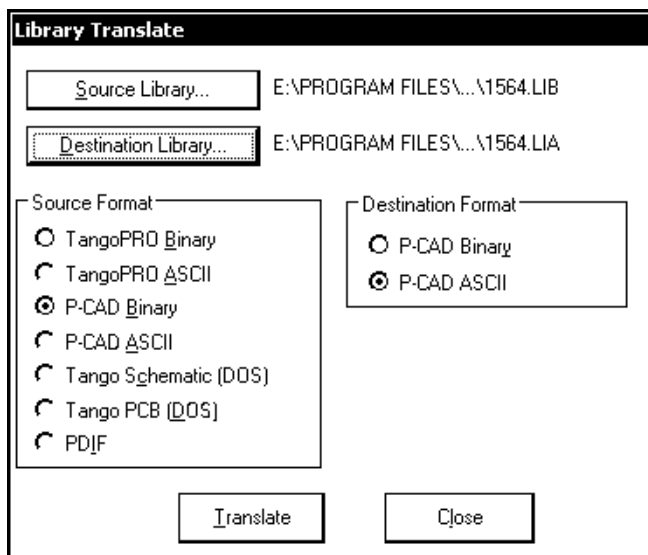


Рис. 2.19. Окно конверсии библиотек P-CAD 200x

3. В поле **Destination Format** (Формат в результате преобразования) указать текстовый формат **P-CAD ASCII**.
4. Щелчком по кнопке **Source Library** (Исходная библиотека) активизировать стандартную процедуру поиска исходного библиотечного файла в двоичном формате *.lib.
5. Щелчком по кнопке **Destination Library** (Целевая библиотека) открыть диалог назначения имени конвертированного библиотечного файла. Имя файла должно иметь расширение lia.
6. Кнопкой **Translate** (Преобразовать) запустить преобразование. Закончить процедуру кнопкой **Close** (Заккрыть).

Следующий этап — собственно конверсия структуры библиотек P-CAD 200x в формат Altium Designer.

1. Запустить процедуру преобразования командой **File | Import Wizard** ("Мастер" импорта). Откроется диалоговое окно импорта. Кнопкой **Next** (Следующий) перейти к первому шагу импорта. В диалоговом окне (рис. 2.20) предоставляется возможность выбора исходного формата данных.

Предусмотрена возможность выбора любого из семи исходных форматов. Выбираем формат данных **P-CAD Design and Libraries Files** (Фай-

лы проектов и библиотек P-CAD) и кнопкой **Next** переходим к следующему шагу.

На следующих двух шагах предоставляется возможность выбора импорта файлов проекта или библиотек.

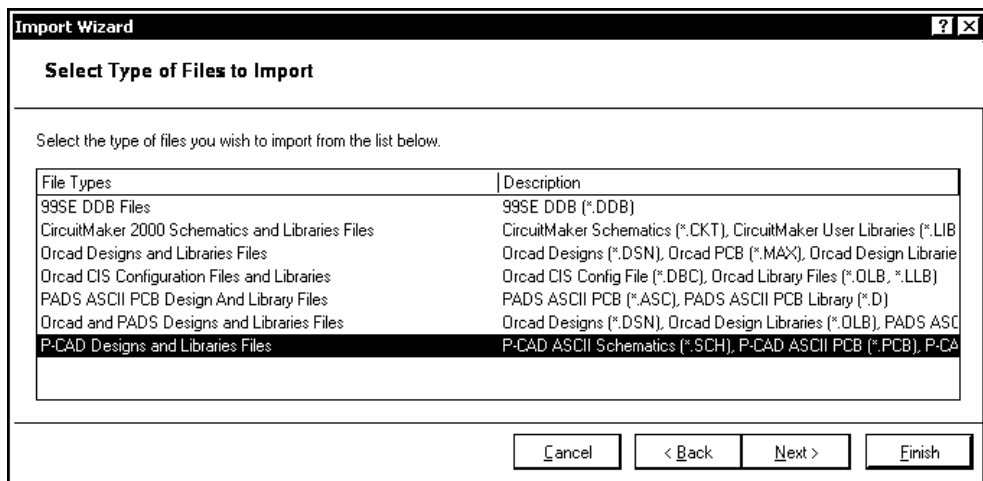


Рис. 2.20. Окно "мастера" импорта Altium Designer

- При выборе импорта библиотек открывается диалоговое окно с обширным предупреждением, сообщаящим о различии форматов библиотек P-CAD и Altium Designer и об особенностях преобразования (рис. 2.21).

В отличие от P-CAD, в котором описания контактов компонента и связи электрических контактов с контактными площадками посадочного места, а также признаки эквивалентности контактов и логических секций сосредоточены в структуре "компонента", объединяющей схемный символ (Symbol) с посадочным местом (Pattern), при конверсии в формат Altium Designer все эти данные передаются в структуру элемента библиотеки схемного редактора AdvSCH. Графика посадочного места и описания контактных площадок передаются в структуру элемента библиотеки графического редактора печатной платы AdvPCB.

Вследствие этих различий программа предлагает выбрать формат имени посадочного места. Имя может состоять из трех полей (рис. 2.21). Поля можно оставить пустыми либо внести в них имя компонента, имя посадочного места P-CAD и графики посадочного места (первичная и т. п.).

Можно также выбрать разделитель полей в имени посадочного места в Altium Designer — точку, дефис или скобки.

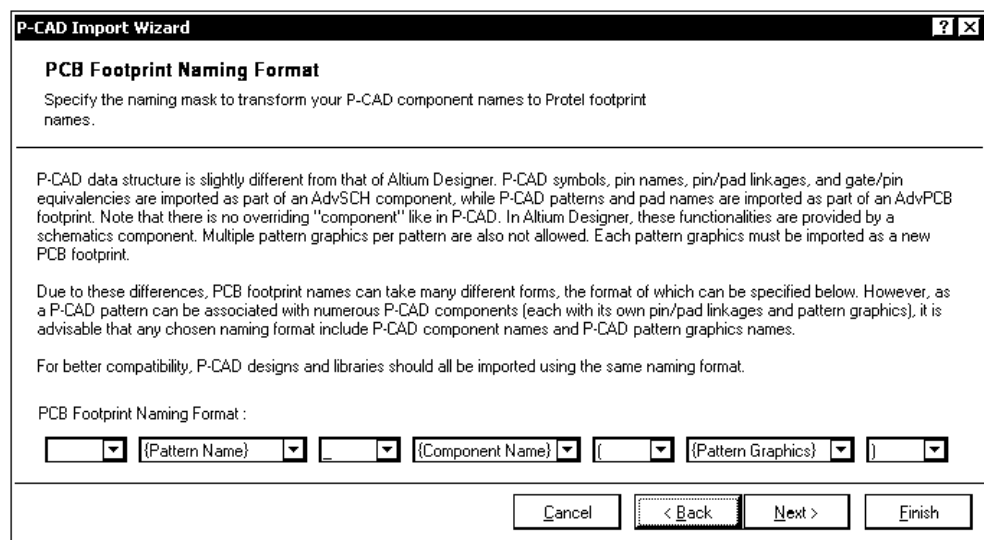


Рис. 2.21. Окно управления ТПМ при импорте библиотеки

Необходимо выбрать формат имени ТПМ в будущей библиотеке Altium Designer и кнопкой **Next** перейти далее (рис. 2.22).

3. На следующем шаге в диалоговом окне (рис. 2.22) указан путь и представлена структура файлов конвертированной библиотеки.

В структуре базы данных Altium Designer строится ссылка на новую библиотеку, как на проект, состоящий из двух компонентов: библиотеки схемных элементов — файл *.SchLib, и библиотеки посадочных мест — файл *.PcbLib. Кнопкой **Next** запустить преобразование.

4. По окончании конверсии завершить процедуру кнопкой **Finish**.

Структура дерева этого проекта отобразится в окне проектов пакета (рис. 2.23). Дерево состоит из двух ветвей: библиотеки символов *.SchLib для схемного редактора и библиотеки посадочных мест *.PcbLib для редактора печатной платы.

5. Щелчком правой кнопки мыши на имени проекта конвертированной библиотеки активизировать команду компиляции интегрированной библиоте-

ки **Compile Integrated Library**. Компилированная библиотека *.IntLib присоединится к рабочей среде пакета.

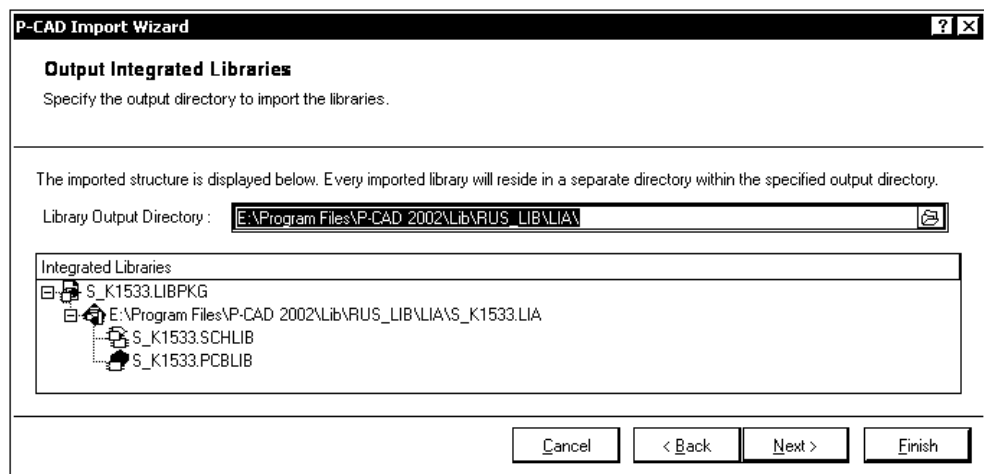


Рис. 2.22. Структура файлов конвертированной библиотеки

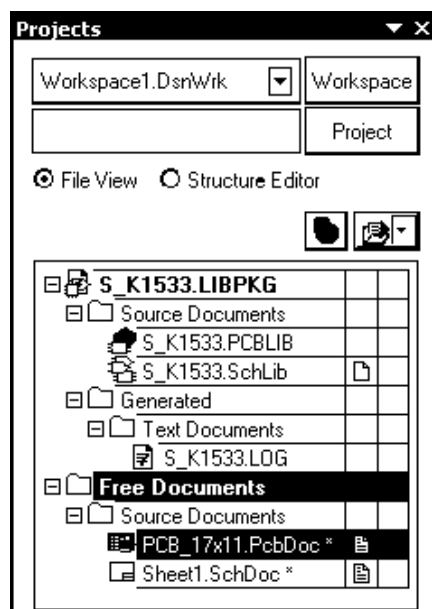


Рис. 2.23. Отображение структуры конвертированной библиотеки на панели **Projects**

Автоматически открывается окно просмотра присоединенных библиотек. В полях просмотра будут представлены имя новой скомпилированной библиотеки и состав входящих в нее компонентов. Многосекционные компоненты отобразятся в виде свернутого дерева, ветви которого можно развернуть так, что станет доступной установка на поле схемы отдельных секций по выбору.

2.7. Включение библиотек в рабочую среду Altium Designer

Для того чтобы получить возможность использования библиотек схемных элементов в редакторе электрической схемы или посадочных мест компонентов в редакторе печатной платы, библиотеки должны быть включены в рабочую среду Altium Designer.

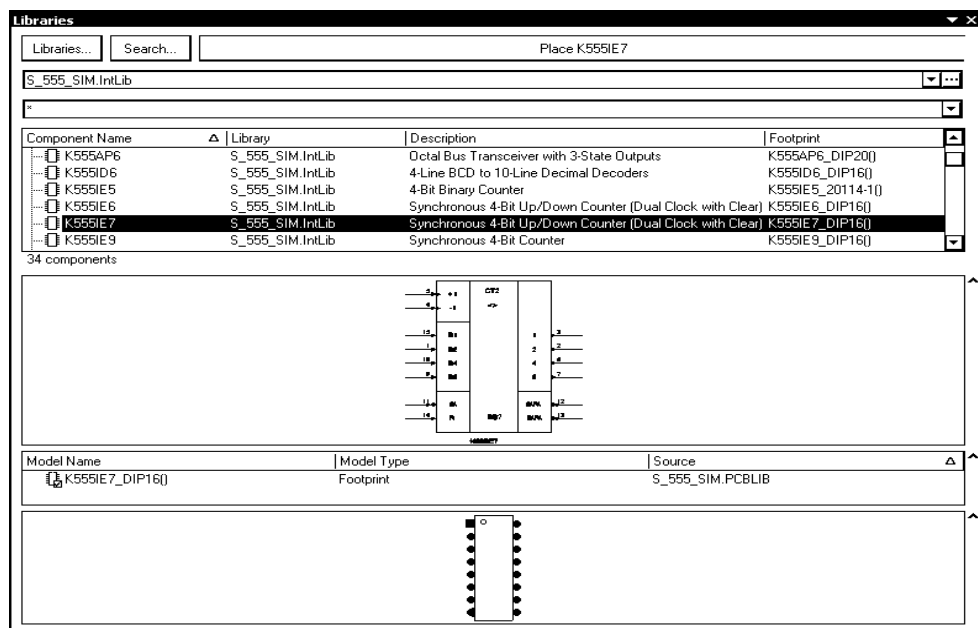


Рис. 2.24. Плавающая панель Libraries

Процедура включения интегрированных библиотек в рабочую среду программы следующая:

1. Активизировать вкладку **Libraries** на краю рабочего окна программы или вызвать цепочку команд главного меню: **View | Workspace Panels | System | Libraries**. Откроется плавающая панель **Libraries** (рис. 2.24).

В полях этой панели представлены имя одной из ранее подключенных библиотек, состав ее компонентов и их краткое описание, условно-графическое обозначение схемного элемента, на который указывает курсор селекции, сведения о модели компонента: в простейшем случае это имя топологического посадочного места (ТПМ) и библиотеки, в которой ТПМ хранится, а также его графическое изображение.

Панель может растягиваться или сжиматься обычным для окон MS Windows способом, перемещаться по экрану и "приклеиваться" к краям рабочего графического окна программы.

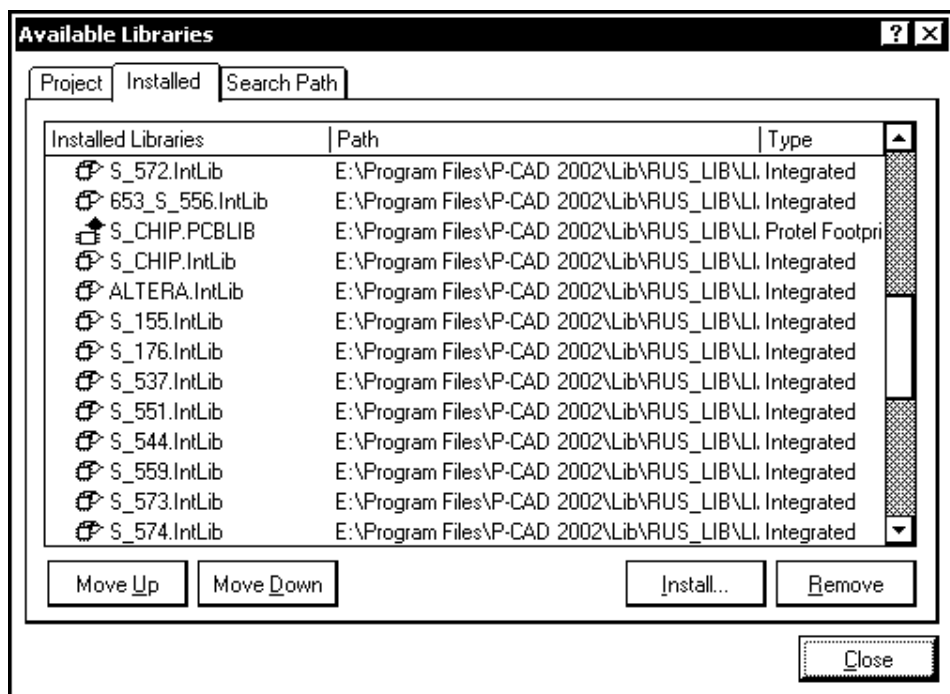


Рис. 2.25. Диалог включения библиотек
в рабочую среду Altium Designer

2. Для присоединения недостающих библиотек щелчком на кнопке **Libraries** в верхней части панели нужно открыть окно со списком установленных библиотек **Installed Libraries** (рис. 2.25).
3. Кнопкой **Install** (Установить) активизировать стандартный диалог поиска библиотек для присоединения их к рабочей среде программы. Найденные библиотеки подключить щелчком на кнопке **Open** (Открыть) в окне поиска.
4. Кнопкой **Close** (Заккрыть) в окне (см. рис. 2.25) завершить подключение библиотек.

Если при этом было открыто окно текущего проекта электрической схемы, выбранные символы подключенных библиотек можно вызвать на поле проектируемой схемы кнопкой **Place<Имя_компонента>**.

Для исключения библиотек из проекта следует вызвать диалоговое окно **Available Libraries** (см. рис. 2.25), указать в нем ненужные библиотеки и удалить их из списка кнопкой **Remove** (Удалить).

2.8. Поиск компонентов в интегрированных библиотеках

Altium Designer располагает эффективными средствами поиска нужного компонента в библиотеках, когда путь к нему и имя библиотеки неизвестны. Поиск выполняется следующим образом:

1. Активизировать поиск кнопкой **Search** (Поиск) в панели **Libraries** (см. рис. 2.24). Откроется диалоговое окно **Libraries Search** (рис. 2.26).
2. В верхнем, текстовом, поле окна указать "маску поиска" — написать группу символов, которая наверняка является частью имени искомого компонента. Звездочки слева и справа от написанных символов расширяют поиск на все имена во всех библиотеках, в том числе и неподключенных, частью которых является указанная последовательность символов.
3. В поле **Options** выбрать в выпадающем списке **Search type** тип объектов поиска: библиотечные компоненты, посадочные места, трехмерные модели или компоненты из баз данных.
4. Кнопкой **Helper** (Помощник) можно активизировать функцию расширения маски поиска — указать дополнительные признаки, их логическое сочетание, параметры и (или) арифметические операции над ними и т. д.

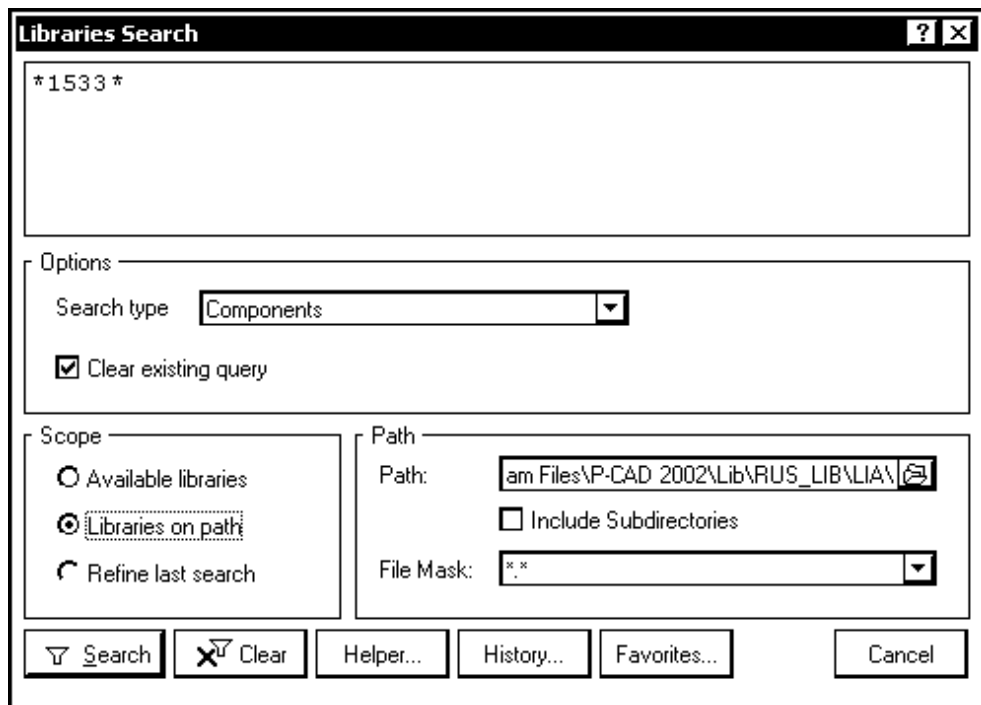


Рис. 2.26. Окно поиска в библиотеках по заданным признакам

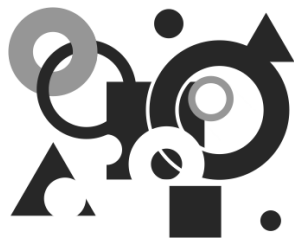
5. В поле **Scope** (Обзор) задать область поиска библиотечного компонента:

- **Available libraries** (Доступные библиотеки) — поиск в доступных (инсталлированных) библиотеках;
- **Libraries on path** (Библиотеки на пути) — поиск по заданному пути.

В последнем случае в поле **Path** необходимо указать путь, по которому следует искать библиотеки.

6. Кнопкой **Search** начать поиск компонента.

Результаты поиска отображаются в плавающей панели **Libraries**, в списке **Query Results** (Результаты запроса), а значки найденных компонентов — в поле просмотра. Имя выбранного компонента появляется на кнопке **Place** в диалоговом окне (см. рис. 2.24). Щелчком на кнопке **Place** компонент, в зависимости от выполняемой стадии проектирования, выносится на поле графического листа схемы или печатной платы, где можно зафиксировать нужное число его копий.



Настройка конфигурации графических редакторов

Проект печатной платы Altium Designer должен содержать как минимум два основных документа: электрическую принципиальную схему и интегральный графический образ печатной платы. Процедура образования нового проекта и включения в его логическую структуру новых документов (листа электрической схемы и листа графического редактора печатной платы) была рассмотрена в *разд. 1.2.3.*

Будем считать, что разрабатывается проект ячейки фильтров нижних частот PCB_Filter.PrjPcb и в структуре проекта образованы документ схемного редактора (лист схемы PCB_Filter.SchDoc) и лист графического редактора печатной платы (PCB_Filter.PcbDoc).

В следующих разделах рассмотрим настройку конфигурации графического листа редактора схемы и графического листа редактора печатной платы. При этом поставим задачу оформления документа электрической принципиальной схемы по ЕСКД.

3.1. Конфигурация графического редактора схем

3.1.1. Лист схемы по ЕСКД

Как при образовании нового схемного документа в структуре активного проекта, так и при открытии свободного документа командой **File | New | Sche-**

matic в рабочем пространстве схемного редактора открывается лист формата A1 одного из "западных" стандартов (размеры листа — 800×565 мм). Размеры поля и графление бланка основной надписи отличаются от принятого в ЕСКД. По краям листа располагается рамка, разбитая на зоны, обозначенные по вертикали латинскими буквами, а по горизонтали — цифрами. Рассмотрим процедуру формирования бланка форматки по ГОСТ 2.301-68 с основной надписью по ГОСТ 2.104-2006 на примере листа формата A2:

1. Активизировать команду главного меню **Design** (Разработка) | **Document Options** (Настройки документа). Откроется диалоговое окно **Document Options** с тремя панелями-вкладками: **Sheet Options** — настройки листа, **Parameters** — реквизиты документа и **Units** — система единиц измерения (рис. 3.1).

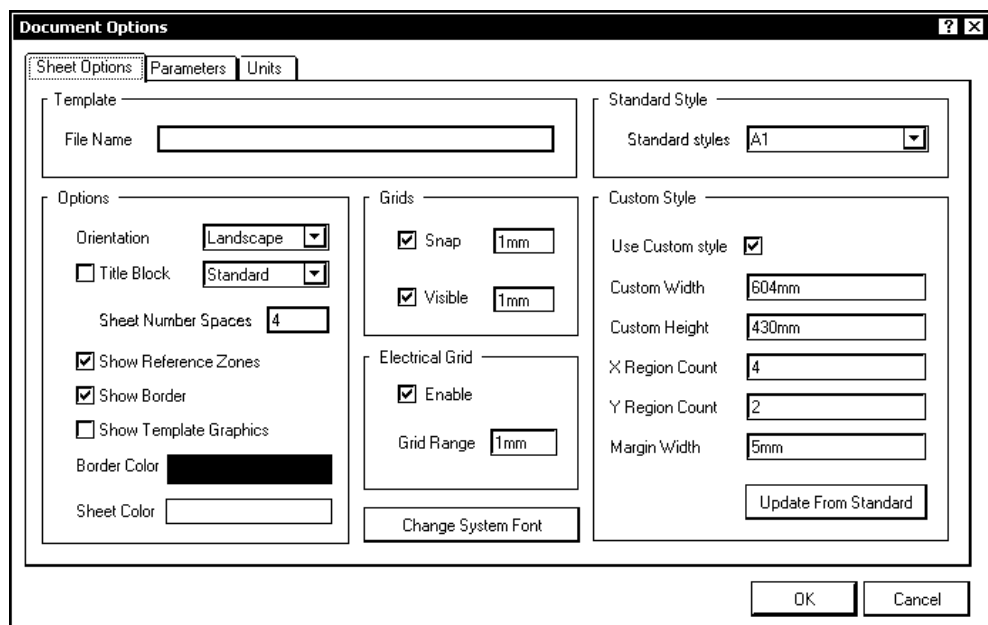


Рис. 3.1. Окно настройки конфигурации редактора схем Altium Designer

2. Открыть вкладку **Units** (Единицы) и установить метрическую систему единиц — **Metric**. Перейти на вкладку **Sheet Options** (Настройки листа).
3. В поле **Options** отключить видимость основной надписи, пришедшей при открытии документа — убрать флажок **Title Block** (Блок "штампа" черте-

жа); оставить видимыми границы листа и обозначения зон разметки — флажки **Show Border** (Показать границы) и **Show Reference Zones** (Показать разбивку на зоны).

4. В поле **Grids** (Сетки) установить значения шага сетки захвата (**Snap**) и видимой сетки (**Visible**) равными 1 мм.
5. В поле настроек **Custom Style** (Пользовательский стиль) активизировать настройку пользовательского формата (флажок **Use Custom style**) и назначить размеры листа по горизонтали и по вертикали.

При этом необходимо учитывать следующее. Зонную разметку нужно сохранить, чтобы впоследствии, в схеме, была возможность перенумерации позиционных обозначений. С другой стороны, если оставить обозначения зон в пределах размеров стандартного листа 594×420 мм, то их размещение в рамке по левому полю листа входит в противоречие с необходимостью расположить в этом поле по ЕСКД дополнительные графы основной надписи. Кроме того, настройка листа позволяет установить единую ширину поля по всему периметру листа, в то время как лист по ЕСКД должен иметь левое поле шириной 20 мм. Поэтому представляется целесообразным установить размеры листа такими, чтобы лист лежал краями по внутреннему краю рамки с разметкой зон. Чтобы добиться этого, прибавим к стандартным размерам листа по 10 мм на каждую сторону и для листа формата A2 назначим пользовательскую ширину **Custom Width** и высоту **Custom Height** 604×430 мм, а ширину полей, содержащих зонную разметку, — **Margin Width** 5 мм.

Число зон разметки по горизонтали и по вертикали (**X Region Count** и **Y Region Count**) назначим 4 и 2 соответственно.

6. Щелчком на кнопке **OK** завершим настройки.
7. В главном меню программы активизировать команду черчения линий **Place** (Разместить) | **Drawing Tools** (Инструменты черчения) | **Line** (Линия). Клавишей <Tab> вызвать диалоговое окно настройки параметров линии (см. рис. 2.3).

Выбрать толщину линии **Small** (Малая) и построить в графическом окне схемного редактора внешний прямоугольник размерами 594×420 мм по внутренней рамке зонной окантовки листа.

Повторить диалог настройки толщины линии: выбрать среднюю толщину линии **Medium** и вычертить внутренний прямоугольник из начальной точки с координатами X = 25 мм, Y = 10 мм так, чтобы он образовал внутрен-

- ную рамку форматки A2: левое поле шириной 20 мм; остальные поля по 5 мм.
8. В правом нижнем углу листа вычертить графы основной надписи; в левом верхнем углу и на левом поле — дополнительные графы по ГОСТ 2.104-2006.
 9. Активизировать команду главного меню **Place** (Разместить) | **Text String** (Строку текста) и расставить шаблон "Text" в те графы основной надписи, где должны располагаться неизменяемые элементы — *Разраб.*, *Пров.*, *Т. контр.*, *Н. контр.* и др.

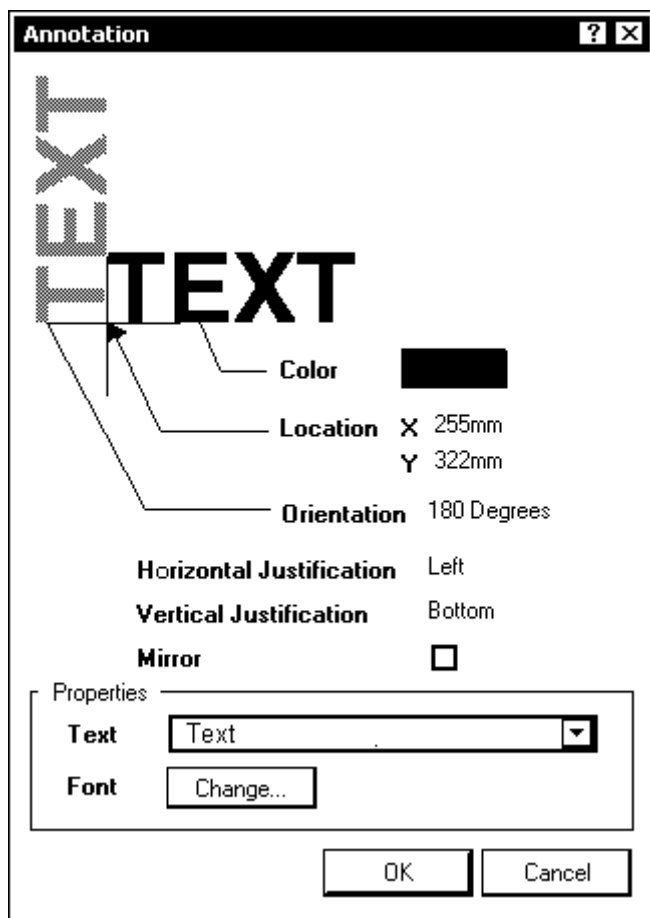


Рис. 3.2. Окно редактирования параметров текстовой строки

После этого двойным щелчком левой кнопки мыши на каждом из шаблонов "Text" по очереди вызвать окно редактирования текстовой строки (рис. 3.2) и ввести на место слова "Text" соответствующий неизменяемый элемент основной надписи. В этом же окне можно изменить ориентацию записи (**Orientation**), ее центровку по горизонтали (**Horizontal Justification**) и по вертикали (**Vertical Justification**), установить зеркальное изображение (**Mirror**), выбрать цвет символов (**Color**).

Кроме того, щелкнув по кнопке **Change** (Изменить), можно изменить гарнитуру текста (**Font**), размер и наклон символов. Применительно к нашей задаче следует назначить для неизменяемых элементов гарнитуру Arial, курсив, высоту (кегель) 14 пунктов.

3.1.2. Реквизиты документа

Реквизитами документа называются элементы основной надписи, относящиеся к разрабатываемому проекту: наименование изделия, обозначение документа по ЕСКД, фамилии разработчика и других лиц, имеющих право работать с документом — всего до 16 записей. В Altium Designer реквизиты, прежде чем оказаться на чертеже, заносятся в список параметров документа:

1. Командой главного меню **Design | Document Options** активизировать диалоговое окно настройки параметров документа (см. рис. 3.1) и открыть вкладку **Parameters**.

В колонке **Name** (Имя) располагаются имена реквизитов (по-английски) — **Author** (Разработал), **Checked By** (Проверил) и др.

В колонку **Value** (Значение) нужно записать имена разработчика, других лиц, наделенных правом проверки и утверждения документа, обозначение документа, а также другие реквизиты проекта.

На стадии формирования шаблона листа целесообразно проанализировать этот список, подобрать в нем подходящие по смыслу англоязычные записи и добавить несколько реквизитов, для которых нет подходящего эквивалента.

ВНИМАНИЕ!

Программа не допускает использования символов кириллицы в именах реквизитов, поэтому имена добавляемых реквизитов приходится писать латинскими буквами.

Приведем список реквизитов документа по ЕСКД и существующих англоязычных эквивалентов:

- Разработал (Author);
 - Проверил (Checked By);
 - Утвердил (Approved By);
 - Обозначение по ГОСТ 2.201-80 (Drawing Number);
 - Наименование изделия (Title);
 - Индекс предприятия (Company Name);
 - Лист (SheetNumber);
 - Листов (SheetTotal).
2. Кнопкой **Add** открыть диалог назначения новых реквизитов и добавить к имеющемуся списку реквизиты, необходимые для конструкторских документов по ЕСКД (ГОСТ 2.104-2006):
- Технологический контроль — имя TechContr;
 - Нормоконтроль — имя NormContr;
 - Тип документа (обозначение типа схемы по ГОСТ 2.701-84) — имя DocType;
 - Литера — три реквизита с именами Lit1, Lit2, Lit3.
3. Сохранить построенный лист как файл шаблона (Schematic Template) с именем A2_ESKD.SchDot в пользовательском подкаталоге ...\\Templates.

3.1.3. Заполнение основной надписи документа

Поскольку такие реквизиты электрической принципиальной схемы радиоэлектронного функционального узла, как наименование узла, обозначение документа по ГОСТ 2.201-80 (на профессиональном жаргоне конструкторов — так называемый десятичный номер), обозначение типа схемы (запись "Схема электрическая принципиальная"), фамилия разработчика, присваиваются документу еще до начала разработки схемы, имеет смысл заполнить на

первой стадии работы над проектом — сразу после того как откроется файл шаблона листа:

1. Комбинацией клавиш <Ctrl>+<N> открыть плавающую панель **Files**, затем в поле **New from Template** (Новый на основе шаблона) указать команду поиска шаблонов листов схемы **Schematic Templates**.
2. Выбрать в пользовательском каталоге ...\\Templates файл шаблона листа по ЕСКД, например A2_ESKD.SchDot, и открыть файл.
3. Щелчком правой кнопки мыши открыть контекстное меню и выполнить последовательность команд **Options | Document Parameters**. Откроется диалоговое окно **Document Options** (см. рис. 3.1) с активной вкладкой **Parameters**.
4. По очереди выбрать в списке параметров (реквизитов документа) необходимые реквизиты документа и для каждого двойным щелчком левой кнопки мыши на пустом поле колонки **Value** открыть диалог редактирования свойств реквизита — **Parameter Properties** (рис. 3.3).

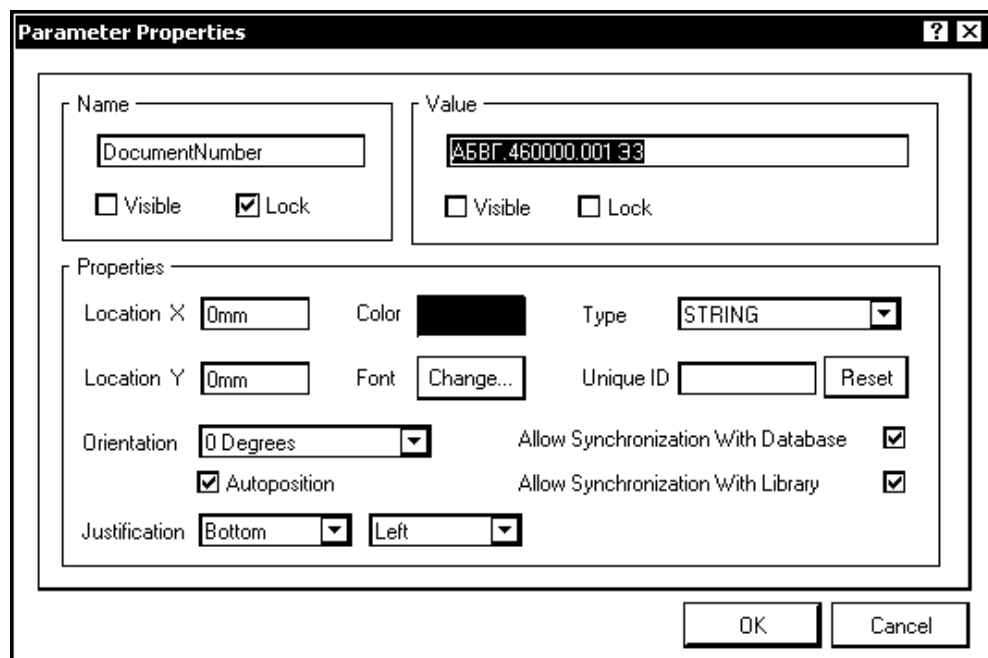


Рис. 3.3. Окно редактирования реквизитов документа

5. В поле **Value** диалогового окна вписать строку "значения" реквизита, например обозначение документа схемы электрической принципиальной по ГОСТ 2.201-80 — АБВГ.467222.001 Э3. Установить для реквизита свойство видимости (**Visible**).
6. В поле **Properties** (Свойства) установить, если необходимо, ориентацию надписи (**Orientation**), ее центровку (**Justification**), задать цвет символов (**Color**) и гарнитуру текста (**Font | Change**).
7. После того как значения всех реквизитов записаны и диалог закрыт, щелчком правой кнопки мыши снова открыть контекстное меню и выполнить в нем последовательность команд **Options** (Настройки) | **Schematic Preferences** (Предпочтения схемного редактора). Откроется диалог настройки функций схемного редактора — окно **Preferences**.
8. В дереве функций в левой половине диалогового окна **Preferences** выбрать ветвь **Schematic | Graphical Editing** (Графическое редактирование), после чего в правой половине окна активизировать опцию **Convert Special Strings** (Конвертировать специальные строки). Нажав кнопку **OK**, закрыть диалог настройки функций схемного редактора.
9. Выполнить команду главного меню **Place | Text String** и нажатием клавиши <Tab> вызвать диалоговое окно редактирования текстовой строки (см. рис. 3.2).
10. В поле **Properties** окна редактирования текста щелчком мыши развернуть выпадающий список в строке **Text** (рис. 3.4). В списке представлены "специальные строки" — имена реквизитов из поля параметров документа с предшествующим им знаком равенства.

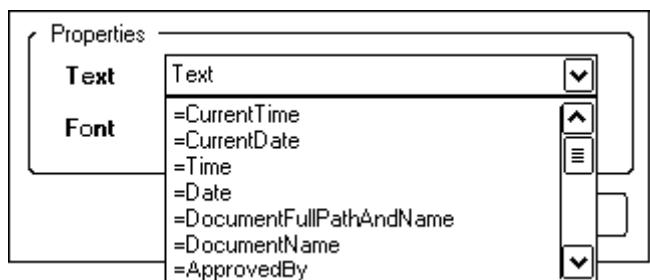


Рис. 3.4. "Специальные строки" в списке параметров документа

11. Указать курсором строку **=DocumentNumber** и закрыть диалог щелчком на кнопке **ОК**. На поле графического листа появится и станет перемещаться вместе с курсором мыши строка с обозначением документа, в нашем случае это АБВГ.467222.001 ЭЗ. Поместить эту строку в первую графу основной надписи документа и зафиксировать щелчком левой кнопки мыши.

Аналогично перенести остальные реквизиты документа из списка параметров в графы основной надписи.

ЗАМЕЧАНИЕ

При выборе в списке параметров строки **=DocumentName** на поле чертежа автоматически выводится имя файла схемы или шаблона независимо от того, какая запись была помещена в поле **Value** списка реквизитов в диалоговом окне настройки параметров документа. Это дает возможность поместить имя файла документа в одну из свободных или дополнительных граф основной надписи, как это требуется по ГОСТ 2.104-2006, в который включены нормы, касающиеся сведений об электронном документе, на основании которого выпускается "твердая копия" (чертеж).

В результате рассмотренной процедуры мы получим чистый лист схемы электрической принципиальной разрабатываемого функционального узла стандартного формата А2 по ГОСТ 2.301-68, с заполненными графами основной надписи по ГОСТ 2.104-2006, готовый к составлению и редактированию схемы.

Сохраним подготовленный таким образом лист схемного документа как свободный документ с именем PCB_Filter.SchDoc.

3.2. Настройка конфигурации графического редактора печатной платы

3.2.1. Задание размеров листа

Для назначения размеров листа документа следует:

1. В плавающей панели **Projects** активизировать документ графического редактора печатной платы PCB_Filter.PcbDoc. В главном окне программы откроется графический лист редактора печатной платы с размещенным в центре листа контуром печатной платы.

2. Выполнить команду главного меню **Design | Board Options** или щелчком правой кнопки мыши в поле графического листа вызвать контекстное меню и указать в нем последовательность команд **Options | Board Options**. Откроется диалоговое окно управления свойствами листа **Board Options** (рис. 3.5).

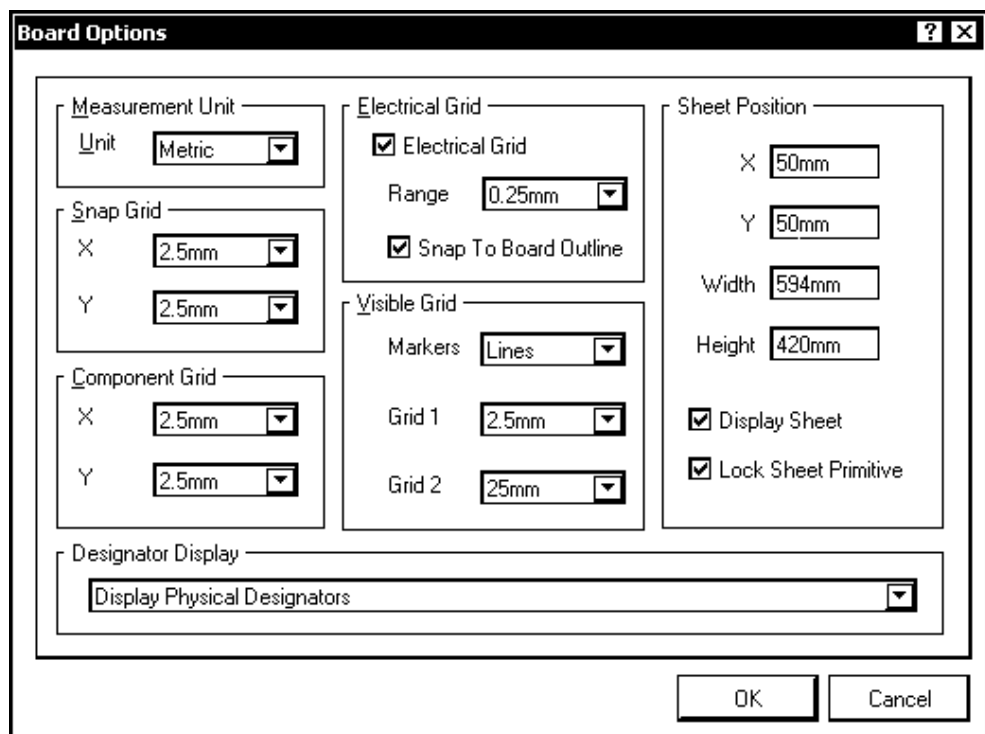


Рис. 3.5. Окно настройки конфигурации графического редактора печатной платы

3. В поле **Sheet Position** (Положение листа) назначить положение и размеры графического листа, на котором может формироваться рабочий чертеж печатной платы, и указать координаты левого нижнего угла листа в рабочем пространстве графического редактора печатной платы.
4. Установить видимость листа. При включенной видимости листа заготовка платы изображается на экране монитора на белом поле.

3.2.2. Сетки проектирования

При разработке проекта для размещения компонентов и других объектов на поле печатной платы назначаются сетки. Управление формированием сеток выполняется по команде главного меню **Design | Board Options** ("горячие" клавиши <D> и <O>) либо из контекстного меню, вызываемого щелчком правой кнопки мыши в поле открытого листа проекта, командами **Options | Board Options, Sheet** или **Grids**. Во всех случаях открывается диалоговое окно **Board Options** (рис. 3.5).

1. Для размещения компонентов (и только для этого) назначить сетку **Component Grid**. Значение шага сетки по горизонтали и по вертикали выбрать из выпадающего списка или ввести с клавиатуры.
2. Для размещения прочих объектов — линий, вырезов и т. п., в том числе и печатных проводников — назначить сетку **Snap Grid** (Сетка захвата). Ее шаг также назначается из выпадающего списка или с клавиатуры и должен находиться в дробном или кратном соотношении с шагом сетки компонентов, чтобы можно было легко попадать курсором в точки расположения выводов компонентов при трассировке печати. "Горячая" клавиша <G> вызывает контекстное меню, в котором шаг сетки **Snap Grid** можно изменить.
3. Назначить "электрическую" сетку (**Electrical Grid**), служащую для размещения и присоединения друг к другу электрических объектов (печатных проводников). Данная сетка обладает приоритетом перед сеткой захвата **Snap Grid**. Шаг сетки назначается в зависимости от класса точности и технологии выполнения печатного проводящего рисунка. Попадая в пределы одного шага данной сетки, электрические объекты соединяются друг с другом — происходит их захват.

ЗАМЕЧАНИЕ

Комбинация "горячих" клавиш <Shift>+<E> активизирует или отключает электрическую сетку в любой момент работы с проектом.

4. Кроме этих сеток нужно назначить две так называемые видимые сетки **Visible Grids**. Они предназначены только для визуального контроля действий при размещении объектов, их выравнивании и других операциях графического редактирования.

При разработке нового проекта необходимо определить геометрию и структуру слоев заготовки печатной платы. Первый шаг по формированию структуры заготовки целесообразно выполнить, пользуясь встроенной в Altium Designer функцией мастера подсказки **PCB Board Wizard**.

Рассмотрим процедуру по шагам.

1. Активизировать плавающую панель меню **Files**. Для этого в главном меню программы выполнить цепочку команд **File | New | Other | Files Panel** либо использовать комбинацию "горячих" клавиш <Ctrl>+<N>.
2. В поле **New from Template** панели **Files** вызвать последнюю команду: **PCB Board Wizard**. Откроется диалоговое окно настройки параметров заготовки печатной платы. Настройка выполняется в несколько шагов. На каждом шаге в диалоговом окне **PCB Board Wizard** (рис. 3.6) предлагается выполнить ряд установок. Последовательный переход от шага к шагу выполняется по щелчкам на кнопках **Next** и **Back**. Состав и содержание полей диалогового окна меняются в зависимости от характера выполняемых на очередном шаге операций.

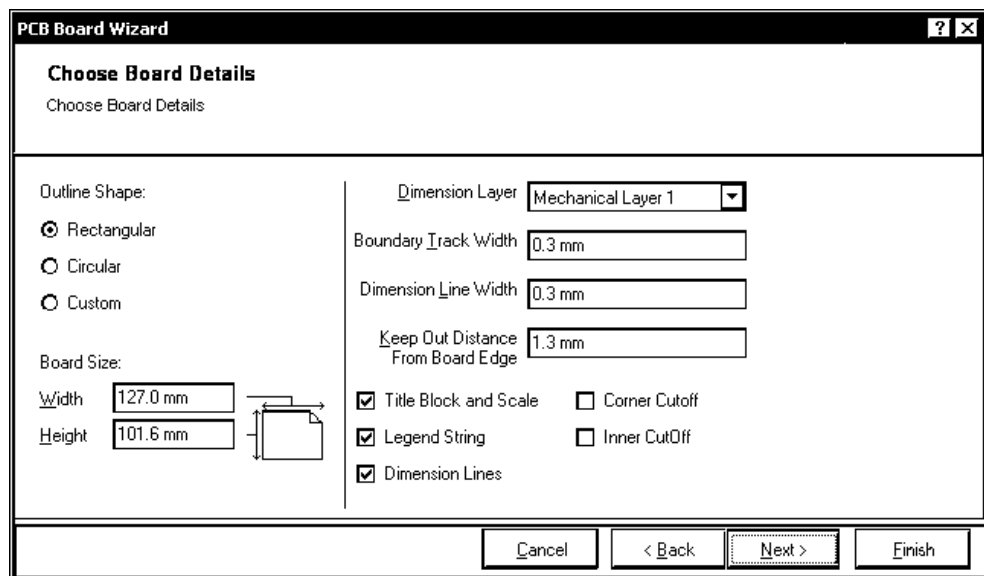


Рис. 3.6. Окно мастера построения заготовки печатной платы

3. На шаге **Choose Board Units** определить систему единиц измерения — указать **Metric**.
4. На шаге **Choose Board Profiles** (Выбрать прототип платы) откроется список из 60 возможных вариантов готовой заготовки. Сюда входят несколько типоразмеров "евро-платы", несколько вариантов адаптера PCI-шины и другие. Для формирования заготовки из ряда типоразмеров УБНК1 по ГОСТ 26765.12-86 выбираем вариант пользовательских установок — **Custom**.
5. На шаге **Choose Board Details** (рис. 3.6) указать:
 - форму платы — прямоугольную (**Rectangular**), круглую (**Circular**) или произвольную (**Custom**), ограниченную прямоугольными отрезками и (или) дугами;
 - размеры платы по горизонтали и по вертикали — **Board Size**;
 - слой для размещения размерных линий (выбрать из числа "механических" слоев — по умолчанию — **Mechanical Layer 1**);
 - толщину (ширину) линии, ограничивающей контур платы — **Boundary Track Width**;
 - толщину размерных линий — **Dimension Line Width**;
 - расстояние от края платы до ограничительной линии зоны трассировки печати — **Keep Out Distance From Board Edge**.
6. На шаге **Choose Board Layers** (Выбрать слои печатной платы) указать число сигнальных (**Signal Layers**) и экранных слоев печатной платы (**Power Planes**). Необходимые значения назначают в полях просмотра диалогового окна.
7. На следующем шаге, **Choose Via Style**, предлагается выбрать один из двух стилей межслойных переходных отверстий — только сквозные (**Thruhole Vias only**) или только слепые и захороненные переходные отверстия (**Blind and Buried Vias only**). В диалоговом окне показывается фрагмент разреза платы для выбранного случая.
8. На следующем шаге, **Choose Component and Routing Technologies** (Выбрать компоненты и технологию трассировки), следует выбрать преобладающий тип компонентов — поверхностно монтируемые (**Surface-mount Components**) или монтируемые штырями в сквозные отверстия (**Through-hole Components**). Для поверхностно монтируемых нужно указать правила установки компонентов — на одной или на обеих сторонах платы. Для

штыревых компонентов предлагается указать, сколько печатных проводников допустимо провести между двумя переходными (монтажными) отверстиями, расположенными с шагом 2,54 мм, — один, два или три.

9. На следующем шаге, **Choose Default Track and Via sizes** (Выбрать ширину проводника и площадки металлизации у переходных отверстий по умолчанию), в диалоговом окне приводятся минимальные значения ширины проводника, зазоров и диаметры контактной площадки и отверстия (табл. 3.1).

Таблица 3.1. Минимальные размеры элементов печати

Число проводников	Ширина проводника, мм	Диаметр контактной площадки, мм	Диаметр отверстия, мм	Зазоры, мм
1	0,3	1,6	0,8	0,3
2	0,2	1,0	0,6	0,2
3	0,2	0,6	0,3	0,2

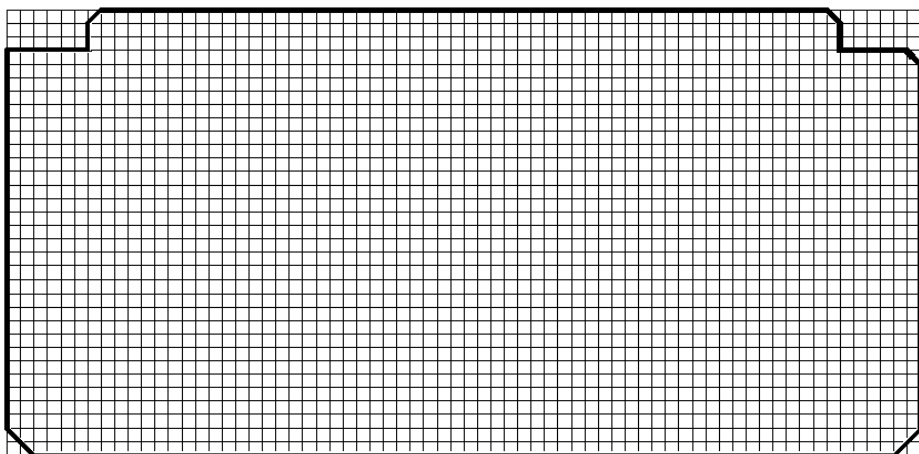


Рис. 3.7. Заготовка печатной платы

Видим, что прокладка более чем одного проводника в зазоре между выводами требует уменьшения диаметра контактной площадки, и в особенности монтажного отверстия, до величины, недостаточной для нормальной уста-

новки штыревых выводов большинства компонентов (микросхемы в DIP-корпусах, резисторы, конденсаторы и др.). Представляется, что данная информация дается только для справки, поскольку приведенные в диалоговом окне размеры можно произвольно изменить, а программа при этом не контролирует, произойдет или нет перекрытие металлизации проводников между собой или с контактными площадками, если задать недопустимые значения.

После выполнения всех этих действий программа переходит в финишное окно и предлагает завершить работу по формированию заготовки щелчком по кнопке **Finish**.

В графическом окне Altium Designer появляется изображение заготовки печатной платы (в нашем случае — прямоугольник размерами 170×110 мм) с наложенной на него сеткой (рис. 3.7).

Поскольку плата в стандарте УБНК1 имеет форму с вырезами и скосами на углах, следует ее доработать. Для этого предусмотрены функции редактирования формы.

3.2.3. Редактирование контуров заготовки

В Altium Designer заложены развитые средства управления геометрией объектов. Не имея возможности подробно рассмотреть все эти средства, ограничимся на данном этапе изложения редактированием контура заготовки печатной платы.

Для формирования контура заготовки УБНК1, установленного по ГОСТ 26765.12-86, следует воспользоваться функцией переопределения контура платы:

1. Щелчком правой кнопки мыши вызвать контекстное меню, активизировать функцию управления шагом сетки захвата **Snap Grid** и установить значение шага сетки в 2,5 мм.
2. Выполнить в главном меню программы цепочку команд: **Design | Board Shape | Redefine Board Shape** (Переопределить контуры платы). Курсор на экране примет вид большого перекрестия, а исходный контур заготовки платы изменит цвет.
3. Щелкая левой кнопкой мыши и перемещая курсор по узлам сетки захвата, создать новый контур заготовки в соответствии с имеющимся чертежом

или эскизом. Удобно, если при этом шаг одной из видимых сеток совпадает с шагом сетки захвата. Щелчком правой кнопки мыши или клавишей <Esc> завершить операцию. Контур заготовки примет новые очертания (на рис. 3.7 прочерчен утолщенной линией).

3.2.4. Слои графического редактора печатной платы

Как и в графических редакторах печатной платы большинства известных САПР, в Altium Designer информация также структурируется по слоям.

Управление структурой слоев осуществляет команда главного меню **Design | Board Layers and Colors**. Открывается одноименное диалоговое окно, в котором представлена структура слоев созданной нами заготовки печатной платы (рис. 3.8).

В Altium Designer существуют слои трех типов:

- ❑ электрические слои (**Electrical Layers**) — проект многослойной печатной платы может содержать до 32 сигнальных слоев (Signal Layers) и до 16 слоев сплошной металлизации для цепей питания и "земли" (Plane Layers). Для добавления, удаления и настройки параметров электрических слоев служит подсистема управления слоями — так называемый менеджер управления слоями Layer Stack Manager;
- ❑ механические слои (**Mechanical Layers**) — 16 слоев общего назначения для размещения на них элементов сборки, обозначений размеров, контура печатной платы, форматки чертежа и др. Состав и свойства механических слоев, применяемых в проекте, настраиваются в диалоге. Перед тем как использовать механический слой, его нужно активировать. Для этого следует:
 - отключить активность опции **Only show enabled mechanical layers** (Показывать только активные механические слои);
 - в поле **Enable** установить флажки активности необходимого числа механических слоев;
 - активировать опцию **Only show enabled mechanical layers**: при этом в диалоговом окне (рис. 3.8) останется видимым список только активизированных слоев.

Видимость механического слоя назначается/отключается в поле **Show**. При активированной опции **Linked to Sheet** объекты, размещенные на механическом слое, подключаются к листу проекта и могут делаться видимыми или невидимыми при управлении видимостью листа проекта;

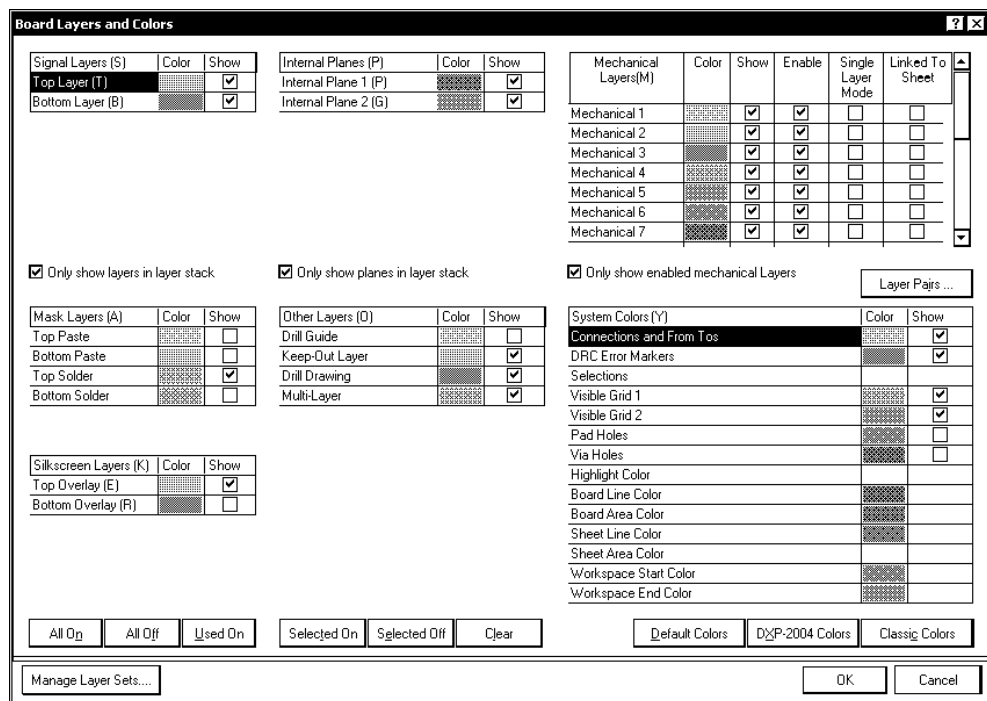


Рис. 3.8. Окно управления видимостью слоев проекта

☐ специальные слои:

- **Top Solder** и **Bottom Solder** — слои защитных масок на верхней и нижней сторонах платы;
- **Top Paste** и **Bottom Paste** — слои трафаретов для нанесения припойной пасты на верхнюю и нижнюю стороны платы;
- **Top Overlay** и **Bottom Overlay** — верхний и нижний слои шелкографии (маркировки и обозначения контуров компонентов);
- **Drill Guide** — слой центров отверстий;

- **Keep-Out Layer** — слой для размещения ограничительных контуров трассировки;
- **Drill Drawing** — слой сверловки;
- **Multi-Layer** — "мультислой" для размещения контактных площадок и переходных отверстий многослойных печатных плат.

Кроме управления свойствами слоев, в диалоговом окне на рис. 3.8 представлены функции управления цветом и видимостью целого ряда объектов, не принадлежащих определенным слоям, но являющихся принадлежностью проекта — сеток, линий электрической связи (**Connections**), маркеров ошибок DRC-контроля, отверстий контактных площадок (**Pad Holes**), отверстий у межслойных переходов (**Via Holes**) и др.

3.2.5. Менеджер структуры слоев

Для управления структурой электрических слоев проекта в Altium Designer есть еще одно наглядное и эффективное средство — так называемый менеджер структуры слоев, вызываемый из контекстного меню. По команде **Options | Layer Stack Manager** откроется одноименное диалоговое окно **Layer Stack Manager** (рис. 3.9).

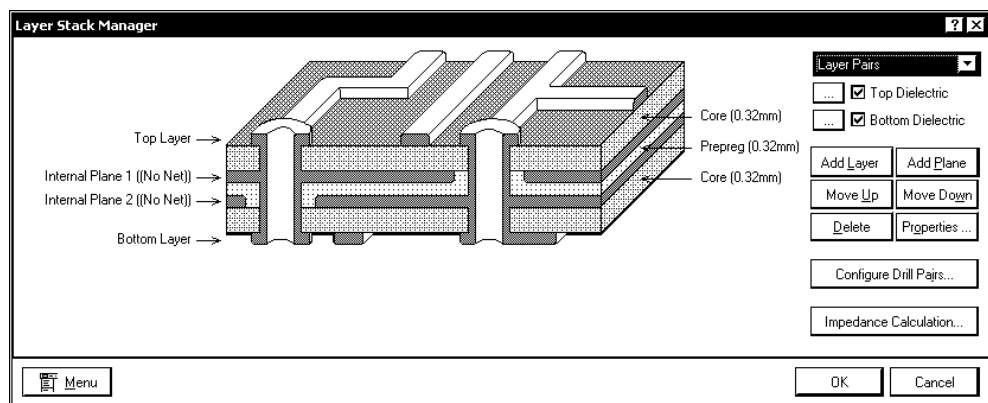


Рис. 3.9. Окно менеджера структуры слоев печатной платы

В левой части окна представлено изометрическое изображение структуры слоев, установленной при формировании заготовки печатной платы. При ука-

зании курсором строки пояснений, относящейся к слою, активизируются функции управления свойствами выбранного слоя:

- ❑ кнопки **Add Layer** и **Add Plane** добавляют новые сигнальные и экранные слои;
- ❑ кнопки **Move Up** и **Move Down** перемещают выбранный слой вверх или вниз, при этом наружные слои **Top Layer** и **Bottom Layer** нельзя переместить во внутреннюю область платы;
- ❑ кнопка **Delete** удаляет выбранный слой из структуры;
- ❑ кнопка **Properties** открывает диалог управления свойствами слоя (типом, толщиной и диэлектрической проницаемостью изоляционных слоев, толщиной фольги проводящих слоев).

Кроме того, менеджер слоев позволяет определить парные слои, соединяемые межслойными переходами (кнопка **Configure Drill Pairs**), а также рассчитать волновые сопротивления печатных проводников. Для этого служит кнопочная команда **Impedance Calculation**.

В открывающемся диалоговом окне приводятся эмпирические расчетные формулы для волнового сопротивления микрополосковой линии, образуемой печатным проводником, расположенным над сплошным экранным слоем и проводником, проходящим между двумя экранными слоями.

3.2.6. Настройка правил проектирования

Заключительный этап подготовки рабочего пространства — настройка правил проектирования. Для этого командой главного меню **Design | Rules** нужно открыть диалоговое окно редактирования правил **PCB Rules and Constraints Editor** (рис. 3.10).

В левой части окна располагается дерево правил проектирования, **Design Rules**. В правой части — список из 26 правил ведения проекта с указанием категории, приоритета, активности, атрибутов (численных значений, логических признаков или словесных описаний).

При двойном щелчке мышью в строке с выбранным правилом или выборе правила в дереве в правой половине окна разворачиваются функции редактирования данного правила (см. рис. 3.10).

Для этой же цели служит подсистема мастера подсказки, вызываемая по команде главного меню **Design | Rule Wizard** (Мастер назначения правил).

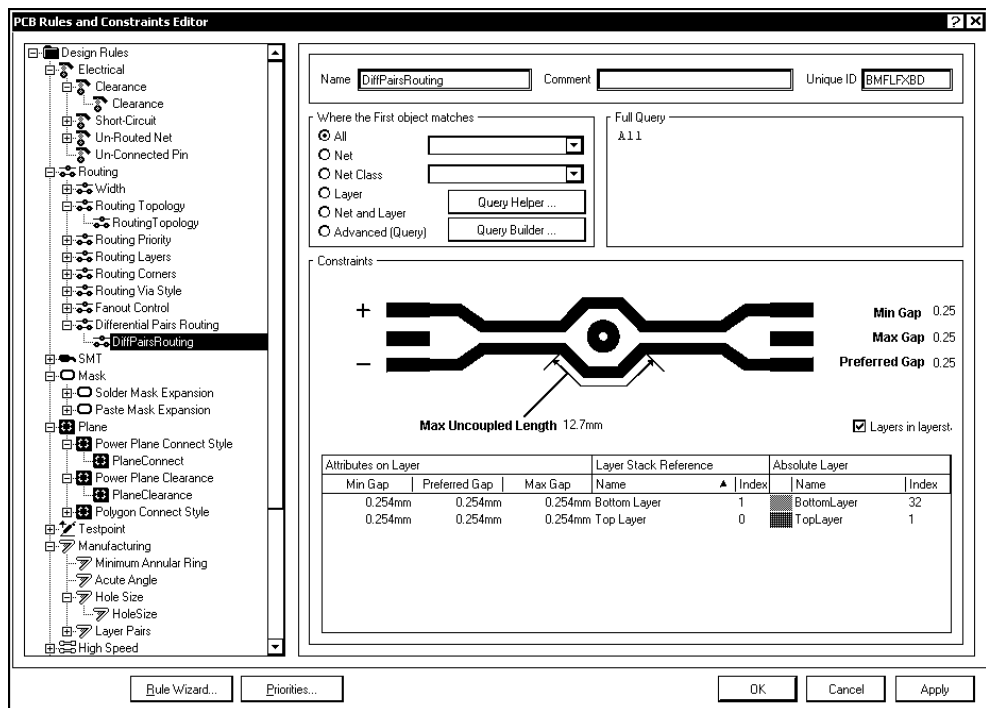


Рис. 3.10. Окно настройки правил ведения проекта

3.2.7. Барьеры трассировки

На плате можно обозначить зоны, в которых запрещается размещение компонентов и печатных проводников. Обычно это место по краям платы, необходимое для крепления рамок, соединителей и т. п. Кроме того, на плате могут быть вырезы, крепежные отверстия, головки винтов, шайбы, гайки и другие области, по которым нельзя проводить печатные проводники.

При использовании в проекте заготовок печатных плат из подкаталога ...Templates пакета запретные зоны, как правило, присутствуют в этих заготовках и импортируются с ними в проект.

При самостоятельном формировании заготовки запретные зоны придется построить. Границы запретных зон можно выполнить с помощью целого ряда графических примитивов — линий, дуг, залитых областей (**Fill**), регионов (**Region**).

Если действие запрета должно распространяться на все слои металлизации, процедура построения будет следующей:

1. Выбрать курсором вкладку **Keep-Out Layer** в нижней части графического окна программы.
2. Активизировать команду **Place | Line** и вычертить замкнутый контур зоны запрета.
3. Клавишей <Esc> или щелчком правой кнопки мыши закончить построение.

Запретная зона, действующая в одном выбранном слое, строится по команде **Place (Разместить) | Keepout (Запретную зону)**:

1. Аналогично предыдущему случаю, выбрать необходимый слой.
2. Активизировать команду **Place | Keepout**, выбрать из выпадающего подменю необходимый графический примитив и прочертить контуры зоны запрета. Запретные зоны, действующие в единственном слое, не экспортируются в выходные файлы фотошаблона и в файлы обменного формата ODB++.

3.2.8. Подключение бланка форматки стандартных листов

Средства Altium Designer позволяют разместить чертеж печатной платы на листе стандартных размеров. Все заготовки, включая бланки форматок чертежных листов, находятся в рабочем каталоге пакета в подкаталоге ...\\Templates. Среди них есть бланки формата A0...A4 в стандарте ISO. Графление этих бланков не соответствует ЕСКД, поэтому для формирования документов по ЕСКД приходится создавать специальные бланки. Собственный бланк форматки можно построить на одном из механических слоев аналогично тому, как строилась форматка для листа электрической схемы.

Подключение форматок из каталога заготовок выполняется следующим образом:

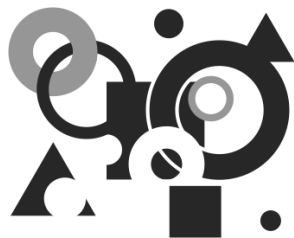
1. Открыть РСВ-файл проекта. Убедиться, что командой меню **View | Sheet** или "горячими" клавишами <V> и <H> включена видимость листа, на который предполагается наложить форматку.
2. Вызвать панель **Files** (<Ctrl>+<N>) и в поле **New From Templates** активизировать команду **PCB Templates**.

3. Выбрать из предлагаемого списка необходимую заготовку, например a3.PcbDoc. Форматка открывается как обычный документ Altium Designer.
4. Селектировать все объекты этого документа (<Ctrl>+<A>) и скопировать в буфер обмена Windows (**Edit | Copy** или <Ctrl>+<C>).
5. Переключиться в окно проекта и присоединить содержимое буфера обмена к проекту (**Edit|Paste** или <Ctrl>+<V>). Бланк форматки размещается на слое **Mechanical16**.
6. Активизировать управление слоями проекта (**Design | Board Layers and Colors**) и установить для этого слоя признаки видимости, активности и присоединения к листу — **Show** (Показать), **Enable** (Установить активность) и **Linked to Sheet** (Связать с листом).
7. Установить также для слоя форматки признак **Single Layer Mode**, чтобы форматка оставалась видимой поверх всех остальных слоев независимо от активности функции просмотра выбранного слоя. Кнопкой **OK** завершить настройку.

Следует отметить, что обрамление интегрального PCB-образа печатной платы форматкой не делает результат этой операции конструкторским документом в том смысле, который придается этому термину стандартами ЕСКД. Стандартные средства Altium Designer, как и многих других "электронных" САПР иностранного происхождения, не дают возможности выполнить полноценные конструкторские документы по ЕСКД:

- ☐ невозможно получить в едином документе проекции печатной платы по правилам машиностроительного черчения в соответствии с требованиями ГОСТ 2.305-68 и ГОСТ 2.417-91;
- ☐ невозможно получить зеркальное изображение платы (вид снизу);
- ☐ при распечатке документа возможно лишь одновременное масштабирование как изображения печатной платы, так и форматки, что не допускается по ЕСКД;
- ☐ средства простановки размеров не дают возможности нанести размеры и предельные отклонения по ГОСТ 2.307-68.

Поэтому для формирования конструкторских документов на печатную плату и функциональный узел (модуль) на ее основе целесообразно конвертировать проект Altium Designer в формат "машиностроительных" САПР, например, AutoCAD, Autodesk Inventor или отечественной системы КОМПАС.



Формирование и редактирование электрической схемы

Считаем, что библиотеки компонентной базы для выполнения проекта подготовлены и включены в рабочую среду Altium Designer, все подготовительные операции, рассмотренные в *разд. 3.1*, выполнены:

- ☐ сформирован или открыт из каталога шаблонов лист схемного редактора стандартного формата по ЕСКД;
- ☐ заполнены реквизиты схемного документа;
- ☐ установлена метрическая система единиц измерения;
- ☐ настроен набор сеток проектирования.

Кроме перечисленных настроек целесообразно установить удобный для работы вид курсора. Для этого активизировать команду главного меню **Tools | Schematic Preferences**, выбрать в диалоговом окне **Schematic Preferences** опцию **Schematic | Graphical Editing | Cursor Type** и указать один из видов курсора:

- ☐ **Large Cursor 90** — перекрестие через весь экран, под углом 90°;
- ☐ **Small Cursor 90** — малое перекрестие, под углом 90°;
- ☐ **Small Cursor 45** — малое перекрестие, наклоненное под 45°;
- ☐ **Tiny Cursor 45** — мелкое перекрестие, наклоненное под 45°.

4.1. Размещение объектов на поле чертежа

Altium Designer делит объекты, находящиеся на поле электрической схемы, на графические и электрические.

Графические объекты:

- ☐ линия (Line);
- ☐ дуга, эллиптическая дуга (Arc, Elliptical Arc);
- ☐ эллипс, в частности окружность (Ellipse);
- ☐ сплайн-кривая (Bezier);
- ☐ прямоугольник, скругленный прямоугольник (Rectangle, Rounded Rectangle);
- ☐ многоугольник (Polygon);
- ☐ секторная диаграмма (Pie Chart);
- ☐ график (Graphic).

Для размещения графических объектов используется команда главного меню **Place | Drawing Tools** с соответствующими подкомандами, смысл которых ясен из их имен и сопровождающих пиктограмм.

Аналогичный результат достигается при активизации щелчком правой кнопки мыши контекстного меню и указания в нем той же команды **Place | Drawing Tools** с ее подкомандами. Для размещения графических объектов также можно выбрать соответствующую кнопку-пиктограмму в плавающем меню **Utilities | Utility Tools**.

Электрические объекты:

- ☐ схемные компоненты (Schematic Component) с их электрическими выводами;
- ☐ линии электрической связи (Wire — проводник);
- ☐ линии групповой связи (Bus);
- ☐ идентификаторы цепей (Net Identifiers).

4.1.1. Размещение схемных символов

1. Активизировать команду главного меню программы **Place | Part** (Разместить | Логическую секцию). Можно также воспользоваться контекст-

ным меню или "горячими" клавишами <P>, <P>. Откроется диалоговое окно **Place Part** (рис. 4.1) с информацией о последнем введенном ранее компоненте.

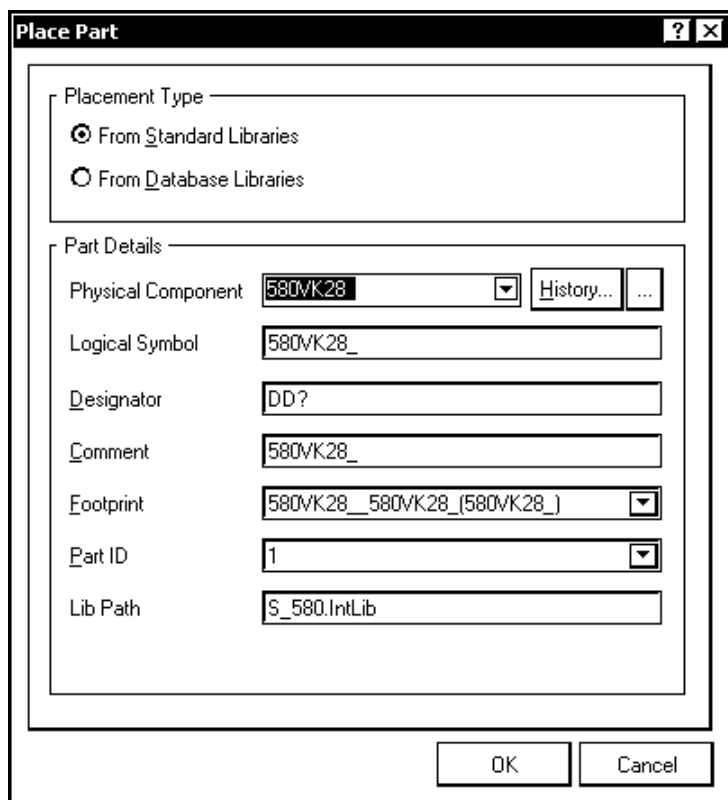


Рис. 4.1. Диалог присоединения компонента к схемному документу

2. В поле **Placement Type** указать область поиска компонента:
 - **From Standard Libraries** — в стандартных библиотеках (инсталлированных в рабочую среду Altium Designer);
 - **From Database Libraries** — в библиотеках баз данных.
3. В поле **Part Details** отображаются сведения о компоненте:
 - **Physical Component** — имя (тип) "физического" компонента;
 - **Logical Symbol** — имя логического символа (может отличаться от имени компонента, в особенности у многосекционных и гетерогенных);

- **Designator** — позиционное обозначение (префикс по ГОСТ 2.710-81 или по другому стандарту), а также знак вопроса "?" — для порядкового номера, назначаемого в схеме;
- **Comment** — текстовый комментарий;
- **Footprint** — имя посадочного места;
- **Part ID** — обозначение секции (для многосекционных);
- **Lib Path** — путь к библиотеке.

Отметим, что редактирование текста во всех этих полях не передается компоненту — для этого есть отдельная операция, речь о которой впереди.

Кнопка **History** вызывает диалоговое окно "истории" **Placed Parts History** (рис. 4.2) со списком ранее вызывавшихся компонентов. Указав компонент в списке, можно вызвать его повторную установку в схему.

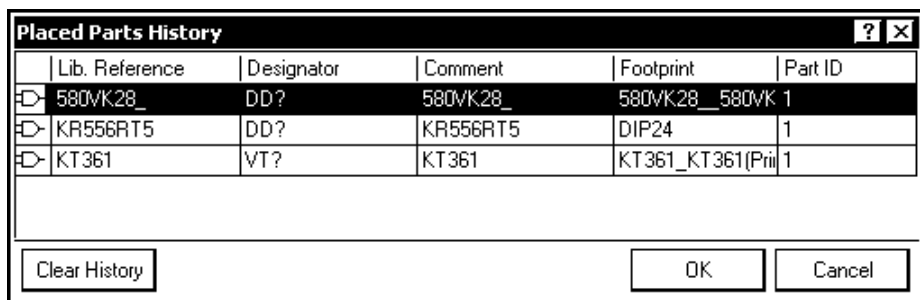


Рис. 4.2. История присоединения компонентов

- Для поиска новых компонентов кнопкой активизировать диалоговое окно поиска библиотек **Browse Libraries** (рис. 4.3), в которых хранится нужный компонент.

В полях окна (рис. 4.3) сгруппированы команды управления поиском и отображается информация о компонентах выбранной библиотеки:

- в поле **Libraries** открывается окно со списком установленных библиотек;
- кнопка активизирует окно (см. рис. 2.25), позволяющее включить в рабочую среду программы необходимые библиотеки или исключить ненужные;

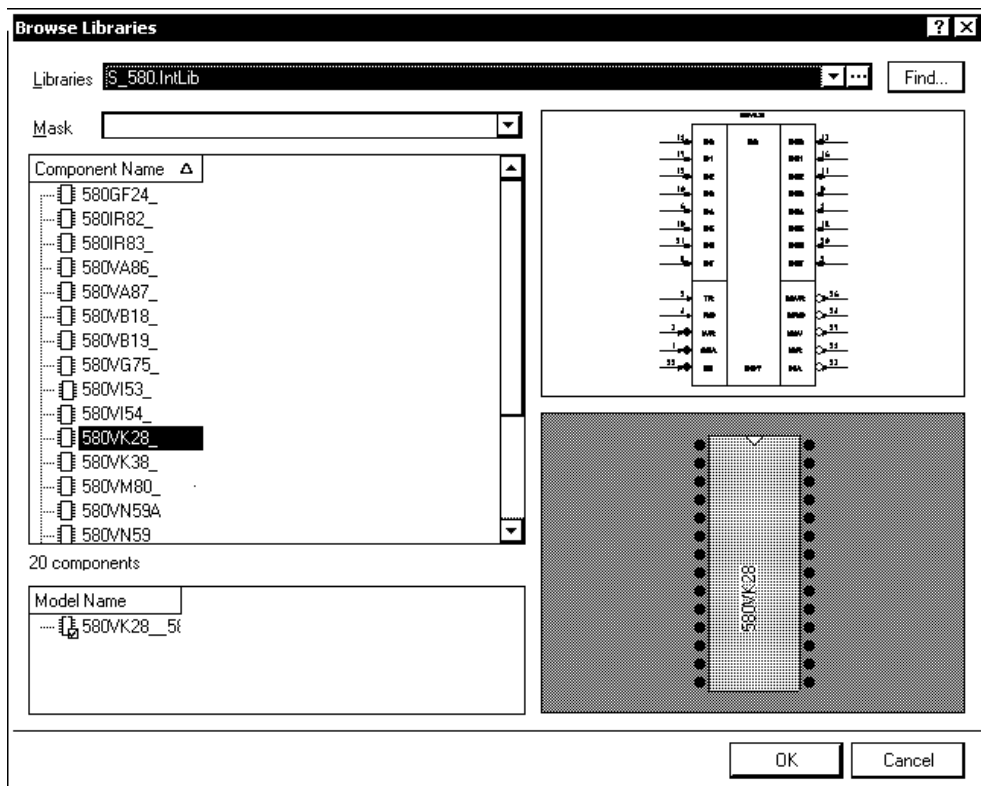


Рис. 4.3. Окно просмотра библиотек

- кнопка **Find** (Найти) открывает диалоговое окно поиска компонента по маске (см. рис. 2.26);
- в поле **Component Name** отображается дерево структуры выбранной библиотеки с условными обозначениями и именами входящих компонентов;
- в поле **Model Name** отображается дерево моделей выбранного компонента (TPIM, SPICE-модель и др.);
- в полях предварительного просмотра отображается УГО схемного компонента и его посадочное место (ТПМ).

5. Завершить выбор компонента кнопкой **OK**.

6. Перед установкой выбранного компонента необходимо отредактировать его позиционное обозначение и указать номер или другое обозначение

секции (в случае многосекционного компонента). Для этой цели служит процедура "горячего" редактирования (Re-entrant Editing), активируемая клавишей <Tab>, открывающей диалоговое окно свойств компонента **Component Properties** (рис. 4.4).

Применительно к текущей задаче следует:

- в поле **Designator** ввести вместо знака вопроса порядковый номер позиционного обозначения;
- кнопками "стрелка влево" или "стрелка вправо" выбрать секцию в случае многосекционного компонента;
- завершить редактирование нажатием кнопки **OK**.

7. Выбранный компонент (его схемный символ) "плывет" вместе с курсором по листу схемы. При этом нажатием клавиши <Пробел> (<Space>) возможен разворот символа на углы, кратные 90°.

Component Properties

Properties

Designator: ☒ Visible ☐ Locked

Comment: ☐ Visible

Description:

Unique Id:

Type:

Library Link - Integrated Component

Library:

Physical Component:

Logical Symbol:

Sub-Design Links

Sub-Project:

Configuration:

Graphical

Location X: Y:

Orientation: ☐ Mirrored

Mode: ☒ Lock Pins

☐ Show All Pins On Sheet (Even if Hidden)

☐ Local Colors

Parameters for DD? - 580VK28

Visible	Name	Value	Type
<input type="checkbox"/>	PDIF_PID	580VK28	STRING
<input type="checkbox"/>	PDIF_TY	256	STRING
<input checked="" type="checkbox"/>	PWGD	(14=gn d,28=+5v)	STRING
<input type="checkbox"/>	Value	*	STRING

Models for DD? - 580VK28


Name	Type	Description
580VK28_580VK28	Footprint	

Рис. 4.4. Окно редактирования свойств схемного компонента

8. Щелчком левой кнопки мыши или клавишей <Enter> зафиксировать компонент на листе схемы. Программа при этом останется в режиме расстановки компонентов. Выбранный компонент можно зафиксировать на листе схемы в нужном количестве копий. При этом инкрементируются номера в позиционных обозначениях компонентов и обозначения секций.

4.1.2. Черчение линий электрической связи

Прокладка проводников

Черчение линий электрической связи (проводников) выполняется по команде главного меню **Place** (Разместить) | **Wire** (Проводник), либо той же командой из контекстного меню, вызываемого щелчком правой кнопки мыши или щелчком на одноименной кнопке **Place Wire**  в линейке инструментов **Wiring**.

Курсор приобретает вид мелкого (Tiny) перекрестия под углом 45° (есть еще малое перекрестие — Small, большего размера, чем Tiny).

Прокладку проводника начинают щелчком мышью в узле сетки захвата **Snap Grid**. Двигая курсор и нажимая левую кнопку мыши в точках поворота линии связи, прокладывают сегменты проводника до точки назначения.

Начало и конец проводника должны попадать в точки присоединения электрических объектов — на электрические выводы компонентов, контакты электрического соединителя, в точку соединения с другим проводником и т. п. При активизированной электрической сетке (клавиши <Shift>+<E>) проводник и другой электрический объект, оказывающийся от него на расстоянии менее одного шага электрической сетки, автоматически соединяются друг с другом. При этом программа обозначает факт соединения "горячей точкой" (Hot Spot) — малым косым перекрестием красного цвета. Щелчком правой кнопки мыши в этой точке или нажатием клавиши <Esc> прокладку проводника завершают.

Пока прокладка проводника не закончена, последний из проложенных сегментов можно удалить нажатием клавиши <Backspace> (Шаг назад).

Прокладка сегментов проводника, который пока не подключается к другому электрическому объекту, завершается также щелчком правой кнопки мыши или клавишей <Esc>.

Повторный щелчок правой кнопки мыши или нажатие клавиши <Esc> завершает работу программы в режиме черчения линий электрической связи.

Формирование изломов проводника

В ходе черчения линии электрической связи с помощью комбинации "горячих" клавиш <Shift>+<Space> можно установить форму изломов проводника:

- ☐ **90 Degree** — изломы под прямым углом;
- ☐ **45 Degree** — изломы под углом 45°;
- ☐ **Any Angle** — прокладка под любым углом (но через узлы текущей активной сетки Snap Grid);
- ☐ **Auto Wire** — автопрокладка: в этом режиме указываются курсором (со щелчком левой кнопкой мыши) начальная и конечная точки линии электрической связи, и программа автоматически прокладывает проводник, обходя препятствия — контуры УГО и электрические выводы компонентов.

Режимы формирования изломов перебираются циклически. Информация о текущем режиме отображается в строке сообщений.

Выполнение схемы по ЕСКД предполагает формирование изломов под прямым углом (требование ГОСТ 2.702-75). Определенную пользу может принести автопрокладка, т. к. в этом режиме изломы также формируются под прямым углом.

Соединение и пересечение проводников

Новый проводник соединяют с существующим щелчком левой кнопки мыши в месте их пересечения, где автоматически ставится точка. По команде **Tools | Schematic Preferences | Schematic | Compiler** в полях диалогового окна **Auto-Junctions** и **Manual Junctions Connection Status** устанавливают цвет и размер точки.

Если в месте пересечения проводников щелчок кнопкой мыши не делать, проводники не соединяются. Если в диалоге **Preferences** активизирована опция **Schematic | General | Display Cross-Overs**, то пересечения показываются с огибанием одного проводника другим. Такое обозначение в документах по ЕСКД недопустимо.

4.1.3. Имена и признаки связности цепей

При выполнении электрической связи программа автоматически присваивает цепи имя, которое состоит из слова NET и примыкающего к нему обозначе-

ния контакта одного из компонентов, присоединенных к цепи, например NETDA2-6.

Всем сегментам проводников, не присоединенных ни одним концом к компонентам, программа присваивает имя N000-1.

Простейший и очевидный признак связности сегментов, образующих линию электрической связи, — примыкание сегментов друг к другу без разрывов и точка в месте соединения проводника, подходящего к другому под прямым углом.

Средства многих САПР, в том числе и Altium Designer, предоставляют развитые способы обозначения принадлежности сегментов линии электрической связи к одной электрической цепи без их непосредственного примыкания. Они аналогичны способам, применяемым при черчении схем вручную, и основаны на использовании так называемых идентификаторов цепей.

В зависимости от сложности проекта и от видов вхождения одних документов проекта в другие проект в Altium Designer и, соответственно, схемные документы проекта могут быть представлены единственным листом схемы, а могут образовывать многолистовую одноуровневую ("горизонтальную") структуру или иерархическую вертикально-связанную структуру.

В PCB-проекте действует функция автоматического определения структуры проекта и устанавливается так называемый горизонтальный или вертикальный охват (Scope) проекта. От этого зависит, как принятая в Altium Designer система идентификаторов цепей устанавливает связность цепей на одном листе, связность цепей в многолистовых проектах по горизонтали и по вертикали.

Далее ограничимся рассмотрением действия идентификаторов цепей в однолистовом и многолистовом одноуровневом проектах.

Всего существует семь идентификаторов:

- ❑ **Net Label** — метка цепи. Объединяет в одну цепь все сегменты, имеющие совпадающее обозначение на одном листе и в одноуровневом (горизонтальном) проекте;
- ❑ **Port** — метка с более высоким приоритетом. При одновременном присутствии на листах схемы меток **Label** и меток **Port** последним автоматически присваивается статус глобальных, и только они объединяют сегменты связей на листах схем проекта в глобальную цепь;

- ❑ **Power Port** — метки цепей питания и "земли", всего 11 разновидностей (рис. 4.5). Объединяют в глобальную цепь сегменты с одинаковой меткой во всем проекте;
- ❑ **Hidden Pin** — скрытые выводы компонентов. Связываются в единую глобальную цепь с именем, обозначенным в поле **Connect to** в диалоге управления свойствами контактов компонента (см. рис. 2.5);
- ❑ **Sheet Entry** — соединитель листов в иерархическом проекте;
- ❑ **Off-sheet Connector** — соединитель листов в многолистовом одноуровневом проекте.

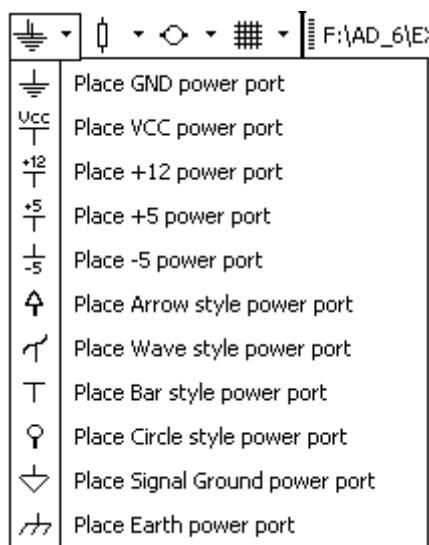


Рис. 4.5. Варианты меток цепей питания

Следует отметить, что у меток **Net Label** и **Port** на разных цепях в одном документе могут оказаться совпадающие имена. Общая цепь образуется только в том случае, если сегменты с такими метками явно соединить проводником.

Для присвоения имени цепи с помощью метки **Net Label** нужно активизировать в главном или контекстном меню команду **Place | Net Label** и далее клавишей <Tab> вызвать диалог редактирования свойств метки (рис. 4.6).

В поле **Properties** ввести имя метки (цепи) или выбрать из выпадающего списка одно из существующих имен, если сегмент проводника, отмечаемый меткой, присоединяется к существующей цепи.

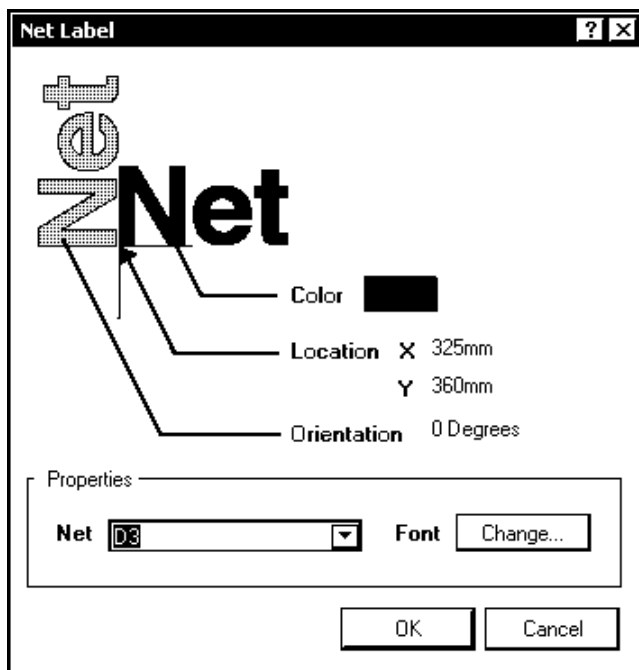


Рис. 4.6. Окно редактирования меток цепей

Для выполнения надписи, содержащей знак инверсии, после каждого символа надписи следует ввести символ обратной косой черты.

Функции настройки свойств метки позволяют также выбрать цвет надписи, ориентацию, гарнитуру и размер шрифта надписи.

Для присвоения имени цепи с помощью метки **Port** необходимо активизировать команду главного меню или всплывающего контекстного меню **Place | Port**, затем клавишей <Tab> вызвать диалог настройки свойств порта. Откроется диалоговое окно **Port Properties** (рис. 4.7).

В полях диалогового окна отредактировать элементы графического изображения и другие свойства метки:

□ в поле **Properties**:

- ввести или выбрать из выпадающего списка имя цепи (**Name**);
- выбрать из выпадающего списка **I/O Type** тип сигнала, проходящего по цепи, — входной (**Input**), выходной (**Output**), двунаправленный (**Bidirectional**) или неопределенный (**Unspecified**);

- в верхней части панели-вкладки **Graphical** установить:
- **Alignment** — расположение надписи в поле метки: слева, справа или по центру;
 - **Text Color** — цвет текста надписи;
 - **Location** — точку геометрической (но не электрической привязки) контура метки;
 - **Width** — ширину поля метки;
 - **Fill Color** — цвет заполнения поля метки;
 - **Border Color** — цвет границы контура метки;
 - **Style** — стиль выполнения заостренных "носиков" поля метки.

Стиль выбирают в выпадающем списке, предусматривающем 8 возможных вариантов: 4 для горизонтального и 4 для вертикального расположения поля метки.

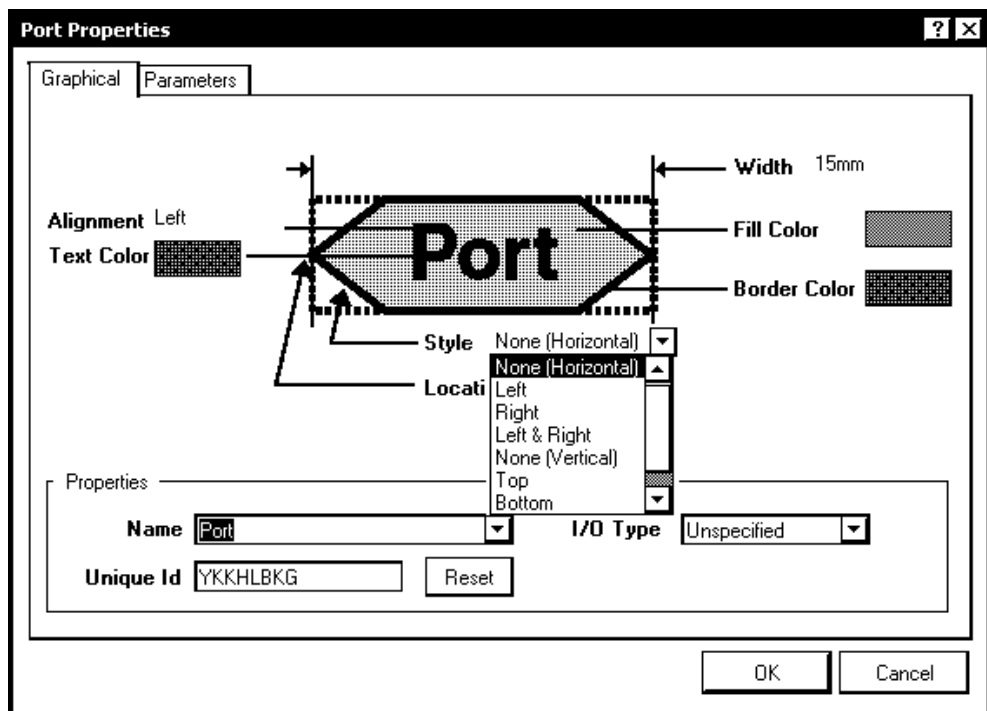


Рис. 4.7. Окно редактирования меток-портов

При любом стиле и обозначении типа проходящего по цепи сигнала у метки-порта программа образует две точки электрического присоединения метки к проводнику. Метку можно присоединить к проводнику любой из этих двух точек, а также и обеими точками. В последнем случае поле метки ложится поверх проводника.

При этом назначенный стиль и тип электрической связи образуют сложную систему приоритетов, влияющую на окончательное изображение метки-порта на схеме.

Предоставляем читателю самостоятельно изучить все возможные комбинации; здесь же остановимся на сочетаниях, позволяющих получить осмысленное изображение меток-портов на схеме:

- ❑ при подключении метки с обозначением цепи **Input** одной точкой перпендикулярно проводнику или к концу проводника заостренный конец поля метки обращен к проводнику, независимо от настройки стиля;
- ❑ при подключении метки с обозначением цепи **Input** двумя точками поверх проводника поле метки имеет оба заостренных конца, также независимо от настройки стиля;
- ❑ при подключении метки с обозначением цепи **Output** одной точкой перпендикулярно проводнику или к концу проводника заостренный конец поля метки обращен от проводника, независимо от настройки стиля;
- ❑ при подключении метки с обозначением цепи **Output** двумя точками поверх проводника поле метки имеет оба тупых конца, независимо от настройки стиля;
- ❑ при подключении метки с обозначением цепи **Bidirectional** как одной, так и двумя точками поле метки имеет оба заостренных конца, независимо от настройки стиля;
- ❑ при подключении метки с необозначенным типом цепи (**Unspecified**) приоритетом обладает стиль настройки поля.

Отметим, что размещение поля метки-порта вдоль (поверх) проводника противоречит требованиям ЕСКД, в соответствии с которыми обозначения имен цепей не должны лежать поверх линий связи. Кроме того, в месте присоединения порта к проводнику ставится точка, обозначающая электрическое соединение, что также не соответствует правилам выполнения электрических схем по ЕСКД. Исключение составляет подключение метки порта к концу проводника.

Указание белого цвета заливки и границ поля метки порта делает поле невидимым на белом листе схемы, однако поле не становится от этого прозрачным: при расположении метки вдоль проводника порт визуально воспринимается как разрыв линии связи с точками электрического соединения на концах видимого разрыва.

4.1.4. Линии групповой связи

Линии групповой связи выполняются при составлении электрической схемы в двух случаях:

- когда представляется целесообразным объединить в одну условную общую линию группу функционально однородных электрических связей. В терминах Altium Designer такая линия групповой связи называется логической (**Logical Bus**);
- когда сведение в одну линию групповой связи множества разнородных цепей освобождает схему от путаницы многочисленных связей и улучшает читаемость схемы. Такая линия групповой связи называется в Altium Designer графической (**Graphical Bus**).

Для формирования линии групповой связи обоих типов нужно сначала вызвать команду главного или контекстного меню **Place** (Разместить) | **Bus** (Линию групповой связи), затем клавишей <Tab> активизировать процедуру "горячего" редактирования свойств линии групповой связи и в диалоговом окне **Bus** (рис. 4.8) указать ширину и цвет линии.

После этого необходимо прочертить линию групповой связи на листе схемы. Сама эта линия не образует электрическую цепь, а служит только для логического или визуального (графического) объединения индивидуальных электрических связей.

Следующий шаг — черчение индивидуальных линий, входящих в групповую, и назначение им необходимых признаков связи.

Предварительно вдоль линии групповой связи командой **Place** | **Bus Entry** можно расставить входы в групповую линию — прямолинейные отрезки проводника, наклоненные под 45° . Назначение этих входов — исключить контакт проводников, подходящих к групповой линии с противоположных сторон, торец в торец. Если допустить такой контакт, проводники сливаются в одну цепь. Такое изображение линий групповой связи применялось в отечественной документации до ввода в действие ЕСКД. По ЕСКД требуется подвод индивидуальных проводников к групповой линии под прямым углом,

без излома. Ничто не мешает выполнить линии групповой связи таким образом и в Altium Designer. Нужно только так смещать проводники, подходящие к линии групповой связи с противоположных сторон, чтобы они не сомкнулись концами.

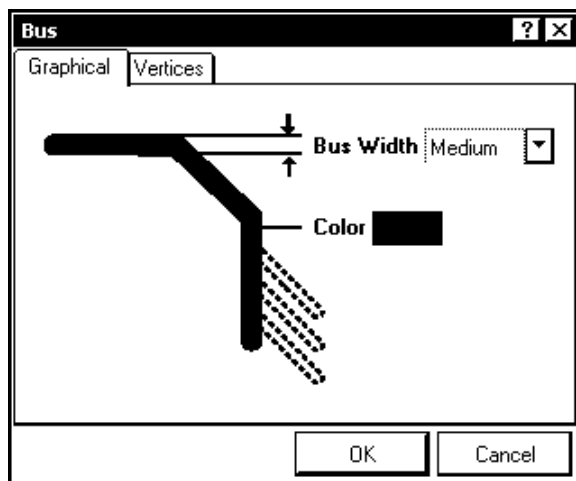


Рис. 4.8. Окно редактирования линии групповой связи

Связность сегментов цепей, входящих в линию групповой связи и выходящих из нее, достигается присвоением им единого имени с помощью идентификаторов **Net Label**.

В случае логической линии групповой связи линия в целом наделяется меткой **Net Label** в формате, например, $D[0..7]$ (рис. 4.9, а), а принадлежащие ей индивидуальные проводники — метками в формате $D0$, $D1$, $D2$ и т. д. Префикс D является признаком, объединяющим проводники в логическую линию групповой связи. Как и у индивидуальных электрических связей, идентификаторы линии групповой связи разных типов не объединяются автоматически даже при совпадении имен. Чтобы метки **NetLabel** и **Port** одинаково идентифицировали линию групповой связи $D[0..7]$, к ней должны быть присоединены обе эти метки. При переходе такой линии групповой связи с листа на лист одноуровневого проекта связность устанавливается с помощью идентификатора **Port** с тем же именем $D[0..7]$.

В случае графической линии групповой связи (рис. 4.9, б) индивидуальным проводникам, входящим и выходящим из групповой линии, могут присваи-

ваться как произвольные, так и структурированные имена (также меткой **Net Label**), а самой групповой линии никакого имени не присваивается. Самую групповую линию можно вообще удалить с чертежа без ущерба для связности цепей как в пределах одного листа, так и в многолистовом одноуровневом проекте. Таким образом, изображение линии-жгута служит лишь для улучшения визуального восприятия и чтения схемы человеком.

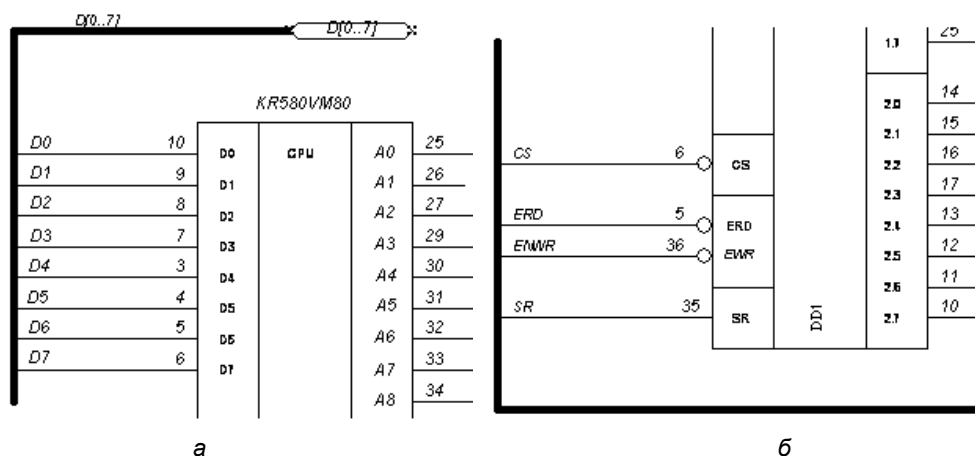


Рис. 4.9. Варианты линии групповой связи:
а — "логическая" линия; б — "графическая" линия

4.2. Редактирование электрической схемы

Редактирование любых объектов, размещенных на листе схемы, начинается с селекции объекта одним из двух способов:

- ☐ указать на объект курсором и нажать левую кнопку мыши;
- ☐ нажав и не отпуская левую кнопку мыши, окружить объект или группу объектов рамкой, после чего сделать щелчок мышью.

Селектированный объект отмечается специальными знаками-прищепками (Grips): компоненты — по контуру; линии, проводники — на каждом изломе.

4.2.1. Редактирование проводников

Перемещение изломов проводника

Навести курсор на излом селектированного проводника, отмеченный знаком-прищепкой, нажать левую кнопку мыши и, не отпуская ее, переместить отмеченный излом по листу. В точке назначения отпустить кнопку мыши. Излом фиксируется в новом положении. При этом может нарушиться ортогональность сегментов проводника или линии групповой связи.

Перемещение сегментов проводника

Навести курсор на середину сегмента выделенного проводника и, аналогично предыдущему случаю, переместить сегмент. Здесь возможны два варианта:

- ☐ при перемещении курсора в поперечном направлении за курсором перемещается только выбранный сегмент, а примыкающие к нему два сегмента деформируются, следуя за его концами;
- ☐ при перемещении курсора в продольном направлении за курсором перемещаются три сегмента — выбранный и два примыкающие к нему. Прямоугольность изломов не нарушается.

Продолжение проводника

Навести курсор на конец селектированного проводника, отмеченный прищепкой, и переместить курсор по листу, удерживая нажатой левую кнопку мыши. Проводник вытягивается в направлении перемещения курсора. Для продолжения проводника в исходном направлении следует перемещать мышь, удерживая нажатой клавишу <Alt>.

Разрыв проводника

По команде **Edit | Break** (Разрыв) на экран выводится рамка-шаблон, которую следует поместить на проводник. Рамка автоматически разворачивается вдоль проводника. Длину вырываемого из проводника участка устанавливают в "горячем" диалоге, вызываемом клавишей <Tab>. В диалоговом окне можно выбрать один из трех вариантов: разрыв длиной в целый сегмент, разрыв длиной в один шаг активной сетки и разрыв длиной в несколько шагов сетки. Разрыв делается щелчком левой кнопки мыши.

4.2.2. Перемещение компонентов

Программа предусматривает два способа перемещения схемных компонентов:

- ❑ **Move** — перемещение только компонента, без подведенных к нему электрических связей;
- ❑ **Drag** — перемещение компонента вместе с подключенными к нему электрическими связями.

Оба вида перемещений, а также еще 16 команд управления положением объектов на листе схемы активизируются из главного меню **Edit | Move** (рис. 4.10).

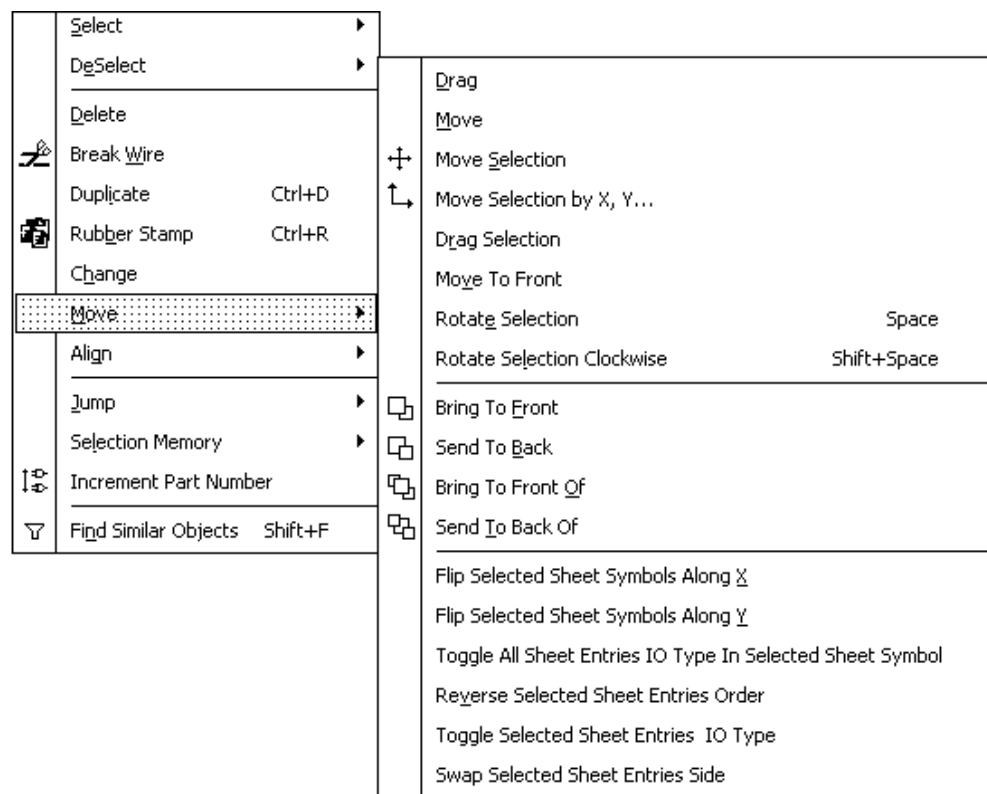


Рис. 4.10. Команды редактирования схемного документа

Альтернативный способ — перемещение компонента непосредственно вслед за курсором: указать курсором компонент, нажать левую кнопку мыши и, не

отпуская ее, перемещать компонент по листу; щелчком левой кнопки мыши в месте назначения зафиксировать объект.

Команды меню **Edit | Move | Move** и **Edit | Move | Drag** и непосредственное перемещение курсором выполняются по-разному. В то время как непосредственное перемещение действует только на компонент, выбранный в текущий момент, и заканчивается вместе с фиксацией компонента в новом положении, упомянутые команды меню переводят программу в режим перемещений (**Move Mode** или **Drag Mode**). Перемещение компонента, на который указывает курсор, начинается по щелчку (не по нажатию) левой кнопки мыши или по нажатию клавиши <Enter> и завершается повторным щелчком или нажатием клавиши <Enter> в точке назначения. Программа остается в режиме перемещений. Выход из режима перемещений происходит по нажатию клавиши <Esc> или по щелчку правой кнопки мыши.

Можно назначить активной только функцию перемещения **Drag**. Для этого в диалоге **Tools | Schematic Preferences** активизировать опцию **Schematic | Graphical Editing | Always Drag**. В этом случае при непосредственном управлении курсором компонент перемещается вместе с подключенными к его выводам электрическими связями. При этом, однако, команда **Edit | Move | Move** обладает приоритетом перед настройкой **Always Drag**: при активизации этой команды компоненты перемещаются, а связи от них отрываются и остаются на месте.

В процессе перемещения компонента над ним можно выполнить ряд дополнительных действий.

Действия при перемещении без связей (**Move**):

- ☐ нажатие клавиши <Пробел> поворачивает компонент на 90° против часовой стрелки; клавиши <Shift>+<Пробел> поворачивают компонент по часовой стрелке;
- ☐ нажатие клавиш <X> или <Y> отражает компонент зеркально относительно горизонтальной или вертикальной оси соответственно;
- ☐ при нажатой клавише <Alt> перемещение ограничивается горизонтальным или вертикальным направлением.

При перемещении с подключенными связями (**Drag**):

- ☐ нажатие клавиш <Пробел> или <Shift>+<Пробел> переключает режим ортогональности проводников;
- ☐ нажатие клавиш <Ctrl>+<Пробел> разворачивает компонент на 90° против часовой стрелки;

- ❑ нажатие клавиш <X> или <Y> отражает компонент зеркально относительно горизонтальной или вертикальной оси соответственно;
- ❑ при нажатой клавише <Alt> перемещение ограничивается горизонтальным или вертикальным направлением.

4.2.3. Копирование и вставка копий

Расположенные на листе схемы объекты можно скопировать в буфер обмена Windows и вставить из буфера обмена на лист или в другие документы. Для копирования следует активизировать стандартную команду главного меню программы **Edit | Copy** (<Ctrl>+<C>) и указать курсором опорную точку, за которую объект будет извлекаться из буфера командой **Edit | Paste** (<Ctrl>+<V>).

Кроме того, программа располагает функцией **Smart Paste** — вставки объектов из буфера обмена с трансформацией объектов одного типа в объекты другого типа, например меток цепей **Net Label** в метки-порты и т. п.

4.2.4. Присвоение позиционных обозначений

Позиционное обозначение (Designator) присваивается компонентам при их первичном помещении на лист схемы. Префиксы позиционных обозначений, назначенные при формировании библиотек, сохраняются при передаче компонентов в схему. Если перед постановкой определенного компонента на лист в диалоге "горячего" редактирования (активизируется клавишей <Tab>) свойств позиционному обозначению присвоен номер, следующие компоненты данного типа нумеруются с инкрементированием. При этом маловероятно, что позиционные обозначения в схеме, в окончательном ее виде, будут располагаться в порядке, определенном в действующих стандартах.

Для приведения позиционных обозначений в соответствие с требованиями ЕСКД служит функция **Annotation**. Порядок выполнения этой операции следующий:

1. Командой главного меню **Tools | Annotate** открыть диалоговое окно **Annotation** (рис. 4.11).

В двух колонках таблицы **Proposed Changes List** (Список предполагаемых изменений) представлены существующие позиционные обозначения (колонка **Current**) и не назначенные пока новые (колонка **Proposed**).

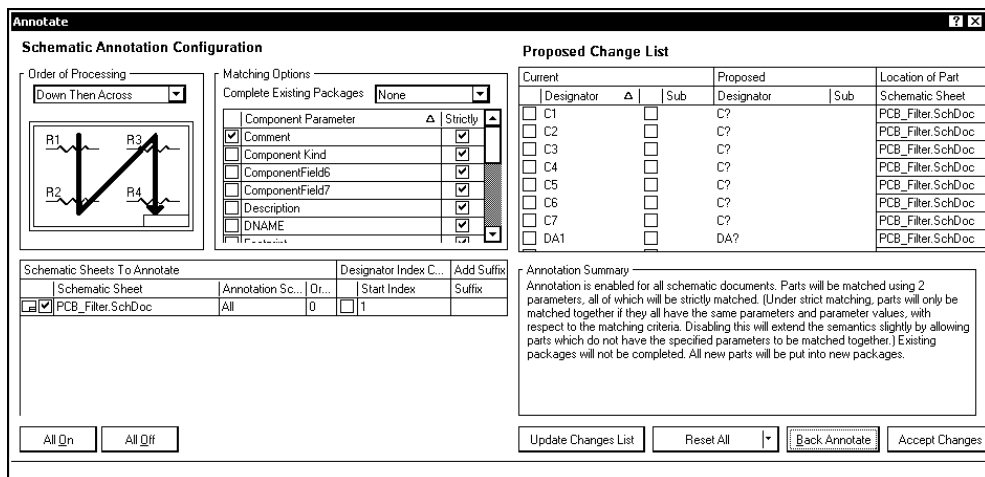


Рис. 4.11. Диалог перенумерации элементов схемы

- В поле **Order of Processing** (Порядок обработки) выбрать из выпадающего списка порядок простановки позиционных обозначений. По ЕСКД это **Down Then Across** — вертикальными колонками слева направо. Выбранный порядок иллюстрирует диаграмма.
- В поле **Schematic Sheets To Annotate** указать документ, в котором выполняется перенумерация.
- В поле **Matching Options** указать или сбросить признаки группировки логических секций в корпус компонента.
- При щелчке мышью на кнопке **Update Changes List** (Обновить список изменений) программа выдаст окно с сообщением о числе вносимых изменений. По щелчку на кнопке **OK** окно закроется, а в колонке **Proposed** окна на рис. 4.11 появятся новые позиционные обозначения компонентов.
- При нажатии кнопки **Accept Changes (Create ECOs)** список изменений передается на обработку. Открывается окно **Engineering Changes Order** (рис. 4.12) со списком изменений, которые предстоит внести в схему.
- По щелчку на кнопке **Execute Changes** (Обработать изменения) выполняется обработка списка изменений. Изменения, выполненные без ошибок, отмечаются значками зеленого цвета в колонке **Status**.
- Кнопкой **Validate Changes** (Установить корректность изменений) можно запустить проверку корректности внесенных изменений.

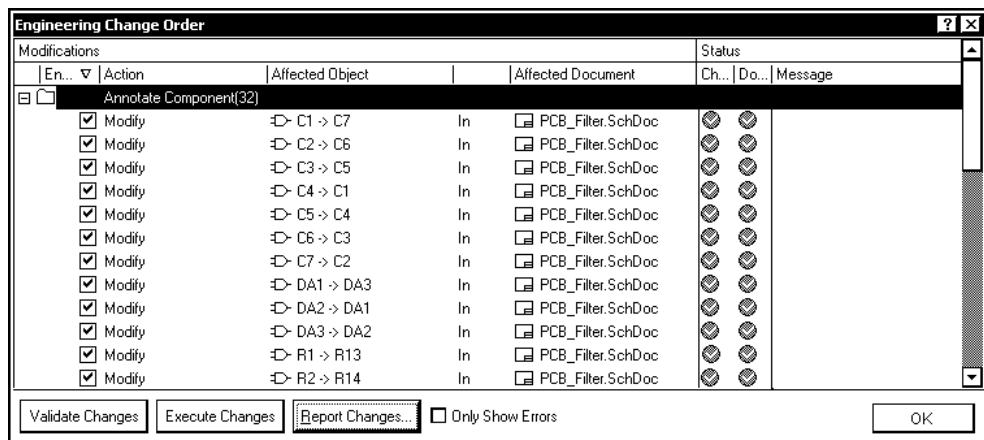


Рис. 4.12. Диалог внесения изменений в документы проекта

9. Щелчок на кнопке **Report Changes** открывает окно просмотра изменений. Список изменений можно распечатать на бумаге.
10. Щелчком на кнопке **OK** диалог закрывается, и внесенные изменения отображаются в окне (рис. 4.11).
11. При необходимости формирования ECO-документа следует указать **Back Annotate** и сохранить файл ECO стандартными средствами ОС Windows.

При повторной перенумерации после действий по пп. 1...4 нужно сбросить позиционные обозначения в колонке **Proposed** кнопкой **Reset All**, после чего продолжить перенумерацию начиная с п. 5.

4.2.5. Присвоение значений параметрам компонентов

При составлении электрической принципиальной схемы радиоэлектронных функциональных узлов, в особенности аналоговых, необходимо присвоить значения электрическим параметрам таких компонентов, как резисторы, конденсаторы, катушки индуктивности. Эти значения, а также целый ряд других сведений о компонентах, должны попасть в перечень элементов к схеме электрической принципиальной, в спецификации сборочных единиц, ведомости покупных изделий и прочие документы. Существуют программные средства, позволяющие формировать такие документы автоматически и включать их в структуру автоматизированного документооборота предприятия.

Полный набор сведений о компоненте можно включить в состав проекта одним из трех способов:

1. Включить в описание компонента при формировании библиотеки.
2. Извлечь из внешней базы данных по компонентам.
3. Составить для выбранного компонента при разработке электрической схемы функционального узла.

В Altium Designer доступен любой из этих способов.

Первый способ предполагает большой объем работы при формировании библиотек компонентной базы.

Второй способ требует наличия на предприятии компьютерной базы данных по электронным компонентам. Используемая САПР должна быть совместима по формату данных или располагать средствами импорта данных именно из этой базы. Данный вопрос будет рассматриваться далее, в *главах 11 и 12*.

Третий способ, при невозможности воспользоваться вторым, представляется наиболее практичным. Он не более трудоемок, чем составление записей в текстовом редакторе, преимущество же его в том, что при автоматическом составлении текстовых документов на основании данных схемы он поддерживает связность документов проекта.

В большинстве известных САПР радиоэлектронных функциональных узлов структура описания компонентов включает атрибуты (в Altium Designer — параметры). Обычно в набор атрибутов входит единственный атрибут Value (величина), позволяющий указать значение основного электрического параметра компонента.

Очевидно, что одного атрибута недостаточно, чтобы сформировать запись перечня элементов по ЕСКД. Добавлять в поле атрибута Value другие сведения о компоненте нельзя, если предполагается схемотехническое моделирование функционального узла.

На примере отечественного металлодиэлектрического резистора типа С-2-33 покажем, каким может быть полный объем сведений о компоненте и как правильно составить запись в конструкторской документации (рис. 4.13).

Для размещения необходимых сведений о компоненте обычно формируются специальные пользовательские атрибуты. Средства Altium Designer также позволяют сформировать полный набор сведений об используемых в проекте компонентах.

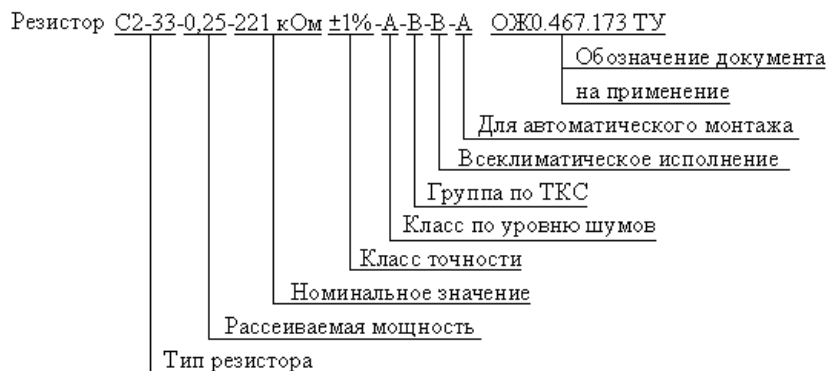


Рис. 4.13. Пример записи в перечень элементов

Для выполнения этой работы необходимо:

1. Активизировать команду главного меню программы **Tools | Parameter Manager**. Откроется диалоговое окно **Parameter Editor Options** (рис. 4.14), предоставляющее возможность выбора объектов, для которых предполагается настройка параметров.

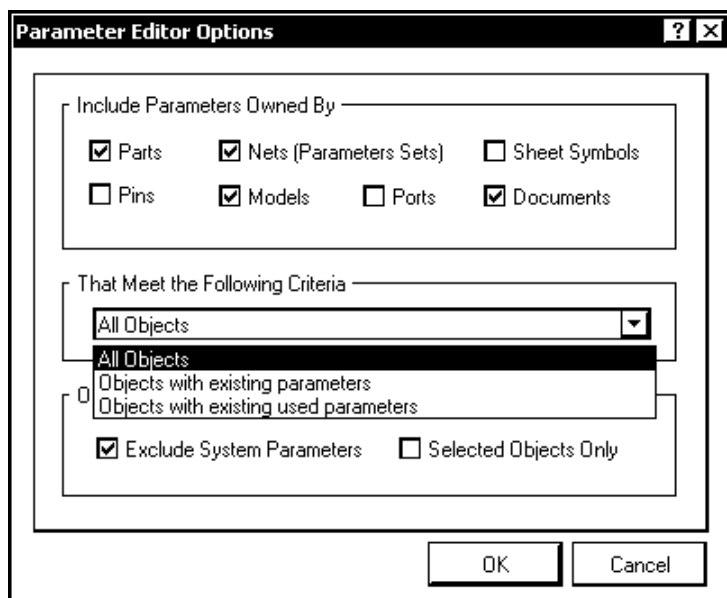


Рис. 4.14. Окно редактирования параметров элементов в документе редактора схем

2. В поле определения принадлежности параметров объектам (**Include Parameters Owned By**) указать **Parts** — схемные компоненты.
3. В поле **That Meet the Following Criteria** (Отвечает следующим критериям) указать в выпадающем списке подмножество объектов, отвечающих задаче, в нашем случае — все объекты (**All Objects**).
4. Кнопкой **ОК** закрыть окно. Откроется окно (рис. 4.15) с таблицей, в колонках которой указаны типы выбранных объектов (**Part** — схемный компонент), их позиционные обозначения по схеме, имя схемного документа и ранее назначенные параметры (атрибуты).

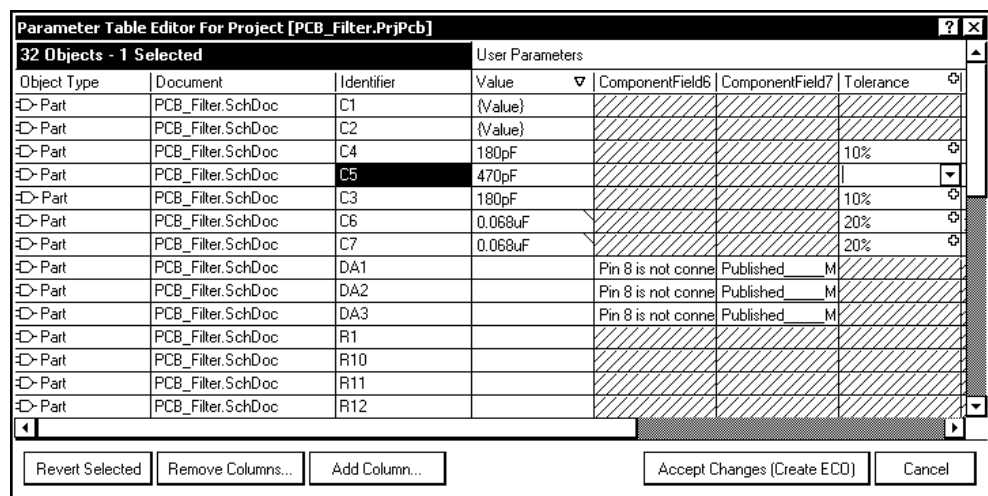


Рис. 4.15. Диалог назначения новых параметров

Число колонок параметров зависит от выбора, сделанного в пп. 2 и 3. Движок горизонтальной прокрутки обеспечивает просмотр содержания всех колонок. Среди этих колонок пока нет таких, в которых можно было бы назначить те параметры компонентов схемы, которые должны попасть в перечень элементов.

5. Щелчком правой кнопки мыши активизировать контекстное меню и указать в нем команду **Add Column** (добавить колонку). В открывающемся диалоговом окне **Add Parameter** назвать имя параметра, включаемого в добавляемую колонку, например **Tolerance** (допуск).
6. Повторить действия п. 5 столько раз, сколько дополнительных параметров предполагается назначить, присваивая соответствующие имена.

7. Выбрать в колонке с позиционными обозначениями компонент (строка, относящаяся к нему, выделяется цветом) и перейти горизонтальной прокруткой в ячейку таблицы на пересечении выделенной строки с колонкой назначаемого параметра. Щелчком левой кнопки мыши активизировать ячейку.
8. Щелчком правой кнопки на выделенной ячейке открыть контекстное меню и указать в нем команду **Add** — добавить. С выделенной ячейки снимется штриховка, и она окажется равномерно залита цветом, назначенным для селектированных объектов.
9. Щелчком правой кнопки вызвать контекстное меню и указать в нем команду **Edit** — редактировать. Ячейка станет доступной для редактирования: цвет селекции снимется; справа появится кнопка вызова выпадающего списка значений параметра. Ввести с клавиатуры или указать в выпадающем списке необходимое значение параметра (см. рис. 4.15). Подходящее значение параметра можно также скопировать (с помощью комбинаций клавиш <Ctrl>+<C> и <Ctrl>+<V>) в текущую активную ячейку из соседних, где оно уже присвоено.
10. Повторить действия пп. 8...10 столько раз, сколько компонентов нуждается в назначении параметров и сколько параметров нужно назначить каждому компоненту.

Назначенные таким способом параметры и присвоенные им значения сохраняются с файлом проекта и могут обрабатываться программами формирования текстовых конструкторских документов. Приведенная процедура не сложнее заполнения перечня элементов в текстовом редакторе. Преимущество ее в том, что она обеспечивает связность документов проекта — электрической схемы и перечня элементов.

4.2.6. Группирование цепей в классы

В электрической схеме, как и впоследствии в печатной плате, цепи, обладающие сходными свойствами, и линии групповой связи можно сгруппировать в классы.

1. Выполнить команду главного меню **Place** (Разместить) | **Directives** (Директивы) | **Net Class** (Класс цепей). На экране появится плавающая с курсором метка класса.
2. Клавишей <Tab> активизировать горячее редактирование параметров класса. Откроется диалоговое окно **Parameters**, в полях которого обозна-

чен тип параметра — **Net Class** (класс цепей), текущие координаты и ориентация метки, а также список имен ранее установленных классов.

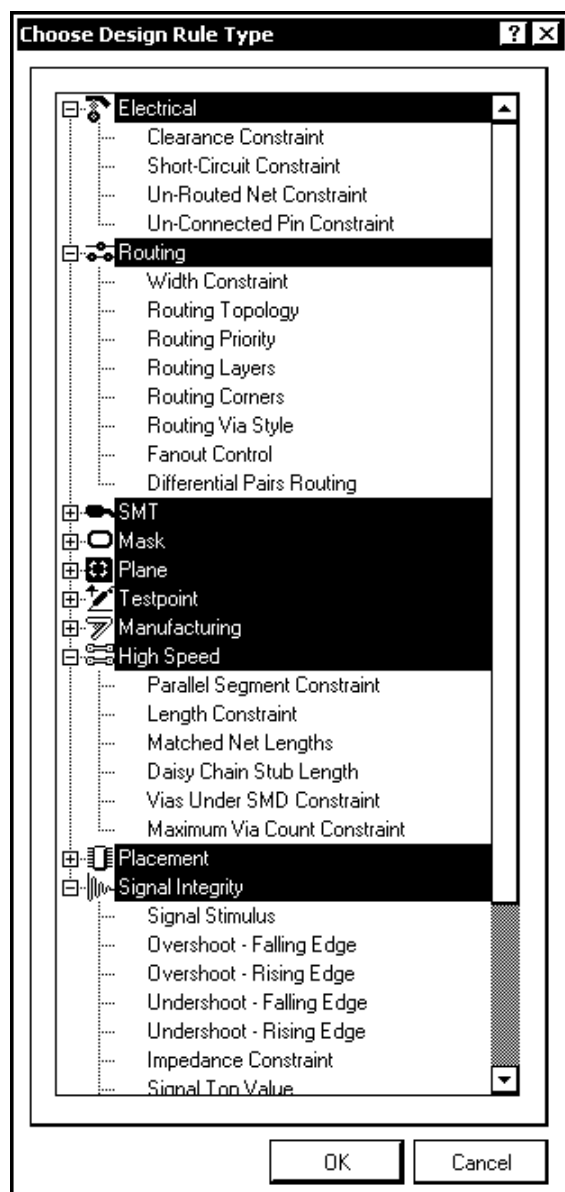


Рис. 4.16. Привязка правил к классам цепей

3. Щелчком правой кнопки мыши вызвать контекстное меню и указать в нем команду **Edit**. В диалоговом окне следующего уровня, **Parameter Properties**, можно изменить назначенное по умолчанию имя параметра **Class Name**, и присвоить ему значение (**Value**), назначить или отключить видимость, а также заблокировать то и другое.
4. Чтобы привязать к классу некоторые правила проектирования печатного монтажа, вместо команды **Edit** в диалоговом окне **Parameters** следует нажать кнопку команды **Add as Rule** — добавить параметр как правило. В диалоговом окне **Parameter Properties** активизируется кнопка вызова процедур редактирования правил **Edit Rule Values**. Открывается диалоговое окно **Choose Design Rule Type** (рис. 4.16) с деревом правил проектирования, определяющих несколько десятков правил ведения проекта — электрические и трассировочные параметры, стиль контактных площадок и ряд других.
5. Двойным щелчком в строке с именем выбранного правила открыть диалоговое окно редактирования его свойств. Например, при указании строки **Width Constraint** (Ограничение ширины) открывается диалоговое окно настройки ширины печатных проводников для цепей текущего класса (рис. 4.17).

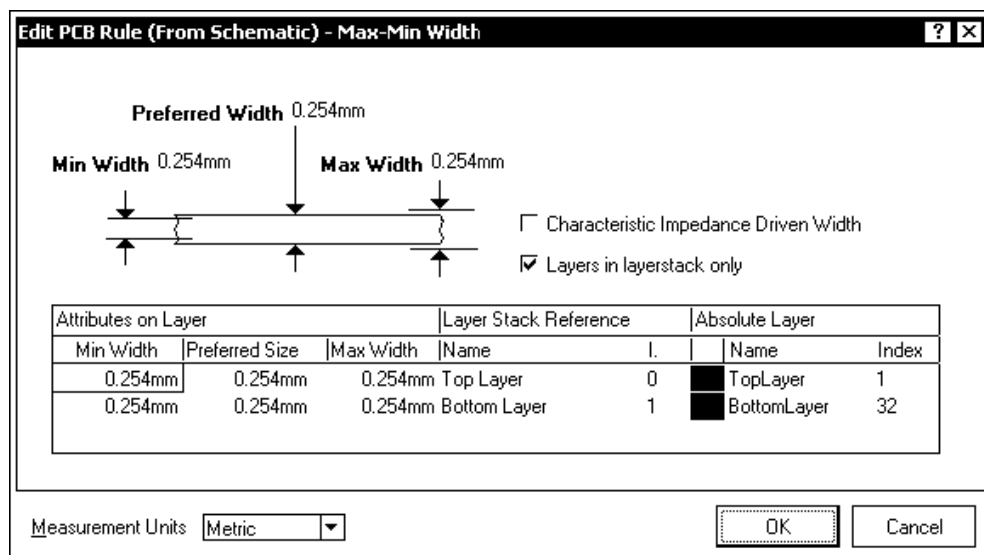


Рис. 4.17. Назначение правила для ширины печатных проводников

6. Установить необходимое значение ширины проводника и допустимое минимальное и максимальное ее значение и завершить операцию щелчком на кнопке **ОК**.
7. Нажатием кнопок **ОК** последовательно закрыть окна **Parameter Properties** и **Parameters**.
8. Зафиксировать метку класса цепей на соответствующих линиях связи в главном окне схемного редактора.

Рассмотренную процедуру можно применить также для группирования в классы линий групповой связи.

При всем изяществе рассмотренной процедуры результаты ее оказываются громоздкими. Настроенные параметры класса передаются не группе цепей, входящих в класс, а каждой цепи индивидуально. Это приводит к неоправданному увеличению списка ограничений (Constraints) в диалоге настройки правил редактирования печатной платы: вместо правила для класса в целом оно указывается для каждой цепи данного класса. Этого недостатка лишена функция настройки правил в среде редактора печатной платы, которая будет рассмотрена в *главе 5*.

4.3. Компиляция проекта

Заключительный этап разработки схемы — компиляция проекта. В процессе компиляции окончательно выстраивается логическая структура проекта, выявляются ошибки, допущенные при составлении электрической принципиальной схемы.

Для простого одноуровневого проекта с несложной однолистовой схемой достаточно визуального контроля в ходе и по окончании формирования документа. В сложном иерархическом проекте ошибки можно не заметить. Чтобы этого не происходило, в Altium Designer заложена развитая система контроля ошибок.

Перед компиляцией следует настроить функции контроля. По команде главного меню **Project | Document Options** откроется диалоговое окно **Options for Project** <имя проекта>. PrjPcb с десятью панелями-вкладками, на которых перечислены все возможные признаки ошибок проекта. Для компиляции наиболее важны настройки, призванные выявить нарушения правил соединения компонентов линиями электрической связи и соответствие обозначений цепей, портов, соединителей листов в многолистовом проекте.

На вкладке **Error Reporting** назначают характер реакции программы на обнаруженные нарушения:

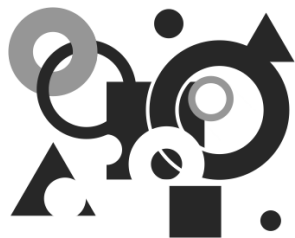
- ☐ **No Report** — не включать обнаруженное нарушение в отчет;
- ☐ **Warning** — вывести предупреждение;
- ☐ **Error** — вывести сообщение об ошибке;
- ☐ **Fatal Error** — вывести сообщение о фатальной ошибке, при которой невозможно выполнение операции.

На вкладке **Connection Matrix** определяют правила проверки электрических соединений схемы и назначается уровень реакции программы на различные виды ошибок — такие как соединение выхода с выходом, выхода с питанием, двунаправленного вывода компонента с выходным и т. п. На пересечении столбцов и строк матрицы стоят цветные метки, обозначающие реакцию программы на соответствующее соединение, обнаруженное при проверке схемы. Выбирая соответствующую метку, можно назначить один из четырех уровней реакции: отсутствие реакции (если нет ошибки), предупреждение, сообщение об ошибке, фатальная ошибка. Каждому уровню соответствует свой цвет — от зеленого до красного.

На вкладке **Comparator** настраивают правила проверки соответствия имен одинаковых объектов, цепей на листах многолиствого проекта и целого ряда других возможных ошибок.

Компиляцию проекта выполняют по команде главного меню **Project | Compile Document** <имя_схемы>.SchDoc. Если при компиляции обнаружены ошибки, то сообщения об ошибках выводятся на панель **Messages**. В этом случае следует проанализировать сообщения, внести в схемный документ необходимые изменения и повторить компиляцию проекта.

Схемный документ, откомпилированный без ошибок, можно передать на проектирование печатной платы.



Проектирование печатной платы

Будем считать, что все подготовительные операции по настройке рабочего пространства графического редактора печатных плат Altium Designer, рассмотренные в *разд. 3.2*, выполнены, схемный документ и файл заготовки печатной платы включены в структуру текущего открытого проекта. Таким образом, мы готовы к следующей стадии проектирования — разработке печатной платы.

5.1. Передача схемы в среду проектирования печатной платы

Когда проект откомпилирован и схемный документ открыт, схемную информацию на печатную плату можно передать следующим образом:

1. Выполнить команду схемного редактора **Design | Update PCB Document** <имя_проекта>.PcbDoc (Восстановить документ печатной платы). Передача схемных данных в редактор печатной платы выполняется как стандартная ECO-процедура. Открывается диалог внесения изменений **Engineering Changes Order** (рис. 5.1).

В полях диалогового окна представлена информация обо всех объектах схемного документа, действиях, которые надлежит с ними произвести (**Add** — добавить, т. е. поместить на пустую плату), и имя документа редактора печатной платы, в который вносятся данные из редактора схемы.

2. Щелчком мышью на кнопке **Validate Changes** активизировать проверку правильности вносимых изменений. Если при этом не выявлено ошибок, в поле **Status** появляются знаки-птички, свидетельствующие о заверше-

нии операции без ошибок. Строки списка, в которых обнаружены несоответствия (например, не найдено ТПМ), помечаются красной меткой с крестиком.

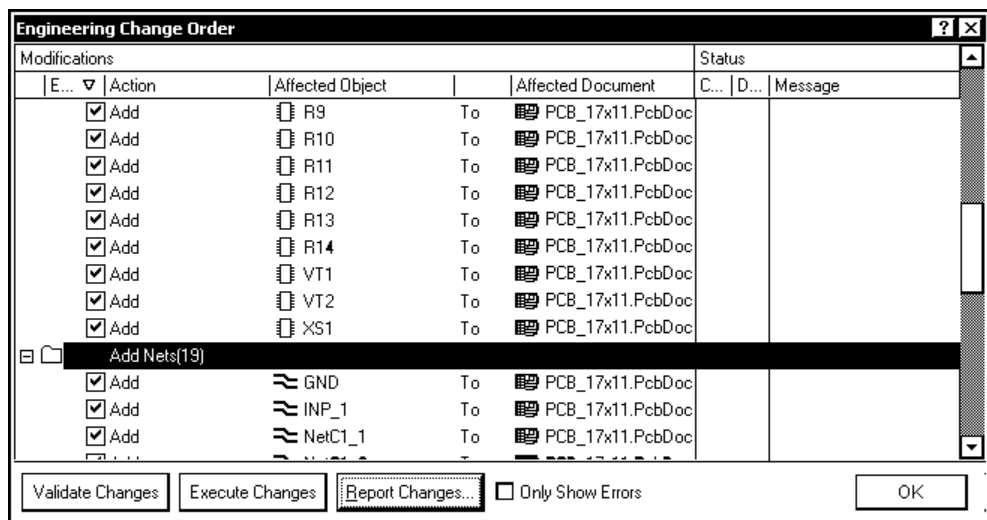


Рис. 5.1. Диалог упаковки схемы на печатную плату

В этом случае необходимо просмотреть сообщения об ошибках на панели **Messages**.

- Щелчком на кнопке **Execute Changes** запустить обработку изменений. Результат обработки передается в среду графического редактора печатной платы. При этом откроется файл платы нашего активного проекта, и в окне графического редактора печатной платы, рядом с заготовленным ранее контуром платы, отобразятся ТПМ компонентов, связанные ниточками электрических связей (рис. 5.2). По умолчанию программа заключает компоненты и связи проекта в прямоугольную область **Room** ("Комнату"). Всю эту область вместе с заключенными в ней компонентами при указании на нее курсором и нажатии левой кнопки мыши можно перемещать за курсором по экрану графического редактора и поместить в пределы контура печатной платы.

Прежде чем начать размещение компонентов по полю печатной платы, определим классы объектов, в том числе цепей, и отредактируем ряд правил проектирования.

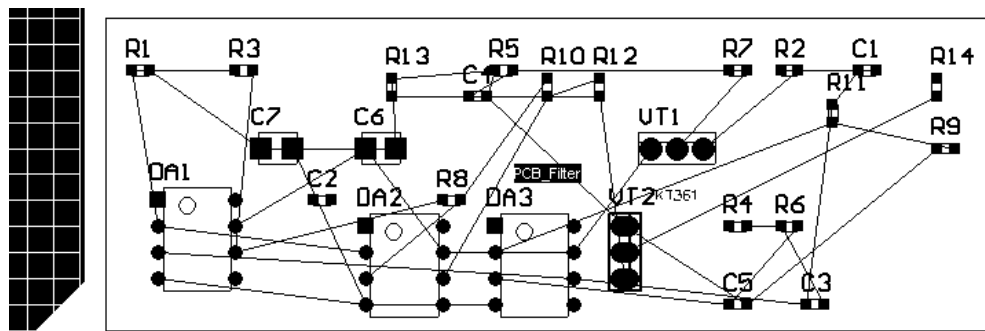


Рис. 5.2. "Комната" с упакованными компонентами

5.2. Группирование объектов в классы

Среда Altium Designer предоставляет обширные возможности группирования объектов по однородным признакам. Такие группы называются классами объектов. В классы можно сгруппировать цепи, компоненты, слои печатной платы, контактные площадки и ряд других объектов. В данном разделе рассмотрим процедуру формирования классов цепей.

Графический редактор печатной платы предоставляет свой, отличающийся от редактора схем, способ назначения классов цепей и других объектов печатной платы:

1. Активизировать команду главного меню программы **Design | Classes**. Откроется диалоговое окно управления классами объектов **Object Class Explorer** (рис. 5.3).

В левом поле окна будет изображено дерево классов с корневым узлом **Object Classes**, содержащее узлы:

- **Net Classes** — классы цепей;
- **Component Classes** — классы компонентов;
- **Layer Classes** — классы слоев печатной платы;
- **Pad Classes** — классы контактных площадок;
- **From to Classes** — классы взаиморасположения объектов;
- **Differential Pair Classes** — классы дифференциальных пар;

- **Design Channel Classes** — классы каналов проекта;
- **Polygon Classes** — классы полигонов.

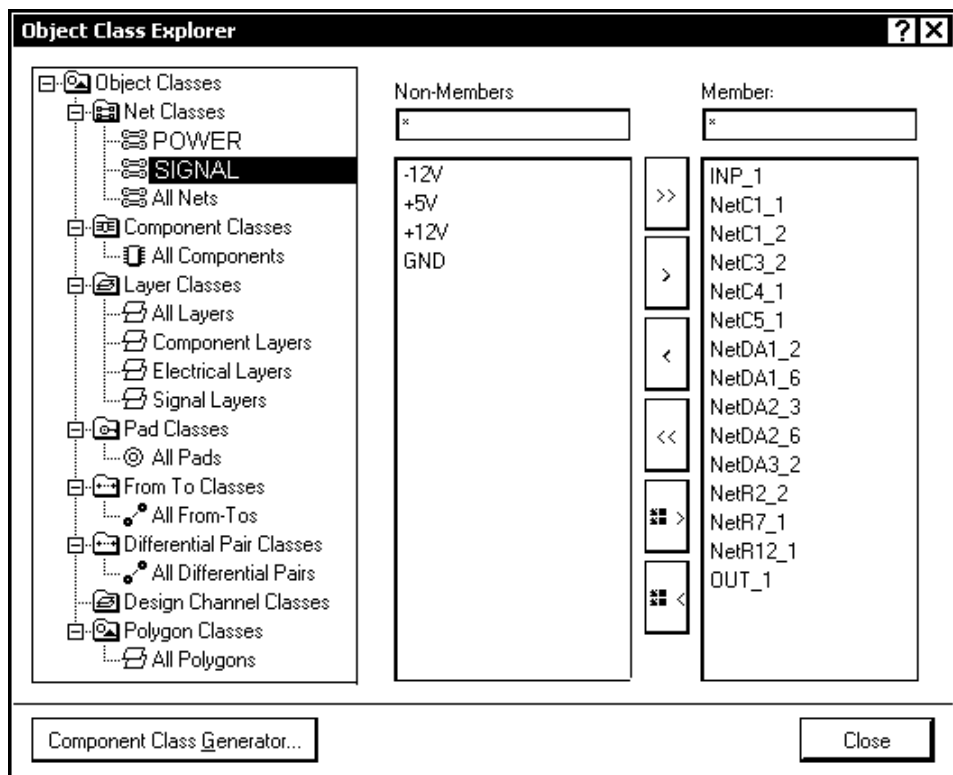



Рис. 5.3. Диалог назначения классов цепей

При определении классов цепей все цепи по умолчанию включены в единственный класс **All Nets**.

Правую часть окна занимают два поля:

- **Non-Members** — объекты, не принадлежащие к классу;
- **Members** — объекты, принадлежащие к классу (члены класса).

- Щелчком правой кнопки мыши в поле с деревом классов открыть контекстное меню и указать в нем команду **Add** — добавить (класс). В ветви **Net Classes** задать узел с именем **New Class** (Новый класс).

3. Навести курсор мыши на имя нового класса, щелчком правой кнопки мыши активизировать контекстное меню и указать в нем команду **Rename** (Переименовать). Имя нового класса оттеняется цветом селекции и окружается рамкой — становится доступным для изменения. Ввести с клавиатуры новое имя для образованного класса, например **Signal**.
4. Селектировать в дереве классов узел образованного класса. Имена всех цепей функционального узла оказываются в поле **Non-Members** — они пока не являются членами нового класса.
5. Пометить курсором, удерживая клавишу <Ctrl>, группу имен цепей, передаваемых вновь образованному классу, и щелчком по кнопке  перенести имена выбранных цепей в поле **Members**.

На рис. 5.3 показан результат группировки сигнальных цепей нашего аналогового функционального узла в класс **Signal**.

В следующем разделе будет показано, что для классов, сформированных таким способом, в каждой категории правил устанавливается единое правило сразу для всех цепей данного класса.

5.3. Настройка правил проектирования

Встроенная в Altium Designer система правил имеет ряд заметных отличий от подобных систем, используемых в других САПР функциональных узлов РЭС на печатных платах.

- ❑ Правила отделены от объектов: они не присваиваются объектам как атрибуты, а сами образуют среду, воздействующую на отдельные объекты или группы объектов, могут модифицироваться и применяться к однородным или разнородным объектам.
- ❑ Правила настраиваются и применяются к объектам через гибкую систему запросов (Query System).
- ❑ Множество однотипных правил можно применить к разным наборам объектов. Например, ширина печатных проводников, принадлежащих одной цепи, может быть разной на разных слоях.
- ❑ Правила наделяются приоритетом — для одного и того же объекта можно назначить несколько правил, и их конфликты (Contentions) разрешаются назначением приоритетов.

- Существуют два типа правил: унарные, определяющие поведение одного объекта, и бинарные, задающие взаимодействие пар объектов.

Проектирование относительно несложного функционального узла модуля фильтров, рассматриваемого в нашем примере, не требует привлечения всех возможностей настройки правил проектирования. Ограничимся настройкой параметров электрических цепей — ширины печатных проводников и зазоров между элементами печатного проводящего рисунка. Установим также значение минимально допустимого зазора между корпусами компонентов.

1. Активируем команду главного меню **Design | Rules**. В дереве правил диалогового окна **PCB Rules and Constraints Editor** (рис. 5.4) можем наблюдать узлы, образованные правилами, настроенными для класса цепей питания и "земли" в среде схемного редактора (см. разд. 4.2.6). Для каждой из четырех цепей ограничения, наложенные на допустимую величину зазоров и допустимую ширину печатных проводников, образуют узлы с именами **Schematic Clearance Constraint** и **Schematic Width Constraint**.
2. Выберем курсором в дереве правил один из узлов **Schematic Width Constraint**. В правой половине окна (рис. 5.4) в поле **Constraints** изображен эскиз фрагмента печатного проводника с обозначением минимальной, номинальной и максимально допустимой ширины проводника. Ниже эти же данные представлены в виде таблицы, в которой приведено также цветовое обозначение проводников на сигнальных слоях **Top Layer** и **Bottom Layer**.
3. В поле **Full Query** помещена надпись **InNet('–12V')**, обозначающая принадлежность проводников к цепи "минус 12 В".
4. В случае необходимости изменить ширину проводников данной цепи следует навести курсор на размерные цифры на эскизе, ввести новые значения и зафиксировать результат кнопкой **Apply** (Применить).

Рассмотрим еще один способ назначения параметров группе печатных проводников, не связанный с заданием классов цепей (для этого служит процедура формирования запросов — Query).

1. В дереве правил диалогового окна (см. рис. 5.4) выбрать узел **Width** (Ширина).
2. В поле **Constraints** установить необходимое значение ширины проводника.
3. Щелчком мышью активизировать команду-кнопку **Query Builder**. Открывается диалоговое окно формирования запросов на определение принадлежности проводника данной ширины цепям схемы **Building Query from Board** (рис. 5.5).

Первоначально в колонке **Condition Type/Operator** стоит предложение добавить (в таблицу) первое условие: **Add first condition.**

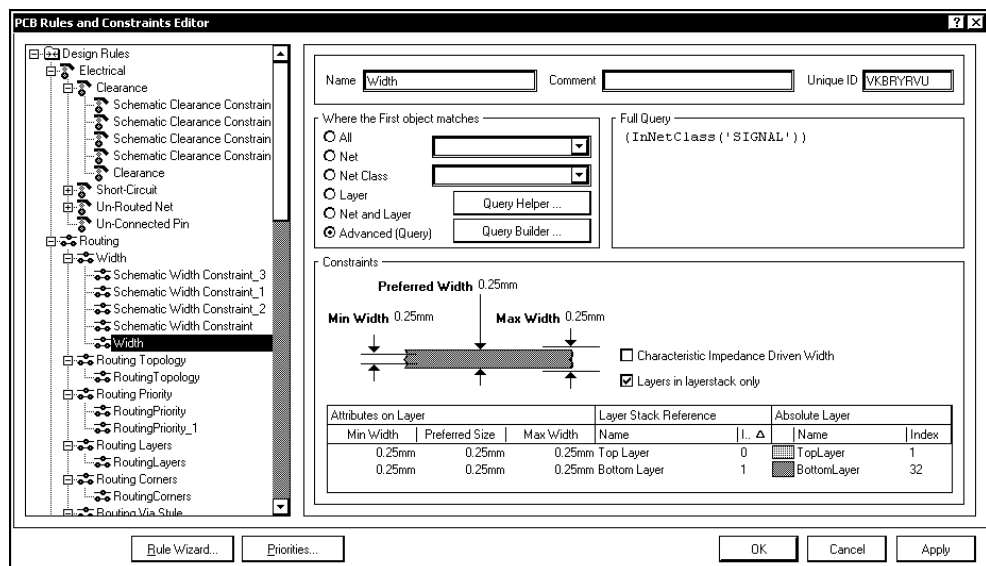


Рис. 5.4. Диалог назначения правил выполнения печатного монтажа

4. Щелчком левой кнопки мыши развернуть выпадающий список условий и выбрать в нем условие **Belongs to Net** (принадлежит цепи). В колонке **Condition Value** повторяется выражение **Belongs to Net**. Навести на него курсор и щелчком мышью открыть выпадающий список цепей функционального узла.
5. Указать в списке имя цепи "-12V". В правом поле окна, **Query Preview**, появляется строка **InNet(' -12V')**, а в колонке левого поля **Condition Type / Operator** выводится предложение добавить следующее условие: **Add another condition.**
6. Повторить действия пп. 4 и 5, указав имя следующей цепи, например "+12V". Между строками первого и второго условий появляется строка с обозначением логической функции, объединяющей два назначенных условия. Это может быть конъюнкция — функция AND (И), либо дизъюнкция — функция OR (ИЛИ). Щелчком левой кнопки мыши развернуть список из этих двух логических функций и выбрать функцию ИЛИ (OR).

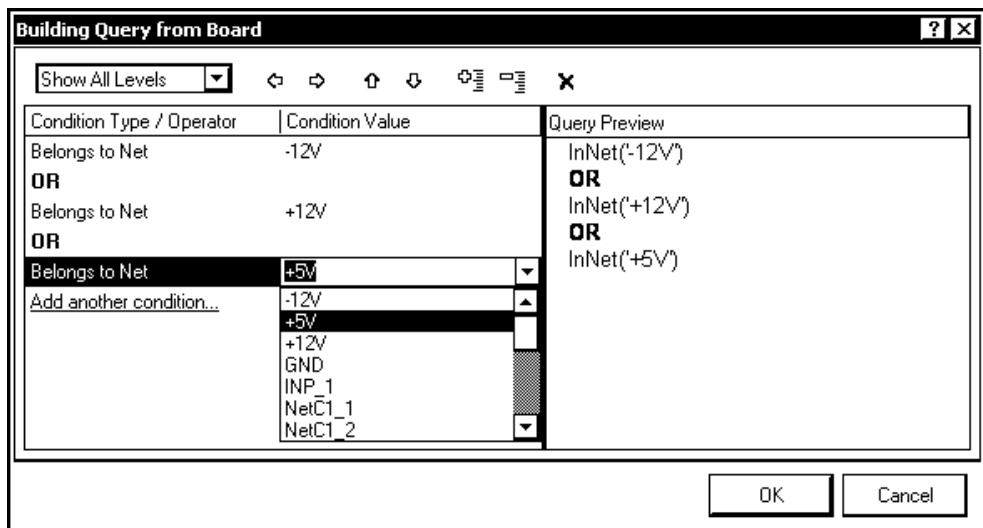


Рис. 5.5. Диалог составления запросов

- Повторяя действия пп. 4...6, составить полный список цепей, для разводки которых предназначается данная ширина проводников. Список этих цепей и правила их объединения в группу отображаются в поле **Query Preview** окна (см. рис. 5.5).
- Щелчком на кнопке **ОК** завершить диалог формирования запросов. В окне настройки правил (см. рис. 5.4) кнопкой **Apply** зафиксировать выполненные изменения правил.

Покажем теперь, как назначается ширина печатных проводников для цепей целого класса **Signal**. Для этого выберем в ветви **Routing | Width** узел **Width**, установим значение ширины проводника 0,25 мм и запустим процедуру формирования запросов **Query Builder**.

Развернем в окне (см. рис. 5.5) выпадающий список условий, но выберем в нем условие **Belongs to Net Class**, после чего в колонке **Condition Value** развернем выпадающий список классов и укажем в нем класс **Signal**.

В поле **Query Preview** выводится строка **InNetClass('Signal')**. Щелчком на кнопке **ОК** закроем формирование запросов, и в окне (см. рис. 5.4) зафиксируем настройку кнопкой **Apply** (Применить).

Назначим для нашего проекта еще одно правило — ограничение минимально допустимого зазора между корпусами компонентов. Активируем в дереве

правил диалогового окна (см. рис. 5.4) узел **Placement | Component Clearance** (Размещение | Зазор между компонентами). В поле **Constraints** изображается диаграмма (рис. 5.6), позволяющая установить необходимые значения зазоров.

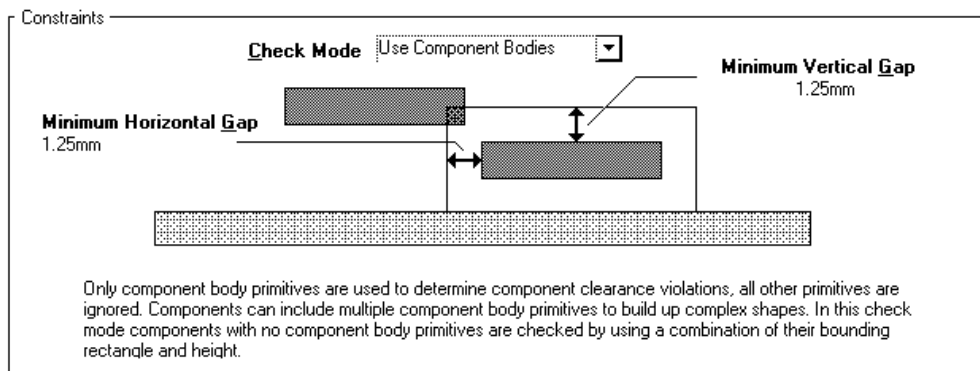


Рис. 5.6. Правило для зазоров между корпусами компонентов

Установим значения зазоров по горизонтали и по вертикали 1,25 мм и зафиксируем настройку кнопкой **Apply**.

5.4. Приоритеты правил

При назначении нескольких однородных правил-ограничений, например ширины проводников для разных цепей или классов, программа выстраивает их приоритеты в той последовательности, в которой назначались правила и обозначает приоритеты порядковыми номерами, начиная с единицы. Для изменения приоритета щелчком кнопки **Priorities** в окне на рис. 5.4 активизировать диалог редактирования приоритетов. Откроется диалоговое окно **Edit Rule Priorities** (рис. 5.7).

Кнопки **Increase Priority** и **Decrease Priority** перемещают выбранное правило вверх или вниз по списку, меняя его приоритет. Из этого можно сделать вывод, что приоритет 1 — высший.

По-другому устанавливается приоритет правила **Routing Priority**, которому назначается атрибут **Priority** непосредственно в правом поле окна на рис. 5.4. Значение атрибута **Priority** можно установить в пределах 0...100 (100 —

высший приоритет). При этом кнопкой **Priorities** обычным образом вызывается окно редактирования приоритетов, в котором правила, как и в первом случае, выстроены по списку с номерами 1, 2 и т. д. и также могут перемещаться вверх или вниз (рис. 5.8).

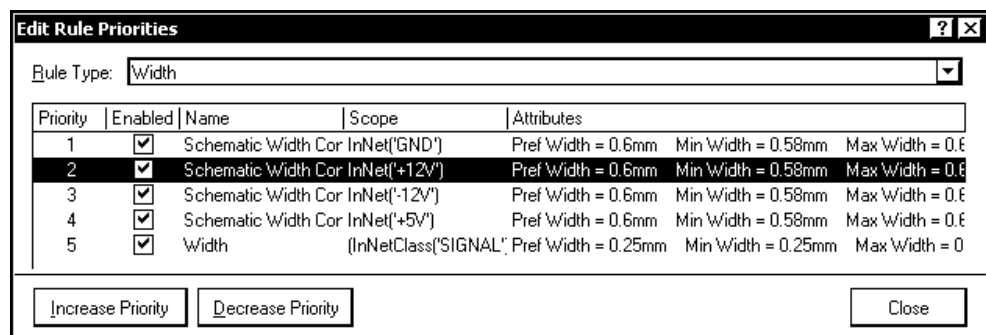


Рис. 5.7. Управление приоритетом правил для ширины проводников

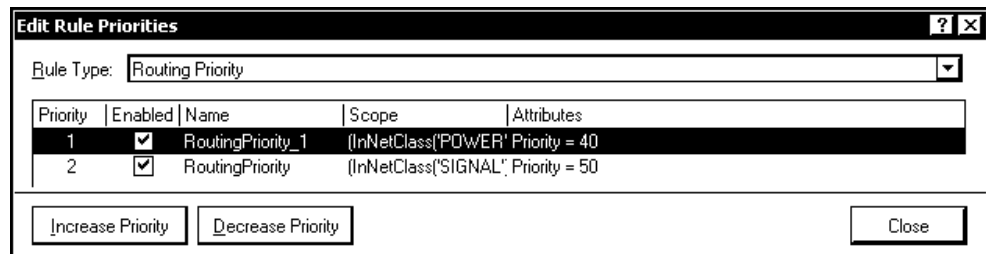


Рис. 5.8. Управление приоритетом проходов трассировки

В результате правило с более высоким значением атрибута **Priority** может оказаться в списке ниже правила с меньшим значением атрибута.

5.5. Порядок применения правил

Порядок применения правил проектирования, в зависимости от сложности проекта, может быть как простым и очевидным, так и весьма сложным. В среде проектирования печатной платы Altium Designer заложены эффек-

тивные средства контроля того, в каком порядке установленные правила проектирования применяются к различным объектам.

Возможны два метода контроля применения правил:

- ☐ выбрать объект и исследовать распространяющиеся на него правила;
- ☐ выбрать правило и провести обзор подпадающих под него объектов.

Как использовать один из этих методов или их сочетание — вопрос личных предпочтений разработчика проекта.

5.5.1. Наложение правил на объекты

Проверять можно унарные и бинарные правила. Процедура проверки приоритетов в обоих случаях практически одинакова.

Для проверки унарных правил следует навести курсор на объект на печатной плате, после чего щелчком правой кнопки мыши вызвать контекстное меню и указать в нем команду **Applicable Unary Rules** (Применяемые унарные правила). Откроется диалоговое окно **Applicable Rules** (рис. 5.9).

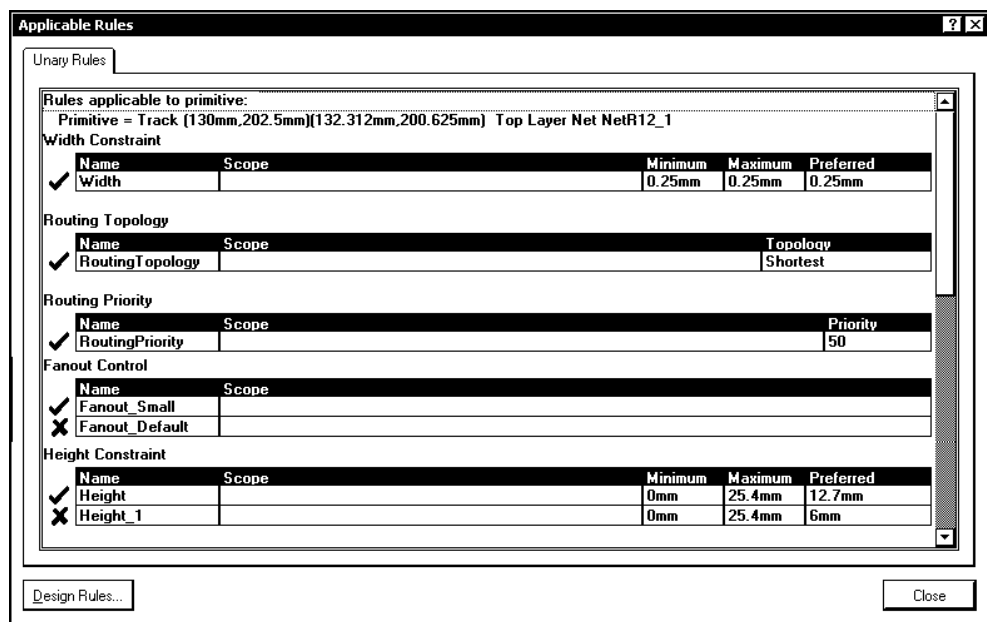


Рис. 5.9. Окно контроля унарных правил

На его активной вкладке **Unary Rules** будет указано имя выбранного объекта и развернут список правил, наложенных на данный объект.

Список разбит на категории. Информация по каждой категории сгруппирована в списки. Правила в списке расположены в порядке убывания приоритета, при этом правило, обладающее высшим приоритетом, отмечается птичкой зеленого цвета, а правила с меньшим приоритетом и правила, действие которых приостановлено, — красными крестиками. Это позволяет визуально оценить порядок применения правил.

Для редактирования какого-либо правила следует выделить его курсором и кнопкой **Design Rules** активизировать рассмотренную ранее стандартную процедуру редактирования правил.

Для проверки и редактирования бинарных правил, т. е. правил, устанавливающих взаимоотношения пары объектов, необходимо указать курсором первый объект. Щелчком правой кнопки мыши вызвать контекстное меню и указать в нем команду **Applicable Binary Rules**.

Выделить щелчками мыши два объекта, для которых назначались правила. Откроется диалоговое окно **Applicable Rules** с активной вкладкой **Binary Rules** (рис. 5.10).

На этот раз на вкладке указаны имена двух объектов, а в списке правил — правила, определяющие взаимоотношения между этими двумя объектами. Приоритет правил также отмечен значками: высший — зеленой птичкой; низший — красным крестиком.

Аналогично унарным правилам, бинарное правило также можно отредактировать.

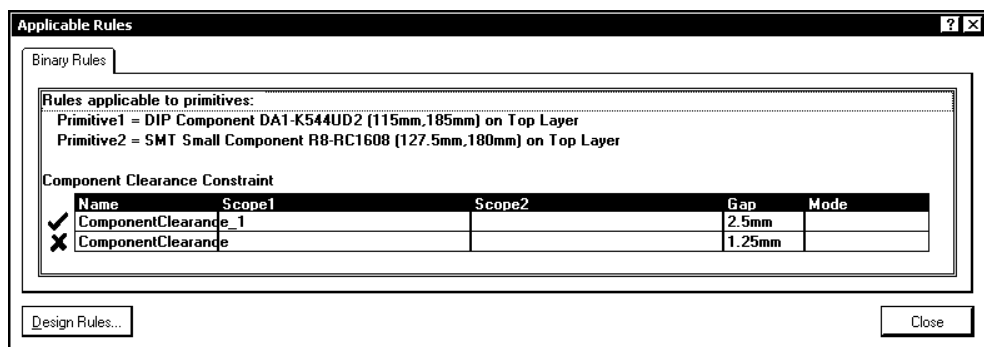


Рис. 5.10. Окно контроля бинарных правил

5.5.2. Обзор объектов, подпадающих под правило

Порядок действий таков.

1. Активизировать в среде графического редактора печатной платы плавающую панель **PCB** (рис. 5.11).
2. Настроить панель на исследование правил проектирования. Для этого в верхнем поле окна (рис. 5.11) развернуть выпадающий список и выбрать в нем строку **Rules** (Правила).
3. Ниже находится поле панели **PCB**, где представлен список установленных классов правил. В этом списке выбрать правило, устанавливающее ограничения на ширину печатных проводников — **Width Constraint**.

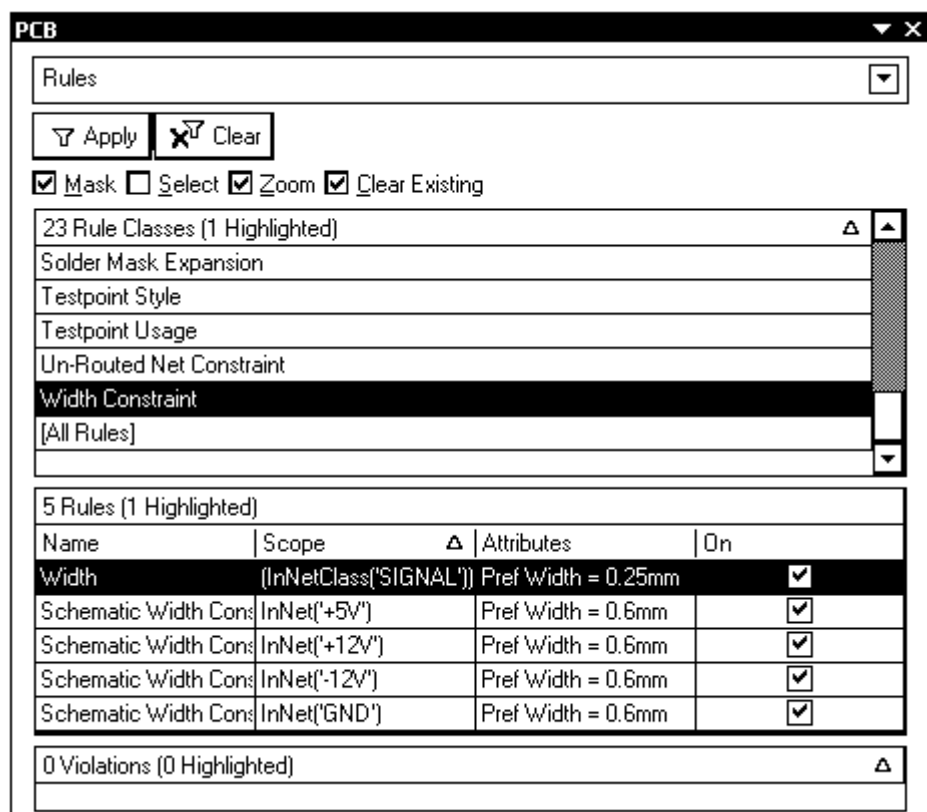


Рис. 5.11. Сводка правил проектирования в панели PCB

4. В поле, расположенном в нижней части панели, откроется список всех установленных ограничений ширины. В нашем случае это ширина проводников, принадлежащих цепям питания, "земли" (GND) и цепям класса Signal.
5. При выборе в панели **PCB** какого-либо правила в среде графического редактора печатной платы открывается процедура фильтрации объектов: в результате на плате подсвечиваются объекты, на которые распространяется это правило. В данном случае — печатные проводники.
6. Двойным щелчком левой кнопки мыши выбранное правило вызывается для редактирования. Открывается диалоговое окно с именем **Edit PCB Rule — Max-Min Width Rule**, фактически повторяющее правое поле окна, приведенного на рис. 5.4. Процедура редактирования правила, рассмотренная в *разд. 5.3*, полностью применима в данном случае.

5.6. Размещение компонентов на печатной плате

Подобно многим распространенным САПР печатных плат, Altium Designer содержит функции ручного и автоматического размещения компонентов. Критерием успеха при автоматическом размещении (там, где оно имеется) обычно служит минимум средней длины связи. При ручном размещении возможна лишь визуальная оценка. Интуитивно конструктор руководствуется подобным же критерием, хотя количественно рассчитать его трудно, поэтому вычисления обычно не делаются. Опыт показывает, что ручное размещение часто оказывается лучше автоматического. Начнем с рассмотрения ручного размещения компонентов.

5.6.1. Ручное размещение

Приемы ручного размещения компонентов на поле печатной платы отличаются от тех, к которым привыкли разработчики, работающие в P-CAD версий 2000 и выше. Основное отличие состоит в отсутствии необходимости предварительного выбора компонента.

Порядок действий таков.

1. Клавишами <Page Up>/<Page Down> отмасштабировать изображение контуров платы и лежащих поверх нее или рядом (см. рис. 5.2) ТПМ компонентов проекта так, чтобы на экране открылась зона размещения.
2. Кнопками прокрутки экрана переместить все изображение так, чтобы на поле печатной платы (в пределах ее контура) было достаточно места для размещения ТПМ в новых положениях.
3. Навести курсор приблизительно на геометрический центр ТПМ, подлежащего перемещению, нажать левую кнопку мыши и, не отпуская ее, начать перемещение ТПМ по полю печатной платы. При этом курсор изменит форму на малое перекрестие (Small Cross) и перескочит в точку привязки ТПМ (Reference Point). Удобно формировать ТПМ так, чтобы этой точкой был ключевой контакт компонента.
4. Переместить курсор (а вместе с ним и ТПМ компонента) из исходного в новое положение. ТПМ перемещается дискретно по узлам сетки Component Grid. Одновременно с этим перемещаются ниточки электрических связей. Кроме того, программа показывает на экране "вектор тяготения" компонента к другим компонентам, связанным с ним электрическими связями. Этот вектор имеет зеленый цвет, если в процессе перемещения ТПМ программа не обнаруживает нарушений правил проектирования: в норме остаются зазоры между компонентами, не пересекаются (если это вообще возможно) и не растягиваются чрезмерно электрические связи, контуры ТПМ целиком остаются в пределах контура платы или области **Room**. Если при перемещении ТПМ размещение (по оценке программы) ухудшается, вектор тяготения приобретает красный цвет.
5. Отпустить кнопку мыши. ТПМ компонента зафиксируется в новом положении.
6. Аналогично переместить ТПМ остальных компонентов.

В процессе перемещения нажатием клавиши <Пробел> (<Space>) ТПМ могут разворачиваться на угол, кратный 90°. При этом текст, связанный с компонентом (позиционное обозначение по схеме, обозначение типа ЭРК), поворачивается вместе с графикой ТПМ.

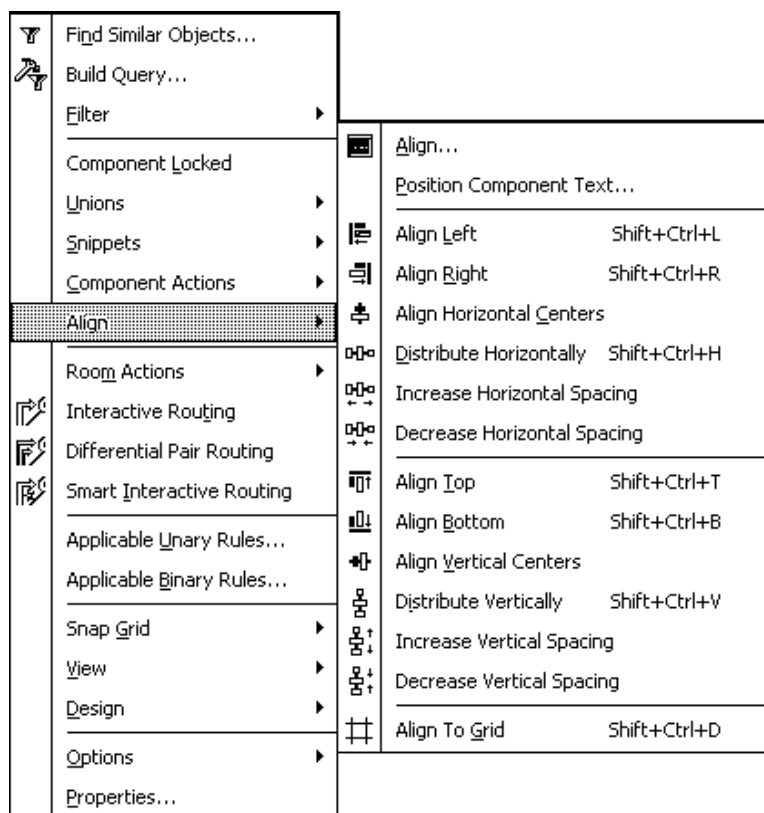
Чтобы переместить или повернуть текст, связанный с компонентом, не затрагивая ориентации ТПМ, следует сначала сделать его видимым — активизировать слой, в котором он сформирован в библиотечном компоненте (обычно это слой Top Overlay). Затем можно указать на текст курсором и

перемещать его при нажатой кнопке мыши или поворачивать нажатием клавиши <Пробел>.

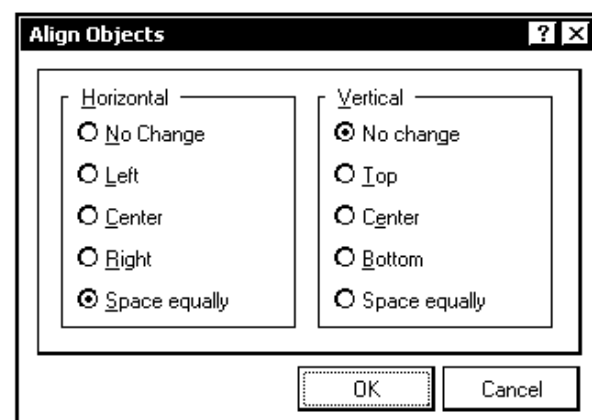
Помочь расположить компоненты, в особенности аналоговые, могут рекомендации, приводимые в руководствах по применению компонентов или публикуемые фирмами-производителями в справочных материалах. Следуя этим рекомендациям, можно избежать неприятностей, связанных, например, с возможностью возникновения паразитных связей между входными и выходными сигнальными цепями.

Кроме рассмотренных, Altium Designer содержит средства интерактивного размещения компонентов (это, в основном, функции выравнивания размещаемых компонентов и изменения расстояний между ними):

1. Нажав и удерживая клавишу <Shift>, выделить несколько компонентов на поле печатной платы. Можно вместо этого окружить группу компонентов рамкой селекции при нажатой левой кнопке мыши.
2. Навести курсор на один из выделенных компонентов и щелчком правой кнопки мыши активизировать контекстное меню (рис. 5.12, а).
3. Указать в контекстном меню команду **Align** (Выровнять). Откроется панель подкоманд:
 - **Align** — при активизации этой подкоманды вызывается диалоговое окно (рис. 5.12, б), в котором можно активизировать первые 8 из 14 команд, перечисляемых далее;
 - **Position Component Text** — изменить положение текста, связанного с компонентами;
 - **Align Left** — выровнять выбранную группу компонентов по левому краю области селекции;
 - **Align Right** — выровнять по правому краю области селекции;
 - **Align Top** — выровнять по верхнему краю;
 - **Align Bottom** — выровнять по нижнему краю;
 - **Align Horizontal Centers** — выровнять центры ТПМ по горизонтали в пределах области селекции;
 - **Align Vertical Centers** — выровнять центры ТПМ по вертикали;
 - **Align to Grid** — выровнять по узлам сетки;



a



b

Рис. 5.12. Команды выравнивания при размещении компонентов:

а — контекстное меню команд; б — окно выбора способа выравнивания

- **Distribute Horizontally** — распределить равномерно по горизонтали в пределах области селекции;
 - **Distribute Vertically** — распределить равномерно по вертикали;
 - **Increase Horizontal Spacing** — увеличить зазоры по горизонтали;
 - **Decrease Horizontal Spacing** — уменьшить зазоры по горизонтали;
 - **Increase Vertical Spacing** — увеличить зазоры по вертикали;
 - **Decrease Vertical Spacing** — уменьшить зазоры по вертикали.
4. По окончании выравнивания вывести курсор на свободное пространство в пределах графического листа редактора печатной платы и щелчком левой кнопки мыши снять выделение с выбранной группы ТПМ.

Хотя рассмотренные операции считаются функциями интерактивного размещения, но по-настоящему повлиять на размещение они не в состоянии и предоставляют лишь некоторые локальные удобства.

5.6.2. Автоматическое размещение

Следует отметить, что большинство известных САПР располагает довольно слабыми средствами авторазмещения. Не составляет исключения и Altium Designer. Тем не менее рассмотрим те средства, которыми он располагает.

В состав функций Altium Designer входят две утилиты авторазмещения компонентов — Cluster Placer и Statistical Placer. Первая предназначена для работы с простыми проектами — не более 100 компонентов на одной печатной плате. В основе второй утилиты заложен усовершенствованный алгоритм, и она пригодна для более сложных проектов.

Перед началом авторазмещения необходимо определить на плате область, в которую должны быть помещены компоненты:

1. Указать мышью в обрамлении графического окна программы вкладку с именем слоя **Keep-Out Layer**.
2. Выполнить команду главного меню **Place | Keepout | Track** (Разместить | Запретную зону | В виде дорожки).
3. Указывая щелчками мыши углы, вычертить замкнутый контур области размещения.

Затем следует указать правила и ограничения, которым должна следовать программа при авторазмещении.

4. Активизировать в главном меню программы либо в контекстном меню (щелчком правой кнопки мыши) или "горячими" клавишами <D> и <R> команду настройки правил **Design | Rules**.
5. Выбрать в дереве правил диалогового окна **PCB Rules and Constraints Editor** (см. рис. 5.4) ветвь **Placement | Component Clearance**.
6. В поле **Constraints** (Ограничения) диалогового окна на рис. 5.4 открыть выпадающий список и указать в нем режим проверки выполнения правил:
 - **Quick Check** (Быстрая проверка) — проверяется величина зазора между прямоугольниками минимального размера, охватывающими все графические примитивы, образующие очертания компонентов;
 - **Multi Layer Check** — то же самое плюс контроль наличия сквозных КП на слое **Bottom**, на случай если выполняется размещение компонентов со штыревыми выводами на одной стороне платы и планарных компонентов — на другой;
 - **Full Check** (Полная проверка) — проверяются зазоры между истинными очертаниями компонентов;
 - **Use Component Bodies** (дословно: использование тела компонента) — правило, распространяющееся на компоненты, имеющие трехмерную модель корпуса: в поле **Constraints** строится диаграмма (см. рис. 5.6), на которой указывается не только зазор между компонентами на плоскости ПП, но и зазор по высоте.
7. В полях **Where the First object matches** (Где встречается первый объект) и **Where the Second object matches** (Где встречается второй объект) сформировать подмножества объектов, между которыми должен быть назначен минимально допустимый зазор. Подмножества объектов формируются посредством процедуры генерации запросов, активизируемой щелчком на кнопке **Query Builder** (см. разд. 5.3). Однако, группируя объекты в подмножества, мы можем задать значения зазоров между объектами первого и второго подмножеств, в то время как внутри каждого подмножества зазор указать затруднительно. Для этого пришлось бы включить все объекты как в первое, так и во второе подмножество. Результат достигается более простым способом: следует выбрать в обоих полях диалогового

окна опцию **All** (Все). В этом случае программа авторазмещения проверит величину зазора между любой парой компонентов.

8. Кнопкой **Apply** (Применить) в окне настройки правил зафиксировать установленные правила управления зазорами.
 9. Кроме рассмотренного, можно настроить еще несколько правил для авторазмещения:
 - **Component Orientations** — ориентация компонентов;
 - **Permitted Layers** — разрешенные слои;
 - **Nets to Ignore** — цепи, которые следует игнорировать (поскольку главным критерием при авторазмещении является длина линий электрической связи, чересчур длинные цепи затрудняют размещение и замедляют его выполнение);
 - **Height** — учет высоты корпусов компонентов.
 10. Активизировать команду главного меню **Tools | Component Placement | Auto Placer**. В открывающемся диалоговом окне **Auto Place** выбрать опцию **Cluster Placer** и щелчком на кнопке **OK** запустить процедуру авторазмещения.
- Утилита **Cluster Placer** размещает компоненты в области **Keepout**, прижимая их к левому нижнему углу области. Если при этом зазор между компонентами или компонентами и границей области **Keepout** оказывается меньше указанного в правилах, такие компоненты, а также границы области размещения оттеняются зеленым цветом. Используя операции ручного или интерактивного размещения, можно добиться удовлетворительного результата.
- Рассмотрим еще один прием, позволяющий несколько улучшить результат авторазмещения, — размещение в областях **Room** ("Комнаты").
1. Перед размещением следует сгруппировать компоненты, подлежащие размещению в первой, второй и т. д. областях **Room** в классы. Процедура объединения объектов в классы была рассмотрена в *разд. 5.2*.
 2. Аналогично пп. 1...3 из *разд. 5.6.2* построить общую область **Keepout** для размещения всех компонентов проекта.
 3. Активизировать диалог настройки правил (командой **Design | Rules** в главном или контекстном меню или клавишами <D> и <R>).

4. В дереве правил проекта (рис. 5.13) указать ветвь **Placement/Room Definition**, щелчком правой кнопки мыши открыть контекстное меню и в нем указать команду **New Rule** — назначить новое правило. В ветви появляется новый узел с именем **Rule Definition**.
5. Выбрать в дереве новую ветвь и переименовать ее — ввести новое имя в поле **Name** в правой половине окна (рис. 5.13).

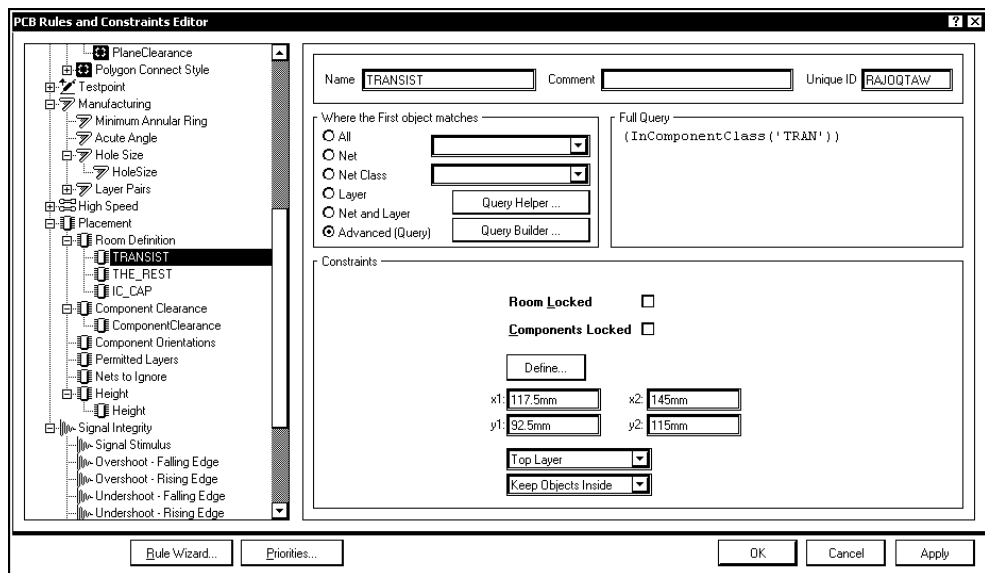
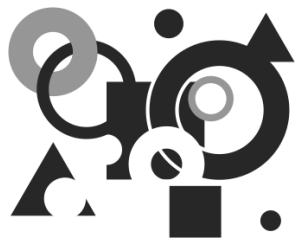


Рис. 5.13. Диалог размещения в области **Room**

6. В поле **Where the First object matches** кнопкой **Query Builder** активизировать построение запросов на назначение классов объектов, подлежащих размещению в области **Room** (см. рис. 5.5). В поле результатов **Full Query** появится сообщение (**In Component Class** ('<имя класса объектов>')).
7. В поле **Constraints**:
 - указать один из наружных слоев печатной платы, на котором размещаются компоненты (на рис. 5.13 указан верхний слой **Top Layer**);
 - задать признак размещения объектов в пределах границ области **Room** — **Keep Objects Inside**;

- кнопкой **Define** активизировать процедуру определения геометрической формы области **Room** (диалоговое окно временно закроется), после чего движениями курсора и щелчками левой кнопки мыши указать углы области **Room** так, чтобы все они лежали в пределах зоны **Keepout**, а сама область была односвязной (без пересечения ребер); если область **Room** должна иметь прямоугольную форму, вместо этого можно ввести с клавиатуры координаты противоположащих углов прямоугольника.
8. Аналогично определить следующие области **Room**, указать в них классы объектов и выполнить размещение объектов в этих областях
 9. Командой **Tools | Component Placement | Auto Placer** завершить процедуру авторазмещения.
 10. Авторазмещение, выполненное таким способом, также уступает по качеству ручному размещению компонентов.



Трассировка печатного монтажа

Проектирование печатных плат в свое время было и в некоторой степени до сих пор остается сродни искусству. Это связано с тем, что конечный результат — удачно спроектированная печатная плата, проводники которой образуют изящные конфигурации, обтекающие препятствия, переходящие со слоя на слой, — требует от конструктора проявления максимума его творческих способностей и пространственного воображения, соединенных с методическим подходом, понимания того, какие приемы позволяют добиться хорошего стиля и качества.

Одна из главных целей использования САПР при проектировании функциональных узлов РЭС на печатных платах — автоматизация разводки трасс печатного монтажа. Только автотрассировка дает настоящий скачок в производительности труда конструктора. Работа, которая вручную выполняется за недели и даже месяцы, в современных высокоэффективных САПР занимает секунды или минуты, в самых трудных случаях — часы. При разработке программных средств автотрассировки в них вкладывают элементы искусственного интеллекта — алгоритмы действия нейронных сетей (на том уровне представлений, который достигнут современной наукой о мозге), топологические алгоритмы поиска пути на графе, анализ геометрии препятствий на пути печатного проводника и т. п. В этой области получен ряд впечатляющих результатов — SPECCTRA, P-CAD Shape Router, отечественный TopoR, автотрассировщик Situs, входящий в состав Protel DXP и Altium Designer, и ряд других.

Тем не менее выполнение трассировки вручную сохраняет право на существование. Автотрассировщик обычно перебирает электрические связи проекта

по очереди и применяет при разводке систему правил и приоритетов, так называемую стратегию, в которой не всегда возможно указать "тонкие" приемы разводки. Конструктор, приступая к ручной трассировке, в состоянии окинуть проект более широким взглядом и выстроить для себя неформальную и более сложную систему правил и приоритетов, чем та, которую можно продиктовать программе. Поэтому человеческий интеллект способен решать задачу прокладки пути на монтажном пространстве с препятствиями не хуже современных программ автотрассировки, правда, ценой затраты значительно большего времени.

Существует мнение, что 90% успеха обеспечивается правильным размещением компонентов, и только 10% — непосредственно трассировкой. Поэтому в процессе трассировки может понадобиться корректировать выполненное первоначально размещение, выполнять пробные попытки, пока наконец не получится приемлемый результат. По этим соображениям начнем изложение вопроса с описания средств ручной трассировки печатного монтажа, заложенных в графический редактор печатных плат Altium Designer 6.

Будем считать, что среда графического редактора настроена, конфигурация печатной платы и структура ее слоев определены, компоненты размещены, правила трассировки (классы цепей, ширина проводников, зазоры) заданы. Все эти вопросы рассматривались в *главе 5*.

Графический редактор печатных плат САПР Altium Designer располагает тремя функциями ручной трассировки, по названию и по существу все они интерактивные:

- ❑ (Standard) Interactive Routing — стандартная интерактивная трассировка: программа подсказывает возможный путь прокладки очередного сегмента печатного проводника;
- ❑ Smart Interactive Routing — "тонкая", изощренная интерактивная трассировка: программа подсказывает и частично выполняет более сложные варианты прокладки трассы;
- ❑ Differential Pair Routing — трассировка дифференциальных пар.

При передаче электрической схемы на печатную плату электрические соединения по умолчанию изображаются нитями светло-серого цвета. Их может быть настолько много, что это затруднит выбор нитей, принадлежащих электрической связи, которая в текущий момент времени должна быть разведена в виде печатного проводника. В отличие от версий P-CAD 2000 и выше, в которых выделение трассируемой цепи цветом и яркой подсветкой указывается при настройке конфигурации графического редактора печатных плат и

затем выполняется автоматически, в Altium Designer такое выделение задается для каждой очередной цепи индивидуально несколькими способами.

1. Маскирование всего остального с использованием функций плавающей панели **PCB**:

- Щелчком левой кнопки мыши на кнопке **PCB** в правом нижнем углу главного окна программы вызвать выпадающее меню и открыть плавающую панель **PCB** (рис. 6.1).
- В поле выбора объектов сверху панели **PCB** выбрать **Nets** — цепи.

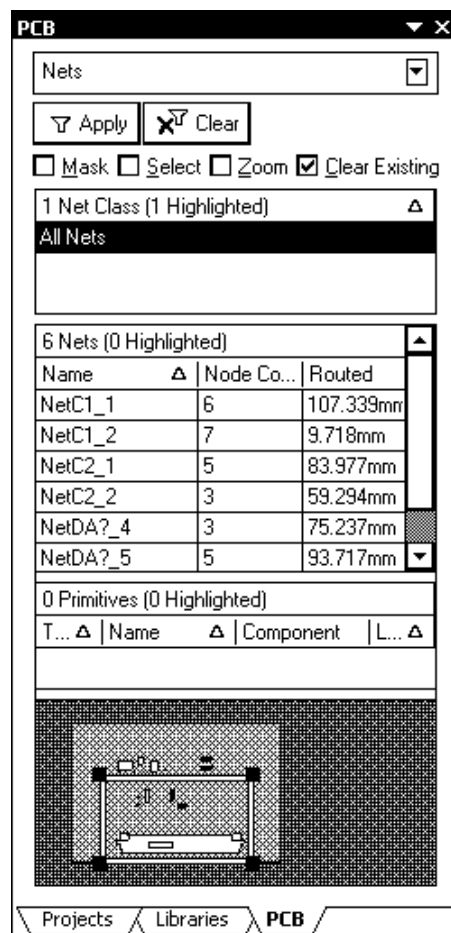


Рис. 6.1. Сводка цепей в панели PCB

- В нижележащих полях панели **PCB** выводятся:
 - ◇ список классов цепей; если классы не назначались, по умолчанию все цепи принадлежат одному классу с именем **All Nets** (Все цепи);
 - ◇ список цепей текущего активного класса: для каждой цепи указывается число узлов (контактов компонентов) и длина печатного проводника, если цепь уже разведена;
 - ◇ список графических примитивов, составляющих выбранную цепь, — контактных площадок, переходных отверстий, сегментов печатного проводника (если цепь разведена).
- В верхней части панели установить флажки опций:
 - ◇ **Mask** — маскировать все, кроме выбранной цепи;
 - ◇ **Zoom** — центрировать и увеличить изображение выбранной цепи до размеров графического окна программы;
 - ◇ **Clear Existing** — снимать выделение объектов предыдущей цепи при выборе следующей.

В результате этих настроек нити электрической связи, принадлежащие выбранной цепи, остаются видимыми на экране с прежней яркостью, а остальные связи маскируются. Маскирование снимается щелчком на кнопке **Clear**.

2. Изменение цвета выделенной цепи:

- В панели **PCB** указать курсором имя интересующей цепи и двойным щелчком левой кнопки мыши активизировать диалог редактирования параметров цепи — окно **Edit Nets**.
- Из многочисленных функций редактирования нас в данный момент интересует управление цветом нитей электрической связи — **Connection Color**.
- Щелчком левой кнопки мыши на цветном поле **Connection Color** активизировать диалог настройки цветовой палитры — окно **Choose Color** — и выбрать для выделенной цепи яркий цвет, отличающийся от цвета, назначенного по умолчанию.

3. Управление видимостью линий электрической связи:

- Активизировать в главном меню программы команду **View** и в выпадающем меню указать команду **Connections** (рис. 6.2, а).

- В подменю следующего уровня указать команду **Hide All** (Скрыть все связи).
- Повторить команду **View | Connections** и указать подкоманду **Show Net** (Показать цепь) (см. рис. 6.2, а).
- Щелчком левой кнопки мыши в произвольной точке графического окна программы открыть диалог назначения имени цепи, которая должна быть видима — окно **Net Name**. Указать имя цепи и щелчком на кнопке **OK** включить ее видимость.

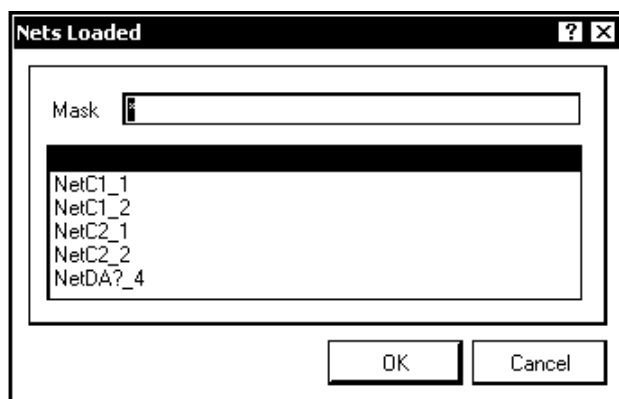
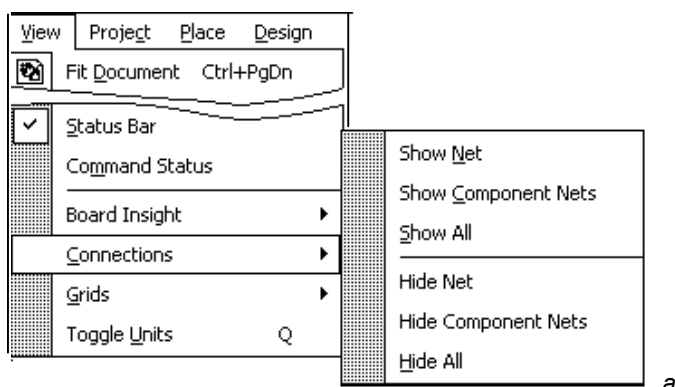


Рис. 6.2. Функции управления видимостью электрических связей:

а — подменю команды **View | Connections**;

б — окно включения/отключения видимости

- Можно не указывать имя цепи, а закрыть окно **Net Name**, щелкнув по кнопке **ОК**. При этом открывается окно **Nets Loaded** со списком всех цепей проекта (рис. 6.2, б). Выбрать курсором имя цепи из списка и щелчком на кнопке **ОК** включить ее видимость.

Далее рассмотрим заложенные в Altium Designer функции интерактивной трассировки печатного монтажа.

6.1. Стандартная интерактивная трассировка

Порядок действий таков.

1. Выбрать сигнальный слой печатной платы, на котором должна начинаться разводка очередной электрической связи. Для этого указать курсором мыши на вкладку с именем слоя в обрамлении внизу графического окна программы.
2. Активизировать команду главного меню **Place | Interactive Routing**. Курсор в графическом окне программы меняет форму на малое перекрестие. В строке сообщений главного окна программы выводится предложение указать стартовую точку трассы: **Choose Starting Location**.
3. Указать курсором на контакт одного из компонентов, охваченных линией электрической связи. Если активизирована электрическая сетка ("горячие" клавиши <Shift>+<E>), то захват контакта компонента происходит при приближении курсора к контакту на расстояние менее одного шага электрической сетки. На пересечении линий курсора возникает кольцо "прицела", обозначающее захват цепью контакта.
4. Щелчком левой кнопки мыши начать прокладку печатного проводника. Проводник изображается на экране в виде сплошной полосы, залитой цветом активного сигнального слоя.

Первый сегмент проводника ложится в направлении первого движения курсора. При отклонении курсора от первоначального направления программа строит пунктиром второй сегмент. Направление прокладки и стиль выполнения сегментов может изменяться перед началом и по ходу прокладки сегментов циклически нажатием "горячих" клавиш <Space> (<Пробел>) и <Shift>+<Space>. Всего возможны 9 вариантов начала и завершения прокладки очередных сегментов текущего печатного проводника (см. табл. 6.1 и рис. 6.3).

Таблица 6.1. Варианты выполнения печатного проводника

Но- мер вари- анта	"Горячие" клавиши		Формирование сегментов	Рису- нок
	<Shift>+ +<Space>	<Space>		
1	Line 45	Start	Начало — под углом 45°, следующий сегмент — горизонтально или вертикально	6.3, а
2		End	Начало — вертикально, следующий сегмент — под углом 45°	6.3, б
3	Line 45	Start With Arc	Начало — под 45° в виде дуги, следующий сегмент — прямой вверх или горизонтально	6.3, в
4		End — Rounded Corner	Начало — вертикально, конец сегмента в виде сопряженной дуги, продолжение — прямое под углом 45°	6.3, д
5	Line 90	Start	Начало — вертикально, следующие сегменты — под углом 90°	6.3, е
6		End	Начало — горизонтально, следующий сегмент — вертикально и т. д.	6.3, ж
7	Line 90	Start With Arc	Начало — вертикально, дугой с внутренним углом 90° или прямолинейным отрезком с дугой на конце	6.3, з
8		End With Arc	Начало — горизонтально, дугой с внутренним углом 90° или прямолинейным отрезком с дугой на конце	6.3, и
9	Any Angle	—	Прокладка сегментов — под произвольным углом, но через узлы активной сетки	6.3, в

5. Щелчком левой кнопки мыши зафиксировать первый сегмент. Второй сегмент из пунктирного превращается в сплошной. Прокладка второго сегмента продолжается, вслед за ним пунктиром строится следующий сегмент. Вторым щелчком мыши оба сегмента фиксируются на плате.
6. Горячая клавиша <1> изменяет характер прокладки трассы: оба сегмента (начальный и предлагаемый программой следующий) выполняются со сплошной заливкой и фиксируются оба сразу одним щелчком левой кнопки мыши.
7. Горячая клавиша <*> на цифровой клавиатуре изменяет слой трассировки. При трассировке многослойной печатной платы сигнальные слои перебираются циклически. В месте перехода автоматически ставится переходное отверстие, параметры которого назначены в правилах проектирования.

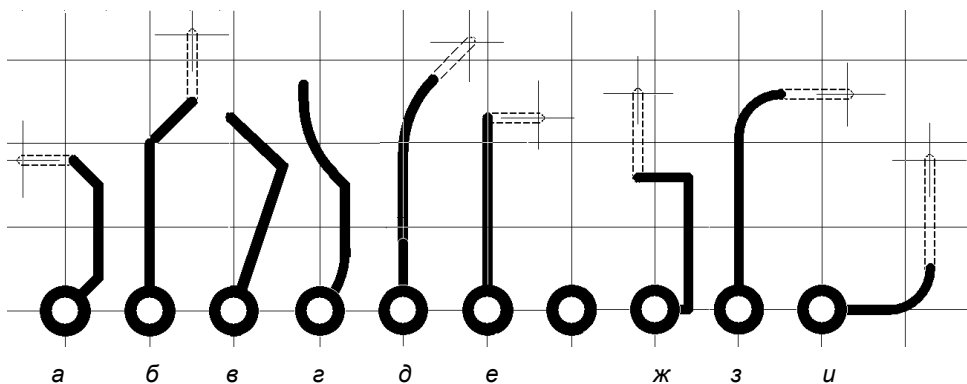


Рис. 6.3. Варианты прокладки трасс при стандартной интерактивной трассировке

8. Фиксируя щелчками левой кнопки мыши очередные сегменты, довести трассу до конечного пункта (контакта компонента). Последним щелчком зафиксировать трассу в пункте назначения и остановить дальнейшую трассировку щелчком правой кнопки мыши или клавишей <Esc>.

6.2. Разрешение конфликтов

В процессе разводки трасс печатного монтажа постоянно возникают конфликты — на пути печатного проводника оказываются цепи, проведенные ранее, контактные площадки компонентов, запретные зоны и другие препятствия. В правилах Altium Designer предусмотрено несколько способов

разрешения конфликтов. Настройка этих правил выполняется в диалоговом окне **Preferences**, в ветви дерева правил **PCB Editor — Interactive Routing** (рис. 6.4).

Всего возможны четыре способа разрешения конфликтов:

1. **Stop at First Conflicting Object** — остановиться у первого конфликтующего объекта. Этот способ установлен в настройках правил по умолчанию и называется в англоязычной лексике САПР *slam'n'jam* — "захлопываться и сжиматься". Печатный проводник останавливается перед препятствием — проводником или контактной площадкой, принадлежащими другой цепи, с соблюдением зазора, установленного правилами (рис. 6.5, а).

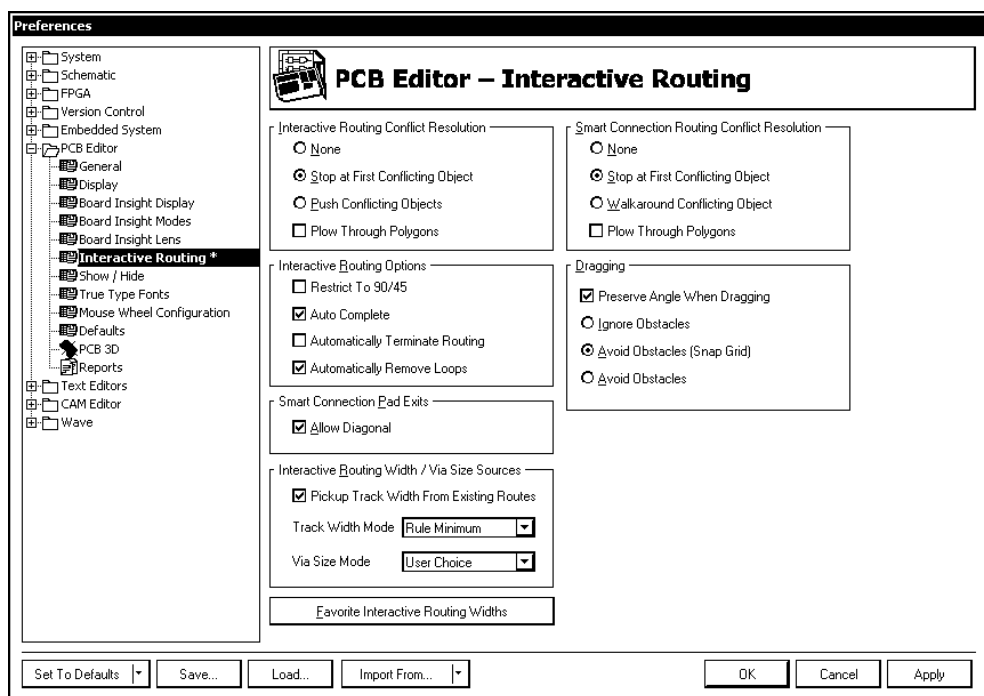


Рис. 6.4. Настройка правил разрешения конфликтов

2. **Push Conflicting Objects** — расталкивать конфликтующие объекты. Этот вариант разрешения конфликтов называется *push'n'shove* — "отталкивать и пробиваться". Он доступен только в стандартной интерактивной трассировке. Мешающие проводники деформируются так, чтобы обеспечивался установленный правилами зазор между ними и новой трассой (рис. 6.5, б).

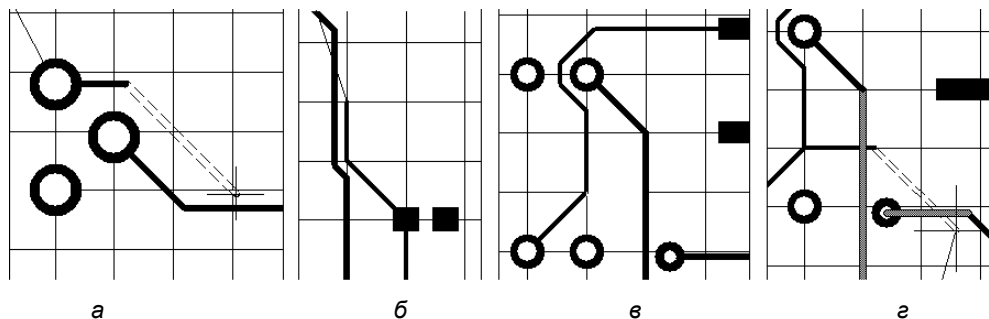


Рис. 6.5. Варианты разрешения конфликтов:

а — остановка у первого конфликтующего объекта; *б* — отталкивание конфликтующих объектов; *в* — обход объектов, мешающих трассе; *г* — игнорирование препятствий

Если "оттолкнуть" препятствие невозможно (например, это контактная площадка компонента), программа переключается в четвертый режим — игнорирование препятствий.

3. **Walk around Objects** — обходить вокруг объектов, мешающих трассе (рис. 6.5, *в*). Данный вариант разрешения конфликтов доступен только в "острой" (Smart) интерактивной трассировке. Если препятствие невозможно обойти, программа возвращается к первому варианту — остановке перед препятствием.
4. **Ignore Obstacles** — игнорировать препятствия. Нарушения правил отмечаются цветовой подсветкой, но новую трассу можно провести поверх препятствий (рис. 6.5, *г*).

Комбинацией "горячих" клавиш <Shift>+<R> возможно циклическое переключение между режимами разрешения конфликтов. Сообщение о текущем режиме выводится в строке статуса главного окна программы.

6.3. Управление шириной печатного проводника

Ширина проводника определяется настройкой правил проектирования. В правилах (см. разд. 5.3) может быть назначена минимальная, предпочтительная и максимальная ширина печатного проводника (рис. 6.6).

По умолчанию ширине проводника дается предпочтительное значение. В процессе прокладки трассы оно может меняться. При этом программа про-

веряет границы значений ширины, установленные в правилах, и не дает назначить ширину, выходящую за эти границы.

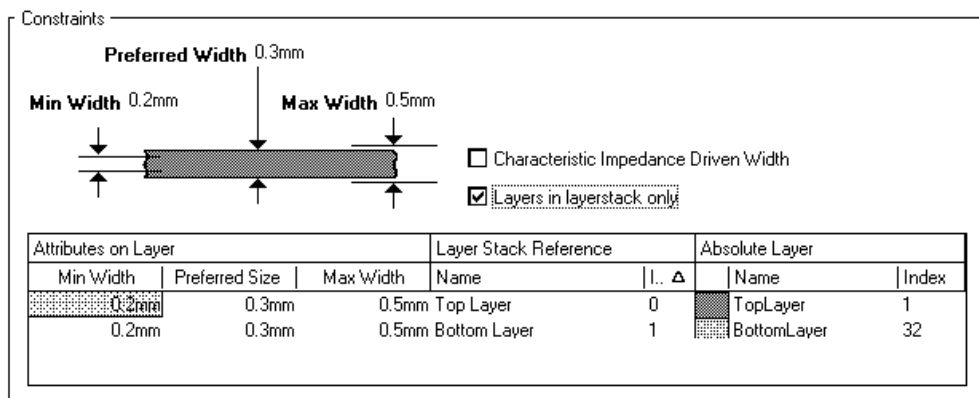


Рис. 6.6. Назначение допустимых пределов ширины трассы

Ширину можно изменять тремя способами "горячего" редактирования:

1. Комбинацией "горячих" клавиш <Shift>+<W> вызвать на экран окно с таблицей допустимых значений ширины проводников (рис. 6.7) и выбрать в ней строку с подходящим значением изменения ширины.
2. "Горячей" клавишей <O> вызвать контекстное меню и указать в нем команду **Favourite Routing Widths** — предпочтительные значения ширины. Откроется диалоговое окно **Favourite Interactive Routing Widths** с той же таблицей и кнопками управления **Add** — добавить, **Delete** — удалить и **Edit** — редактировать.
3. До фиксации очередного сегмента активизировать клавишей <Tab> диалог настройки **Interactive Routing** (рис. 6.8).

В полях зоны **Properties** — свойства — выводятся значения ширины печатного проводника (**Trace Width**), диаметр площадки металлизации (**Via Diameter**) и самого межслойного переходного отверстия (**Via Hole Size**), назначенные для текущей цепи или класса цепей в правилах проектирования (по команде **Tools | Preferences | PCB Editor | Interactive Routing**), а также информация о текущем слое трассировки, для которого назначены эти правила.

Там же приводится графическая иллюстрация назначенных правил.

Imperial		Metric		System Units
Width	Δ	Width	Units	Units Δ
5	mil	0.127	mm	Imperial
6	mil	0.152	mm	Imperial
8	mil	0.203	mm	Imperial
10	mil	0.254	mm	Imperial
12	mil	0.305	mm	Imperial
20	mil	0.508	mm	Imperial
25	mil	0.635	mm	Imperial
50	mil	1.27	mm	Imperial
100	mil	2.54	mm	Imperial
3.937	mil	0.1	mm	Metric
7.874	mil	0.2	mm	Metric
11.811	mil	0.3	mm	Metric
19.685	mil	0.5	mm	Metric
29.528	mil	0.75	mm	Metric
39.37	mil	1	mm	Metric

☒ Apply To All Layers

Рис. 6.7. Варианты выбора при управлении шириной трассы

Interactive Routing

Properties

Trace Width

0.75mm

Apply to all layers ☒

(Impedance N/A)

Via Hole Size

0.6mm

Via Diameter

1.27mm

Layer

Top Layer

Routing Width Constraints

Edit Width Rule...

Trace Width is currently constrained by the rule 'Width' to a minimum of 0.2mm and a maximum of 0.5mm.

Trace Width (0.75mm) is out of range for the current rule and will be clipped.

Via Style Constraints

Edit Via Rule...

Via style is currently constrained by the rule 'RoutingVias'. The minimum and maximum hole sizes are 0.58mm and 0.6mm. The minimum and maximum diameters are 1.27mm and 1.27mm.

Interactive Routing Conflict Resolution

☐ None

☒ Stop at First Conflicting Object

☐ Push Conflicting Objects

☐ Flow Through Polygons

Interactive Routing Options

☐ Restrict To 90/45

☒ Auto Complete

☐ Automatically Terminate Routing

☒ Automatically Remove Loops

Smart Connection Pad Exits

☒ Allow Diagonal

Interactive Routing Width / Via Size Sources

☒ Pickup Track Width From Existing Routes

Track Width Mode

User Choice

Via Size Mode

User Choice

Rule Minimum

Rule Preferred

Rule Maximum

Favorite Interactive Routing Widths

Menu

OK

Cancel

Рис. 6.8. Управление правилами по ходу трассировки

Для изменения ширины текущего сегмента прокладываемой трассы щелчком левой кнопки мыши в поле **Track Width** активизировать список предустановленных значений ширины и выбрать в нем необходимое значение (его можно также ввести с клавиатуры). Отметим еще раз, что вводимое значение должно лежать в пределах от минимального до максимального значений, назначенных в правилах проектирования. При выходе за эти пределы программа выводит предупреждение вида "Trace Width (0.75 mm) is out of range for the current rule and will be clipped" — ширина трассы (0,75 мм) выходит за пределы, установленные текущим правилом и будет усечена (до верхнего или нижнего значения, в зависимости от того, в какую сторону допущено отклонение).

6.4. Разводка групповых трасс

Один из приемов стандартной интерактивной трассировки — разводка функционально однородных цепей (шин адреса и данных ПК и микроконтроллеров, устройств памяти и т. п.) Разводка таких множественных трасс состоит из двух фаз: формирования веера при уходе трассы от контактов компонента и дальнейшей разводки групповой трассы (рис. 6.9).

Порядок действий таков.

1. Выделить курсором группу выводов компонента одним из двух способов:
 - последовательным указанием на контакты при нажатой клавише регистра <Shift>;
 - рамкой селекции, формируемой курсором при нажатой клавише <Ctrl>, — при этом выделяются только контакты компонента.
2. Активизировать команду меню **Place | Multiple Traces** (Разместить | Множественные трассы). В строке статуса выводится подсказка: **Choose Master Primitive** — выбрать главный примитив (опорный контакт, от которого начинается разводка).
3. Указать курсором на контакт компонента, вокруг которого строится веер печатных проводников. Контакт окружается кольцом "прицела".
4. Нажать левую кнопку мыши и, не отпуская, начать прокладку групповой трассы. Печатные проводники групповой трассы собираются веером под наклоном 45°. Форма веера меняется в зависимости от направления перемещения курсора — параллельно линии размещения контактов или перпендикулярно этой линии (рис. 6.9).

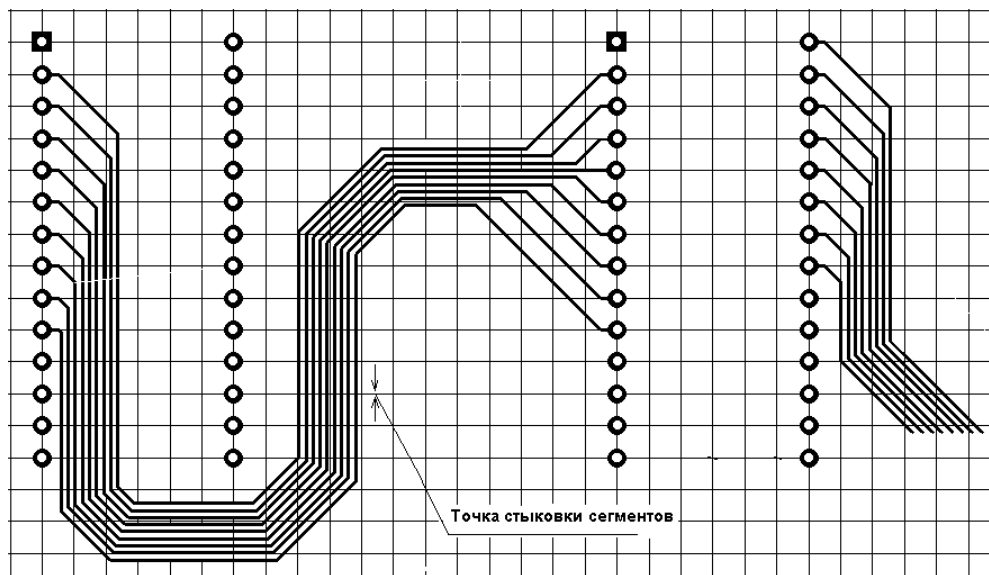


Рис. 6.9. Пример разводки групповых трасс

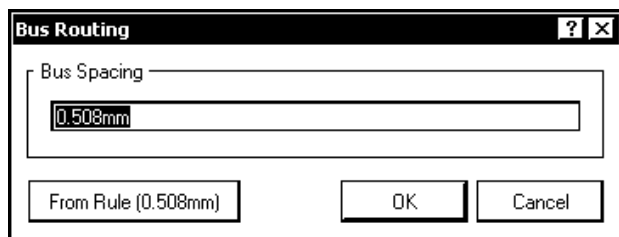


Рис. 6.10. Окно "горячего" редактирования ширины зазора

- Клавишей <Tab> активизировать "горячее" редактирование шага между проводниками трассы. В диалоговом окне **Bus Routing** (рис. 6.10) предлагается установить значение шага.

На кнопке **From Rule** в качестве подсказки размещается минимальное значение расстояния между осями печатных проводников (сумма ширины проводника и зазора), вытекающее из назначенных правил. Установить необходимое значение шага и щелчком на кнопке **OK** вернуться к прокладке трассы.

- Переместить курсор в нужном направлении и щелчком левой кнопки мыши завершить первую фазу трассировки — формирование веера.

7. При отпущенной левой кнопке мыши продолжить прокладку трассы, фиксируя изломы щелчками. Групповая трасса имеет некоторые особенности:
 - печатные проводники групповой трассы ложатся параллельно с соблюдением установленного зазора;
 - изломы трассы выполняются только под углом 45°;
 - переход трассы со слоя на слой не допускается;
 - трассировка происходит в режиме Ignore Obstacles — игнорирования препятствий; по окончании разводки программа отмечает цветной маркировкой случившиеся нарушения;
 - программа может собирать веер не только от контактов, стоящих в ряд, но и от разбросанных произвольно по плате, хотя поскольку при разводке игнорируются препятствия, на практике такой веер может испортить всю остальную трассировку.
8. В финишной точке зафиксировать трассу последним щелчком левой кнопкой мыши.

Собрав проводники веером в стартовой точке, программа не в состоянии развести их веером в точке конечного назначения. Выход — поочередно провести трассу с двух концов и состыковать встречные сегменты одноименных цепей торец в торец (см. рис. 6.9).

6.5. Редактирование готовой разводки

Кроме само собой разумеющейся перепрокладки существующих трасс по новым путям Altium Designer предлагает ряд интерактивных приемов, опирающихся на искусственный интеллект программы.

- Устранение петель. В настройке функций интерактивной трассировки (цепочка команд **Tools | Preferences | PCB Editor | Interactive Routing**) установить флажок опции **Automatically Remove Loops** — автоматически удалять петли (см. рис. 6.4). При трассировке сложных проектов печатные проводники могут проходить по весьма сложной траектории. Если при более внимательном взгляде на готовую разводку оказывается, что можно развести некоторые цепи более коротким путем, новая трасса прокладывается вручную по этому короткому пути, а старая трасса автоматически удаляется.

- ❑ Защита существующих трасс. Иногда трассу, проложенную каким-то специфическим образом, требуется защитить от действия функции устранения петель. В этом случае в диалоговом окне настройки следует снять активность опции удаления петель **Remove Loops**.
- ❑ Групповое перемещение сегментов многолинейных трасс:
 - в диалоге настройки **Tools | Preferences | PCB Editor | Interactive Rooting** активизировать флажок опции **Preserve Angle When Dragging** (Сохранять угол при перемещении);
 - выделить сегменты групповой трассы, указывая на них курсором при нажатой клавише регистра <Shift> или окружив рамкой селекции. Сегменты помечаются признаком селекции — "прищепками" ярко-белого цвета на концах и посередине, соединенными белым прямолинейным отрезком;
 - навести курсор на один из сегментов в стороне от меток-прищепок так, чтобы курсор приобрел вид перекрещенных стрелок;
 - нажать левую кнопку мыши и, не отпуская ее, переместить курсор в нужном направлении. Вся группа выделенных сегментов перемещается вслед за курсором, сохраняя исходный угол наклона. При перемещении может возникнуть конфликт с существующими трассами. Комбинация "горячих" клавиш <Shift>+<R>, как и при трассировке, изменяет режим разрешения конфликтов.

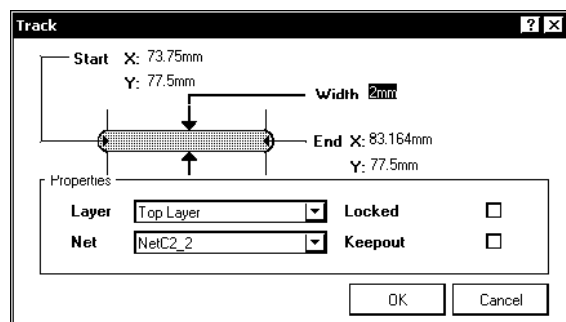
Конечный результат перемещения зависит от настроек функций разрешения конфликтов:

- при активной настройке **Ignore Obstacles** (Игнорировать препятствия) перемещенные сегменты могут быть зафиксированы в любом положении; программа помечает нарушения цветовой подсветкой;
 - при активной настройке **Avoid Obstacles** (Уклоняться от препятствий) программа "не пускает" сегменты в новое положение, если они наталкиваются на препятствие.
- ❑ Продолжение сегментов групповой трассы:
 - селектировать оконечные сегменты групповой трассы, удерживая клавишу <Shift> или рамкой селекции;
 - указать курсором на конец одного из сегментов так, чтобы курсор приобрел вид двойной стрелки.

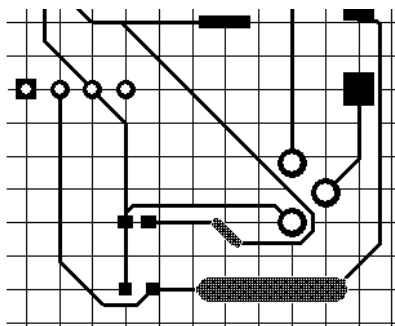
- нажать левую кнопку мыши и, не отпуская ее, переместить концы сегментов в новое положение. Отпустить кнопку мыши. Концы сегментов зафиксируются. На сегментах вновь обозначается признак выделения;
- снова указать курсором на конец сегментов и переместить их дальше в том же направлении или в сторону. Вслед за перемещением курсора строятся новые сегменты групповой трассы. Изломы трассы формируются под углом 45° .

Кроме рассмотренных, целый ряд функций редактирования трассировки активизируется подкомандами команды главного меню **Edit**:

- **Change** — изменить ширину сегмента трассы. После указания на сегмент активизировать клавишей <Tab> "горячее" редактирование и ввести с клавиатуры новое значение ширины (рис. 6.11, а). Если назначенное значение ширины выходит за границы, указанные в правилах проектирования, программа отмечает ошибку цветовой маркировкой (рис. 6.11, б);



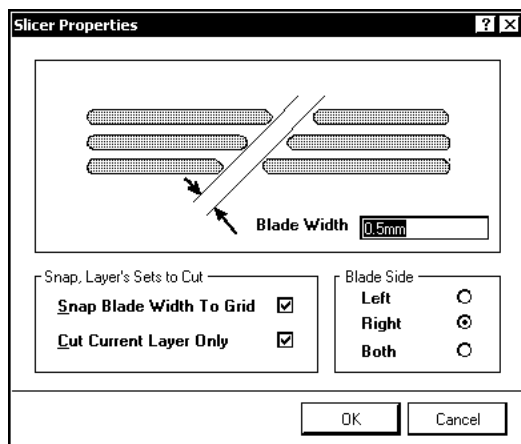
а



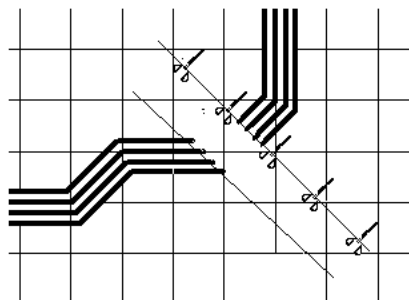
б

Рис. 6.11. Редактирование ширины сегментов готовой трассы печатного монтажа:
а — диалоговое окно управления шириной; б — результат редактирования

- **Slice Tracks** — прорезать проводники, создав разрыв заданной ширины и под заданным углом наклона. После указания исходной точки разреза клавишей <Tab> активизировать редактирование параметров разреза: фиксированной ширины, размером в один шаг сетки или произвольной ширины, а также направления разреза от исходной точки (рис. 6.12, а). Наклон линии разреза переключается "горячей" клавишей <Пробел>. Возможны два варианта — наклон под произвольным углом и наклон под углом, кратным 45° (рис. 6.12, б);



а



б

Рис. 6.12. Рассечение групповой трассы:
а — диалоговое окно настройки; б — результат рассечения

□ **Move** (Перемещать): нас в данном случае интересует группа подкоманд, относящихся к редактированию трасс печатного монтажа:

- **Re-Route** — перетрассировать;
- **Break Track** — разорвать трассу;
- **Drag Track End** — перетащить конец трассы;
- **Move/Resize Tracks** — двигать/изменять размеры трассы.

6.6. "Тонкая" интерактивная трассировка

Второй инструмент выполнения ручной трассировки — так называемая "тонкая" (Smart) интерактивная трассировка. От стандартной ее отличает наличие функции поиска пути для печатного проводника на участках от стартового до ближайшего следующего контакта компонента и один дополнительный способ разрешения конфликтов — обход препятствий (Walk around Obstacles).

Рассмотрим последовательность действий.

1. Активизировать команду главного меню **Place | Smart Interactive Routing** и выполнить действия аналогично пп. 1–4 разд. 6.1.

2. Сегменты трассы от начальной точки до текущего положения курсора выполняются со сплошной заливкой. От текущего положения курсора до контакта ближайшего по ходу электрической связи компонента тянется цепочка штриховых сегментов (рис. 6.13).

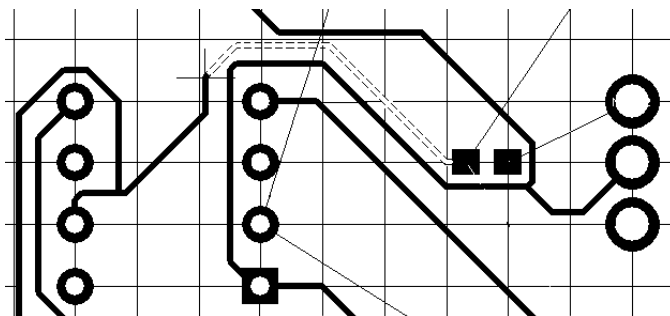


Рис. 6.13. Прокладка участка трассы в режиме "тонкой" (Smart) интерактивной трассировки

В отличие от стандартной интерактивной трассировки, программа не строит закругленные сегменты — доступны только изломы в начале или в конце сегмента под углом 45° или 90° .

Управляют характером изломов, нажимая "горячие" клавиши <Пробел> (<Space>) или <Shift>+<Пробел> (табл. 6.2). При этом клавишами <Shift>+<Пробел> циклически перебирают все варианты, а клавишей <Пробел> указывают вариант выполнения начала или конца сегмента только под углом 45° или 90° в зависимости от того, какой выбор сделан до этого клавишами <Shift>+<Пробел>.

Таблица 6.2. Варианты "тонкой" интерактивной трассировки

Номер варианта	"Горячие" клавиши		Формирование сегментов	Рисунок
	<Shift>+<Space>	<Space>		
1	Line 45 Start	Line 45 Start	Начало — под углом 45° , следующий сегмент — горизонтально или вертикально	6.3, а
2	Line 45 End	Line 45 End	Начало — вертикально, следующий сегмент — под углом 45°	6.3, б

Таблица 6.2 (окончание)

Номер варианта	"Горячие" клавиши		Формирование сегментов	Рисунок
	<Shift>+<Space>	<Space>		
3	Line 90 Start	Line 90 Start	Начало — вертикально, следующие сегменты — под углом 90°	6.3, е
4	Line 90 End	Line 90 End	Начало — горизонтально, второй сегмент — вертикально и т. д.	6.3, ж

3. Способ разрешения конфликтов, как и в стандартной интерактивной трассировке, выбирается комбинацией "горячих" клавиш <Shift>+<R>. В отличие от стандартной, в Smart-трассировке доступен способ разрешения конфликтов обходом вокруг препятствий (см. рис. 6.13). При этом соблюдаются правила проектирования: обходной участок трассы прокладывается только, если ширина прохода достаточна для размещения проводника с соблюдением установленных зазоров.
4. Завершение прокладки участка трассы выполняется по-разному, в зависимости от настройки функции **Auto Complete** (автоматического завершения) в конфигурации редактора печатной платы (см. рис. 6.4):
 - при отключенной функции **Auto Complete** программа показывает сегменты трассы только до текущего положения курсора. Сегменты фиксируются последовательно щелчками левой кнопки мыши;
 - при активной функции **Auto Complete** прокладку участка трассы можно выполнить последовательной фиксацией сегментов щелчками левой кнопки мыши, а можно завершить целиком — единственным щелчком левой кнопки мыши при нажатой клавише <Ctrl>.

Включение/отключение активности функции **Auto Complete** может выполняться в текущий момент трассировки "горячей" клавишей <5>.

Полный список "горячих" клавиш (рис. 6.14), используемых при управлении Smart-трассировкой, вызывается в любой текущий момент прокладки трассы клавишей <~> (тильда). Приведем его перевод с некоторыми разъяснениями:

- ☐ <F1> — получение справочной информации;
- ☐ <Ctrl>+<Click> — завершение трассировки участка по щелчку левой кнопки мыши с нажатой клавишей <Ctrl> (если установлена активность функции автозавершения);

Help	F1
Commit autocomplete segments (if applicable)	Ctrl+Click
Remove Last Segment or Shift+Backspace to remove last placed cluster	BkSp
Terminate Current Trace	Esc
Toggle Routing Mode	Shift+R
Choose Favorite Width	Shift+W
Place Segment	Enter
Next Layer	Num +
Previous Layer	Num -
Next Signal Layer	Num *
Cycle Placement Mode	Space
Edit Trace Properties	Tab
Cycle Track-Width Source	3
Cycle Via-Size Source	4
Toggle Auto Complete	5
Change Via mode	6
Switch Leader trace or switch routing target in single trace mode	7

Рис. 6.14. "Горячие" команды управления Smart-трассировкой

- ❑ <BkSp> — удалить нажатием клавиши <Backspace> (Шаг назад) последний сегмент трассы или группу сегментов при нажатии <Shift>+<Backspace>;
- ❑ <Esc> — прервать нажатием клавиши <Esc> разводку текущего участка трассы с потерей сегментов, равноценно щелчку правой кнопкой мыши: если трассировка участка закончена на контакте компонента, то программа выходит из режима трассировки;
- ❑ <Shift>+<R> — переключение режима разрешения конфликтов;
- ❑ <Shift>+<W> — активизация функции управления шириной текущего сегмента трассы;
- ❑ <Enter> — зафиксировать текущий сегмент трассы — то же, что и щелчок левой кнопкой мыши;
- ❑ <Num +> (плюс на цифровой клавиатуре) — выбор следующего слоя;

- ❑ <Num -> (минус на цифровой клавиатуре) — выбор предыдущего слоя;
- ❑ <Num *> (звездочка на цифровой клавиатуре) — переход трассы на следующий сигнальный слой;
- ❑ <Space> — выбор направления прокладки в начале или конце текущего сегмента;
- ❑ <Tab> — активизация "горячего" редактирования свойств текущего сегмента (см. рис. 6.8);
- ❑ <3> — циклический перебор значений ширины текущего сегмента;
- ❑ <4> — циклический перебор значений диаметра площадки металлизации у переходного отверстия — минимального, номинального или максимального;
- ❑ <5> — включение/отключение функции автозавершения трассировки участка цепи;
- ❑ <6> — переключение типов переходных отверстий;
- ❑ <7> — если трасса от исходного контакта расходится в нескольких направлениях — прекращение трассировки в данном направлении и переход к трассировке в другую сторону.

Печатные проводники, выполненные Smart-трассировкой, могут редактироваться, как это описано в *разд. 6.4*.

6.7. Трассировка дифференциальных пар

Необходимость в передаче дифференциального сигнала по линиям связи возникает в случаях, когда нет возможности осуществить соединение короткими проводами, а цепи питания и "земли" имеют достаточно высокое электрическое сопротивление. Это происходит главным образом при межблочном монтаже. Решение проблемы — формирование пар противофазных сигналов и использование соответствующей схмотехники электронных компонентов. Такая схмотехника известна — это широко распространенные дифференциальные или операционные усилители в аналоговых функциональных узлах и схемы эмиттерно-связанной логики (ЭСЛ) в цифровых устройствах. Каждый сигнал дифференциальной пары может передаваться по отдельному экранированному проводу или радиочастотному коаксиальному кабелю либо сигналы дифференциальной пары передаются по кабелю из двух скрученных проводов — так называемой "витой паре".

Оба способа передачи дифференциальной пары сигналов отличаются высокой помехоустойчивостью. Физика явлений известна:

- поскольку по проводам дифференциальной линии передачи текут равные по величине и противоположные по направлению токи, такая линия не создает помех в окружающем пространстве;
- внешние помехи создают в проводах дифференциальной линии передачи синфазную помеху, которая существенно подавляется дифференциальными входными каскадами аналоговых и цифровых схем;
- выравнивание времен распространения сигнала по проводам линий связи обоих типов в значительной мере достигается само собой — за счет равенства длины проводников в витой паре или за счет равенства длины двух отрезков коаксиального кабеля, образующих дифференциальную линию.

Кроме того, на приемном конце такой линии связи возникает эффект "виртуального" заземления, в результате чего согласование импеданса линии с входным сопротивлением каскада на приемном конце может выполняться без заземления — одним согласующим элементом, импеданс которого должен равняться удвоенному значению волнового сопротивления в случае коаксиального кабеля с заземленной оболочкой либо волновому сопротивлению витой пары.

При передаче однофазного сигнала по коаксиальному кабелю или витой паре также происходит взаимная компенсация магнитных полей прямого и обратного токов, в результате чего помехи в окружающее пространство не излучаются. Что касается чувствительности к внешней наводке, то степень подавления емкостной помехи определяется экранирующим действием оболочки коаксиального кабеля или заземленного провода витой пары. Индукционная наводка создает нескомпенсированную помеху в активном проводе витой пары, в особенности, если линия связи образует петли.

Приходящую на печатную плату из внешней разводки симметричную или асимметричную линию связи можно довести от соединителя или от контактов на краю платы до выводов электронного компонента только парой параллельно расположенных печатных проводников. Такую пару печатных проводников также называют дифференциальной парой.

Дифференциальная пара на печатной плате не обеспечивает столь же высокую помехоустойчивость, как кабельные линии связи, вследствие того, что значение коэффициента связи параллельных печатных проводников не пре-

вышает 0,5, в то время как у коаксиального кабеля и в витой паре это значение составляет 0,98 и выше.

Таким образом, наиболее ощутимым эффектом, который обеспечивает применение дифференциальных пар на печатной плате, является выравнивание времени распространения сигнала до входных контактов электронного компонента. Этому, как и согласованию волнового сопротивления линии передачи с нагрузкой, могут препятствовать особенности трассировки и размещения компонентов, необходимость перехода со слоя на слой, наличие полигонов металлизации в сигнальных слоях или на внутренних экранных слоях типа Plane. Подробное рассмотрение радиотехнической стороны данного вопроса выходит за рамки данной книги. Приведем только некоторые рекомендации по выполнению дифференциальных (передающих симметричный дифференциальный сигнал) линий связи на печатных платах¹:

- ❑ волновое сопротивление каждого проводника пары должно равняться половине волнового сопротивления внешнего кабеля (витой пары);
- ❑ для каждого проводника пары сопротивление нагрузки на приемном конце должно быть согласовано с волновым сопротивлением проводника;
- ❑ переходы линии со слоя на слой допустимы, если удастся обеспечить постоянство волнового сопротивления по всей длине линии;
- ❑ оба проводника пары должны иметь одинаковую длину, с точностью, обеспечивающей разность задержек в пределах допуска, установленного для применяемого семейства логических интегральных микросхем.

Покажем на примере, какое значение может принимать разность длины проводников дифференциальной пары в высокоскоростных функциональных узлах, работающих с тактовой частотой порядка 1 ГГц.

Известно, что скорость распространения сигнала вдоль микрополосковой линии связи в $\sqrt{\epsilon}$ раз меньше скорости в свободном пространстве. При типичном для печатных плат значении диэлектрической проницаемости стеклотекстолита $\epsilon = 4,8$ время задержки распространения сигнала составляет около 7,3 нс/м.

Если принять, что разность времени прихода сигнала на входы логики не должна превышать 0,1 тактового интервала, т. е. 0,1 нс, разность длины проводников дифференциальной пары не должна превышать 13 мм.

¹ Lee W. Ritchey, Differential Signalling Doesn't Require Differential Impedance.
<http://www.speedingedge.com/PDF-Files/diffsig.pdf>.

Так или иначе, нам следует рассмотреть заложенные в Altium Designer функции трассировки дифференциальных пар¹.

6.7.1. Объявление дифференциальной пары

Объявить две сигнальные цепи дифференциальной парой можно как в редакторе схемы, так и в редакторе печатной платы.

Дифференциальные пары в редакторе электрической схемы

Выполнить команду главного меню **Place | Net Label** и присвоить двум цепям, образующим дифференциальную пару, одно имя с суффиксами **_P** для одной цепи и **_N** — для другой.

Активизировать команду **Place | Directives | Differential Pair** и поместить метку дифференциальной пары на каждую из цепей пары (рис. 6.15).

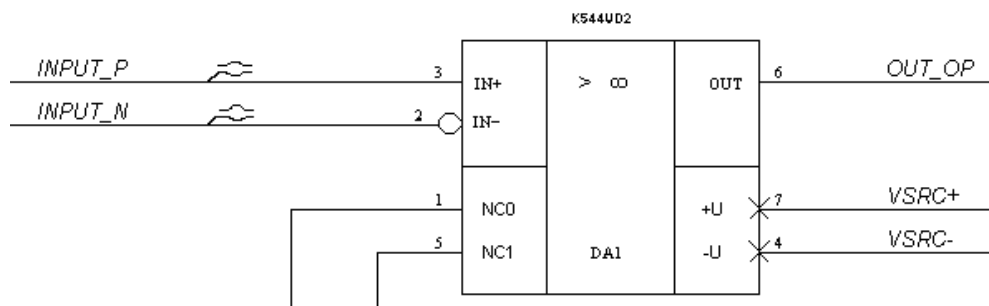


Рис. 6.15. Назначение дифференциальной пары в схемном документе

Цепи, на которую наложена директива **Differential Pair**, присваивается параметр с таким же именем **Name: Differential Pair** и булевым значением **Value: True**.

Выполненные таким образом назначения передаются в проект печатной платы при компиляции и синхронизации проекта.

¹ Документ HELP-системы Altium Designer: Interactive and Differential Pairs Routing. Application Note AP0135 (v. 1.2) June 19, 2006.

Дифференциальные пары в редакторе печатной платы

Пары цепей, не объявленные как дифференциальные в редакторе схемы, можно объявить и оттрассировать как дифференциальные. Здесь возможны два варианта действий.

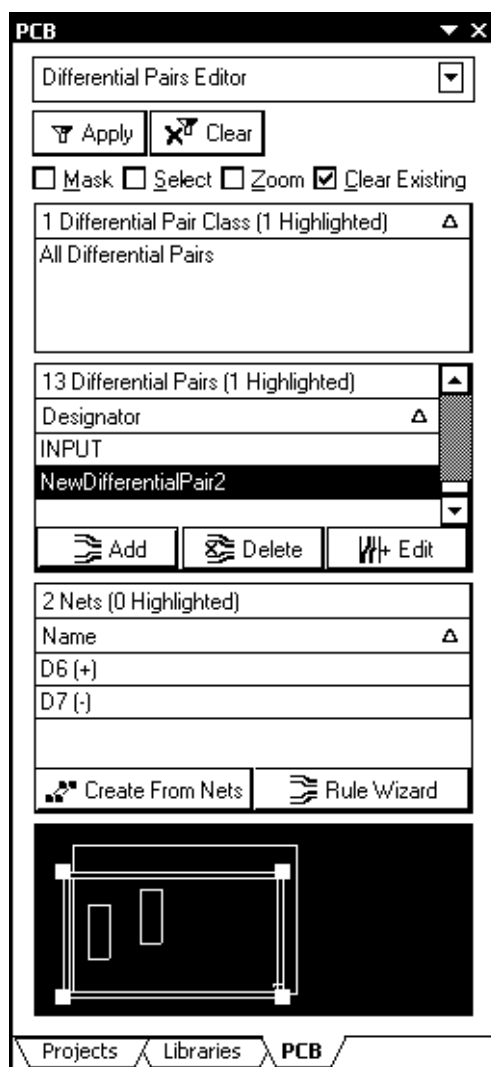


Рис. 6.16. Объявление дифференциальных пар в панели PCB

В первом случае необходимо сначала активизировать плавающую панель **PCB** и переключить ее в режим редактирования дифференциальных пар **Differential Pairs Editor** (рис. 6.16). В среднем поле панели выводится список всех определенных в проекте дифференциальных пар, в том числе ранее назначенных в редакторе схемы.

Далее надо выполнить последовательность действий:

1. Щелчком на кнопке **Add** активизировать объявление новой дифференциальной пары. Откроется окно формирования дифференциальной пары **Differential Pair** (рис. 6.17).
2. В полях **Positive Net** и **Negative Net** поочередно развернуть списки цепей, не объединенных пока в дифференциальные пары, и выбрать в каждом списке необходимую цепь.
3. В поле **Name** (Имя) зоны **Properties** ввести имя новой дифференциальной пары.
4. Щелчком на кнопке **OK** завершить формирование пары.

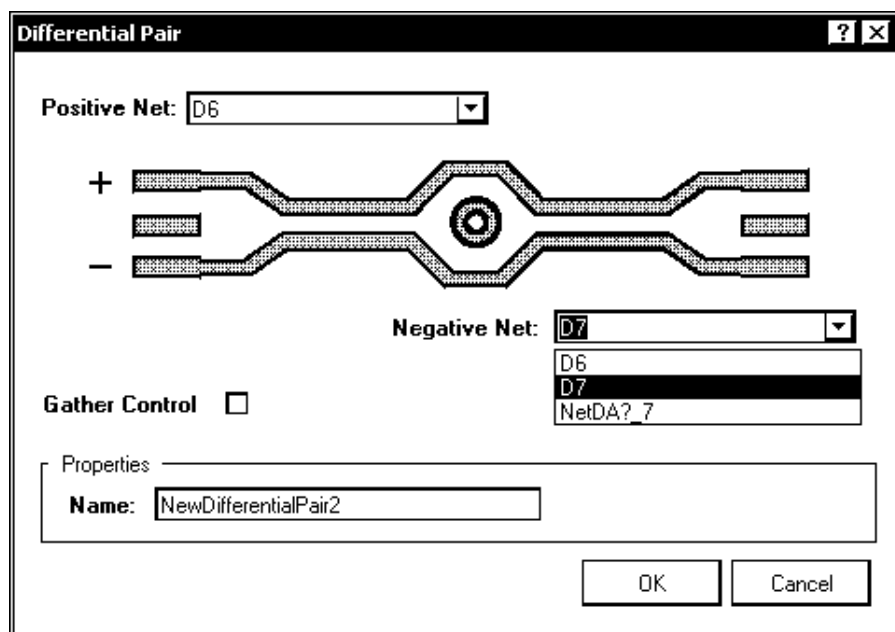


Рис. 6.17. Диалог назначения новой дифференциальной пары

Результат отобразится в полях панели **PCB** (см. рис. 6.16). При выделении курсором имени новой пары в колонке **Designator** центрального поля панели в нижележащее поле выводятся имена двух цепей, объединенных в дифференциальную пару. К ним добавлены индексы (+) и (–).

Второй вариант требует, чтобы цепи, которые предполагается объединить в дифференциальную пару, имели общее имя и различались суффиксами **_H** и **_L**. При данном условии можно сформировать дифференциальные пары из цепей с такими именами.

Для этого следует активизировать диалог формирования щелчком в панели **PCB** на кнопке **Create From Nets**. Откроется диалоговое окно **Create Differential Pairs from Nets** (рис. 6.18).

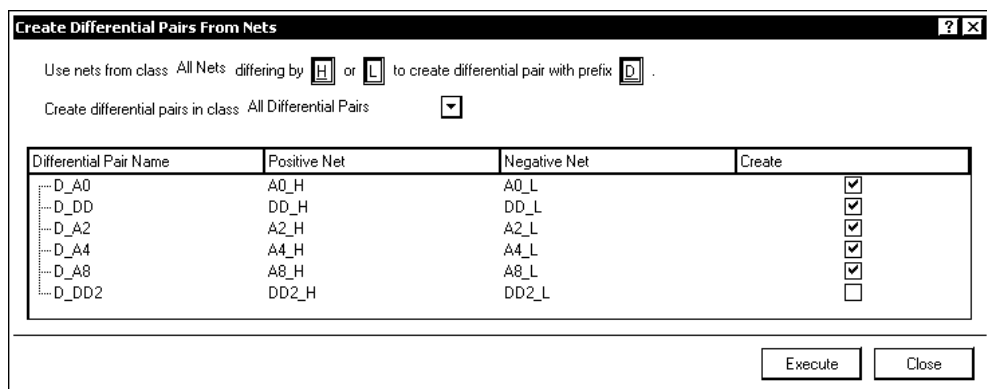


Рис. 6.18. Составление дифференциальных пар из цепей с совпадающими именами и суффиксами **_H** и **_L**

В верхней части окна выводится предложение: **Use nets from class All Nets differing by H or L to create differential pair with prefix D**. Приведем перевод этого сообщения с некоторыми комментариями: "Используйте цепи из класса *Все цепи* (это класс, существующий по умолчанию; в проекте могут быть назначены и другие классы цепей), различающиеся (суффиксами) **_H** или **_L**, для образования дифференциальной пары с префиксом (к имени) **D**".

Ниже располагается еще одно сообщение: **Create differential pairs in class All Differential Pairs** — "Образуйте дифференциальные пары в классе *Все дифференциальные пары*". Справа располагается кнопка вызова списка классов дифференциальных пар, если такие классы назначены.

В центральном поле окна выведен список всех существующих пар цепей с именами, имеющими суффиксы `_N` и `_L`, и список дифференциальных пар, которые могут быть составлены из этих цепей.

Для формирования дифференциальных пар нужно поставить флажки в колонке **Create** (Создать) против имен соответствующих пар и завершить диалог щелчком на кнопке **Execute** (Выполнить).

6.7.2. Правила трассировки дифференциальных пар

Правила трассировки дифференциальных пар настраиваются так же, как и для всех остальных элементов проводящего рисунка печатной платы.

Перечислим ключевые правила для дифференциальных пар.

- ☐ **Routing Width** — ширина печатных проводников дифференциальной пары. Настройка правила для ширины печатных проводников рассматривалась в *разд. 5.3* (см. рис. 5.4).
- ☐ **Differential Pairs Routing** — трассировка дифференциальной пары. Это правило касается главным образом зазора между проводниками дифференциальной пары и настраивается в диалоге, активизируемом по команде главного меню **Design | Rules**. В диалоговом окне **PCB Rules and Constraints Editor** (рис. 6.19) выбрать ветвь **Design Rules | Routing | DiffPairsRouting** и назначить в поле **Constraints** следующие параметры:
 - **Min Gap** — минимальный зазор;
 - **Max Gap** — максимальный зазор;
 - **Preferred Gap** — предпочтительное значение зазора;
 - **Max Uncoupled Length** — максимально допустимая длина участка обхода препятствий, на котором уменьшается индуктивная связь проводников пары.
- ☐ **Matched Net** — выравнивание длины проводников в паре добавлением в короткий проводник лесенки или змейки. Данная настройка доступна в ветви **High Speed/Matched Net Lengths** дерева правил проектирования. Она также вызывается из плавающей панели **PCB** щелчком на кнопке **Rule Wizard** (см. рис. 6.16). Вызывается "Мастер подсказки", на одном из шагов которого открывается диалог настройки **DiffPair_MatchedLengths** — выравнивание длины проводников в дифференциальной паре (рис. 6.20).

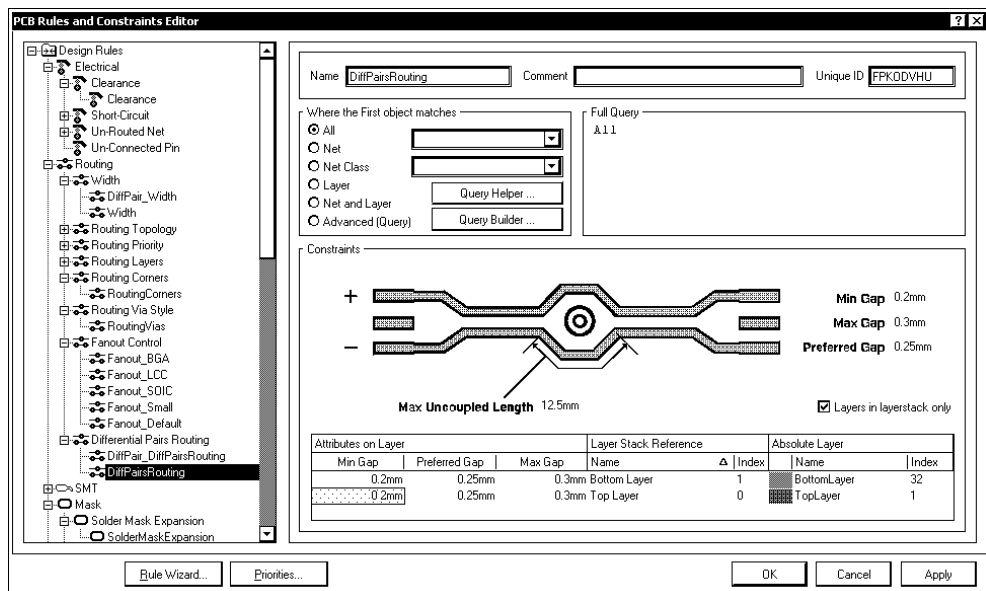


Рис. 6.19. Диалог настройки правил разводки дифференциальных пар

- ❑ В поле **Rule Properties** (Параметры правила) назначают:
 - **Tolerance** — допуск, при превышении которого строится удлиняющая конфигурация;
 - **Gap** — зазор между коленами;
 - **Amplitude** — высота колен;
 - **Style** — стиль построения удлиняющей конфигурации: прямоугольными коленами, наклон колен под углом 45° или закругленными сегментами.
- ❑ В поле **Rule Priority** (Приоритет правила) выводится список всех назначенных правил с параметрами их настройки и указанием объектов, на которые налагаются эти правила, и приоритетов их выполнения. Кнопками **Increase Priority** (Повысить приоритет) и **Decrease Priority** (Понизить приоритет) приоритет выбранного в списке правила можно увеличить или уменьшить.

Объекты, на которые налагается то или иное правило, указывают с помощью рассмотренной ранее процедуры формирования запросов (см. разд. 5.3).

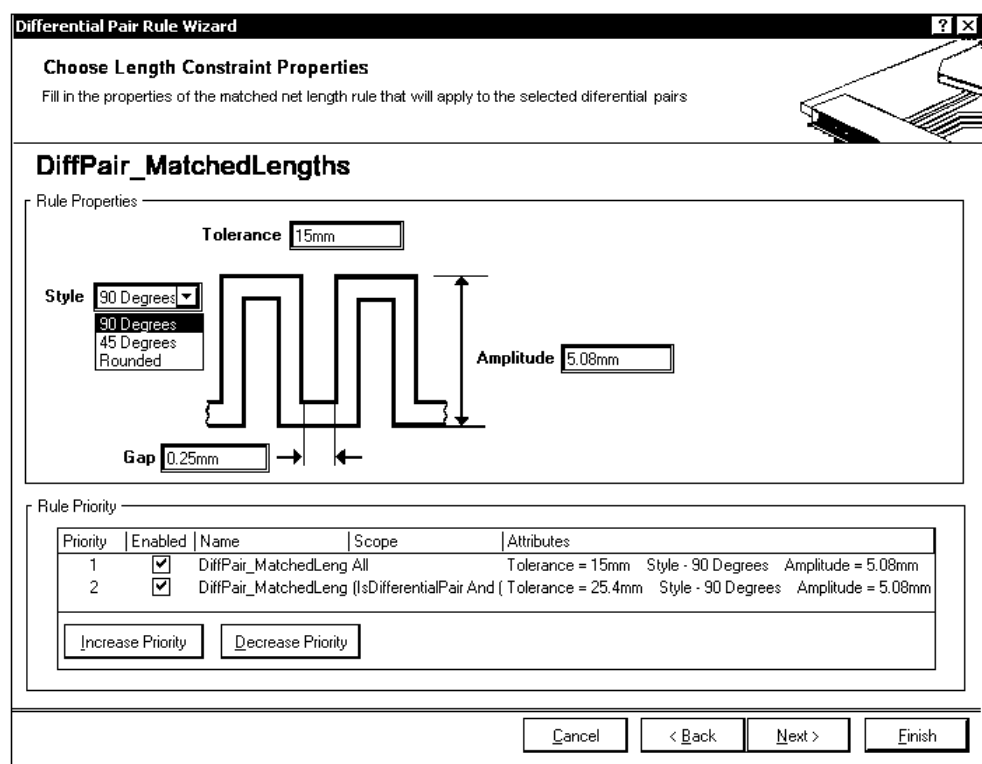


Рис. 6.20. Выравнивание длины проводников дифференциальной пары

6.7.3. Трассировка дифференциальной пары

Трассировка дифференциальной пары ведется точно так же, как и рассмотренная ранее Smart-интерактивная трассировка. Отличие лишь в том, что проводятся сразу оба печатных проводника дифференциальной пары.

1. Выделяют маскированием или цветом линии связи дифференциальной пары, подлежащей трассировке (см. пп. 1–3, с. 133–136).
2. Активируют команду главного меню **Place | Differential Pair Routing**. В строке статуса программы выводится предложение выбрать первый объект дифференциальной пары: **Choose first differential pair object to route**.
3. Наводят курсор на один из контактов компонента, с которого начинается трассировка дифференциальной пары, и щелчком левой кнопки мыши

дают старт прокладке трассы. Программа сводит проводники дифференциальной пары до расстояния, определенного в правилах, и далее ведет оба проводника параллельно (рис. 6.21).

Как и в Smart-трассировке, при трассировке дифференциальной пары работают "горячие" клавиши и их комбинации, в частности:

- клавишей <Пробел> или комбинацией клавиш <Shift>+<Пробел> можно выбрать способ построения изломов трассы под углом 45° или 90°;
 - комбинацией клавиш <Shift>+<R> выбирают способ разрешения конфликтов (остановка трассы, игнорирование или обход препятствий);
 - клавишей <*> на цифровой клавиатуре меняют слой трассировки.
4. Программа строит сегменты трассы от начала до текущего положения курсора сплошной линией, ширина которой указана в правилах, и продолжает предполагаемую оставшуюся часть трассы штриховой линией.
 5. Прокладку оставшейся части трассы завершают щелчком левой кнопки мыши при нажатой клавише <Ctrl>. Программа остается в режиме трассировки дифференциальных пар.
 6. Щелчком правой кнопки мыши или клавишей <Esc> окончательно завершают трассировку.

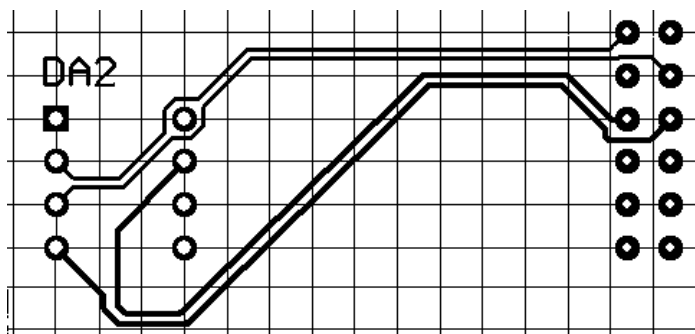


Рис. 6.21. Пример разводки дифференциальных пар

6.8. Автоматическая трассировка печатного монтажа

В состав пакета Altium Designer входит встроенный высокоэффективный топологический автотрассировщик Situs (от лат. *situs* — место). В отечествен-

ной науке употребляют термин топология (от греч. *topos* — место) — раздел математики, исследующий свойства геометрических объектов.

Работа автотрассировщика заключается в анализе топологии платы, подготовленной к разводке печати, — в определении очертаний размещенных на плате компонентов, отверстий, вырезов, других препятствий, поиске пути (канала) для прокладки каждого очередного печатного проводника и затем в применении целого ряда алгоритмов (проходов), в результате чего в найденном канале размещается печатный проводник, отвечающий назначенным при настройке графического редактора печатной платы правилам. Топологические алгоритмы автотрассировки считаются более эффективными, чем сеточные и даже так называемые бессеточные, нейронные алгоритмы. Здесь мы не имеем возможности обсуждать эти преимущества. Рассмотрим основные приемы настройки и практического использования автотрассировщика Situs¹.

6.8.1. Стратегии и проходы автотрассировки

Набор приемов (проходов) автотрассировки в Altium Designer, как и в целом ряде других САПР, называется стратегией трассировки. Основу стратегии, применяемой к текущему проекту, составляют правила проектирования, назначаемые на этапе составления электрической принципиальной схемы (см. *разд. 4.2.6*) и (или) на этапе проектирования печатной платы (см. *разд. 5.3–5.5*).

Сначала нужно командой главного меню **Auto Route | Setup** (Автотрассировка | Настроить) или **Auto Route | All** (Автотрассировать | Все) активизировать диалог настройки стратегии трассировки. Автотрассировщик Situs анализирует назначенные правила и выстраивает их в стратегию. В поле **Routing Strategies** диалогового окна **Situs Routing Strategies** (рис. 6.22) приводится список доступных в текущий момент стратегий и их краткая характеристика.

Всего по умолчанию предусмотрено шесть стратегий:

- ☐ **Cleanup** — подчистка выполненной разводки;
- ☐ **Default 2 Layer Board** — трассировка по умолчанию двухсторонней печатной платы;
- ☐ **Default 2 Layer with Edge Connectors** — трассировка по умолчанию двухсторонней платы с печатными контактами краевого соединителя;

¹ Документ HELP-системы Altium Designer: Situs Autorouting Essentials AR0128 (v1.1) July 29, 2005.



Рис. 6.22. Настройка выбора стратегии
автотрассировщика Situs

- ☐ **Default Multi Layer Board** — стратегия трассировки по умолчанию многослойной печатной платы;
- ☐ **General Orthogonal** — основная стратегия с взаимно-перпендикулярным направлением прокладки проводников;
- ☐ **Via Miser** — стратегия с жесткой минимизацией числа переходных отверстий.

Каждая из predetermined стратегий включает свой определенный набор проходов трассировки. Редактирование этих "чистых" стратегий не допускается, однако если при выполнении проекта необходимо изменить набор приемов (проходов) трассировки, можно образовать новую пользовательскую стратегию.

Если за основу новой решено принять одну из "чистых" стратегий, то создать свою стратегию можно щелчком на кнопке **Duplicate** (Дублировать). Открывается диалоговое окно редактирования стратегии **Situs Strategy Editor** (рис. 6.23).

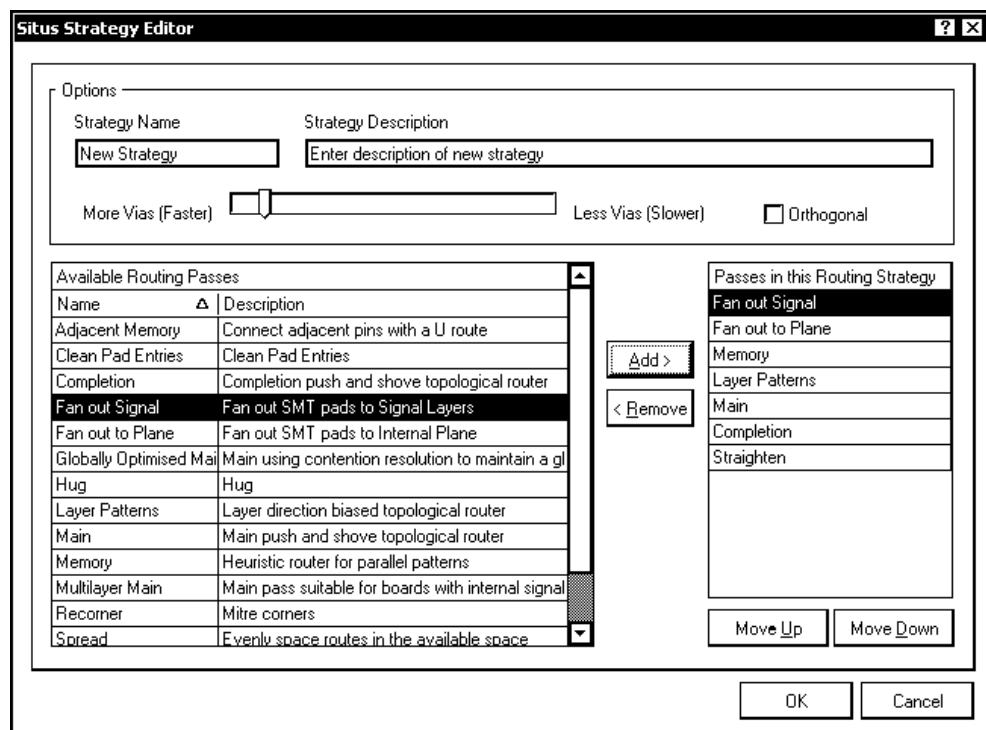


Рис. 6.23. Окно редактирования стратегии
автотрассировщика Situs

В поле **Options** этого окна:

- ☐ ввести имя новой стратегии (**Strategy Name**) и ее краткое описание (**Strategy Description**);

- ❑ установить движок в линейке **More Vias (Faster) ... Less Vias (Slower)**, задавая большее или меньшее число межслойных переходных отверстий, что в свою очередь отразится на ускорении или замедлении трассировки;
- ❑ при необходимости активизировать опцию **Orthogonal** — трассировка с изломами проводников под прямым углом.

В двух нижележащих полях приводится список возможных проходов авто-трассировки (**Available Routing Passes**) и проходов, включенных в текущую стратегию (**Passes in this Routing Strategy**). Кнопками **Add** > и < **Remove** указанные курсором проходы переносят из одного поля в другое, в результате в правом поле составляется необходимый список проходов для редактируемой стратегии.

Приведем обзор доступных проходов трассировки.

- ❑ **Main** — "главный" тип прохода, использующий топологическую карту печатной платы для поиска пути прокладки трассы и функцию раздвижки и проталкивания (**Push and Shove**) для превращения предполагаемого пути в реальный печатный проводник.
- ❑ **Multilayer Main** — тот же "главный" тип прохода, но выполняемый при разводке многослойных ПП, с учетом цены (системы штрафов и поощрений, заложенной в алгоритме), назначенной для многослойной трассировки.
- ❑ **Globally Optimized Main** — трассировка с оптимизацией, игнорирующая столкновения трасс и препятствий на первой итерации и затем итеративно выполняющая перетрассировку с увеличенной ценой ошибки до тех пор, пока не будут разрешены все конфликты.
- ❑ **Completion** — тот же тип **Main**, но с измененной ценой конфликта на завершающей стадии прокладки сложных соединений.
- ❑ **Memory** — регулярная трассировка связей между контактами компонентов с одинаковыми координатами по оси *X* или *Y*, по типу линий адреса (данных) в устройствах памяти.
- ❑ **Adjacent Memory** — прокладка U-образных трасс между контактами соседних компонентов.
- ❑ **Clean Pad Entries** — уход трассы от контакта вдоль длинной стороны контактной площадки.
- ❑ **Fan out Signal** — разводка веером на сигнальном слое под управлением функций **Fanout Control** в дереве правил проектирования (см. разд. 5.3).

- ❑ **Fan out to Plane** — разводка веером с подключением связей к внутренним слоям металлизации типа Plane.
- ❑ **Hug** — трассировка со сжатием трасс до минимально допустимого значения зазоров, указанного в правилах проектирования.
- ❑ **Layer Patterns** — трассировка связей, для которых указаны направления разводки на слоях.
- ❑ **Recorner** — сглаживание прямоугольных изломов печатных проводников — выполняется по умолчанию, в том числе и при активной опции ортогональной трассировки.
- ❑ **Spread** — трассировка с расширением зазоров и равномерным распределением проводников по площади платы.
- ❑ **Straighten** — трассировка со спрямлением проводников и минимизацией длины трассы.

Назначить и редактировать новую стратегию можно щелчком на кнопке **Add** (Добавить). Вся процедура подготовки новой стратегии полностью идентична рассмотренной для дублированной стратегии.

6.8.2. Правила и отчеты автотрассировщика

В окне настроек автотрассировщика **Situs Routing Strategies** (см. рис. 6.22) размещено поле отчета о настройке правил **Routing Setup Report**, касающихся трассировки печати (ширина проводников, зазоры между элементами печатного рисунка, топология проводников, стиль разводки веером и др.). Прокручивая этот список, можно получить в поле **Routing Setup Report** подробные сведения по каждому правилу (рис. 6.24).

При указании курсором на имя правила со щелчком левой кнопки мыши запускается процедура редактирования выбранного правила (на рис. 6.24 — ширина печатных проводников). Открывается диалоговое окно, аналогичное правому полю окон на рис. 5.4 или 6.19. По контексту в этом окне представлены функции настройки выбранного правила, включая генератор запросов на установку приоритетов.

Редактирование правил также можно активизировать щелчком на кнопке **Edit Rules**.

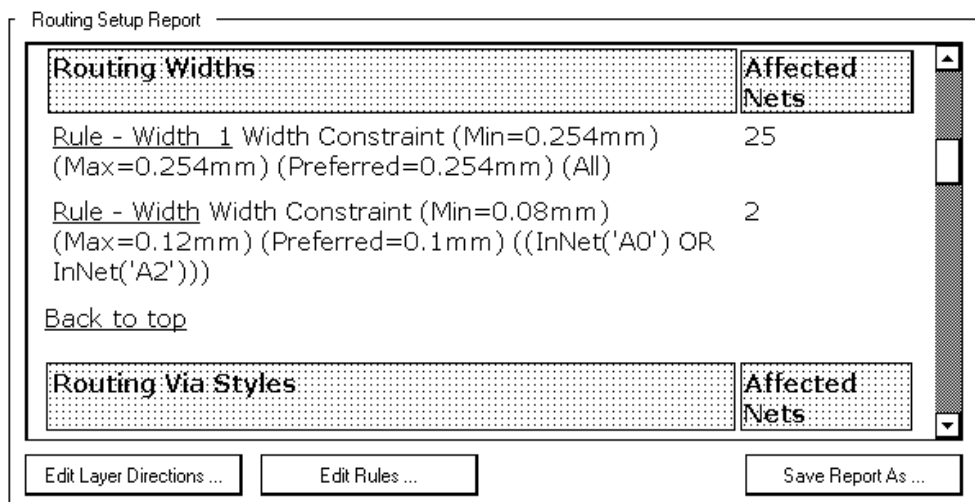


Рис. 6.24. Окно отчета о назначенных правилах автотрассировки

6.8.3. Автотрассировка

Заложенные в Altium Designer функции автотрассировки доступны из главного меню программы по команде **Auto Route** и ее подкомандам (рис. 6.25).

Рассмотрим основные функции автотрассировки.

- ❑ **All** — трассировка всей платы: выполняется после настройки стратегии по щелчку на кнопке **Route All** в диалоговом окне на рис. 6.22;
- ❑ **Connection** — трассировка связи, соединяющей контакты двух компонентов (остальные участки цепи не трассируются). При указании курсором на контакт, от которого связь уходит в одну сторону, трассировка выполняется автоматически по щелчку левой кнопки мыши. При указании курсором контакта, от которого связь расходится в разные стороны, открывается диалоговое окно, в котором приводится перечень сегментов выбранной связи и графическая иллюстрация их расположения на плате (рис. 6.26, а);
- ❑ **Net** — трассировка выбранной цепи. Выполняется по двойному щелчку левой кнопкой мыши после указания курсором на начальный или конечный контакт, принадлежащий данной цепи. При указании на контакт, от которого линии связи расходятся в разные стороны, открывается диало-

говое окно, рис. 6.26, б, в котором надлежит выбрать один из сегментов электрической связи, образующей цепь. Цепь разводится целиком;

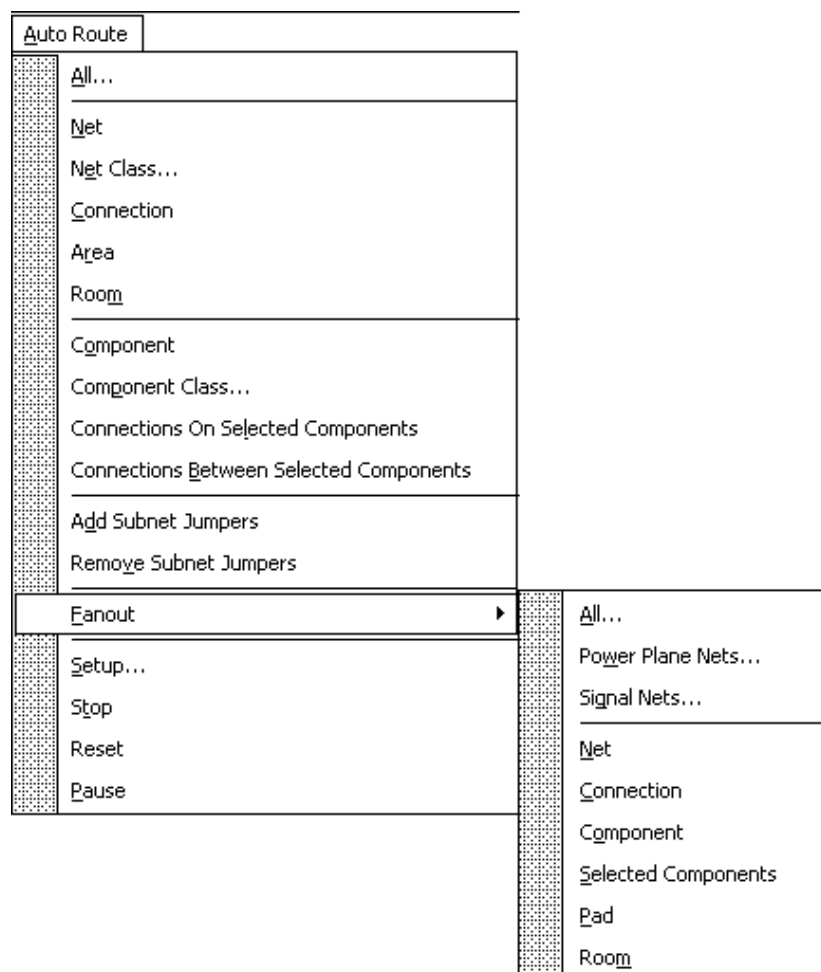
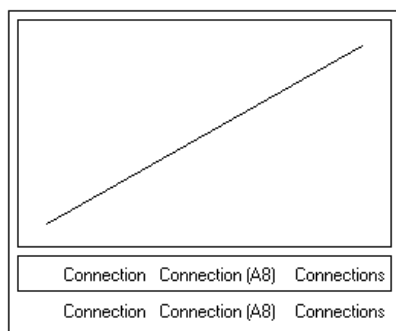
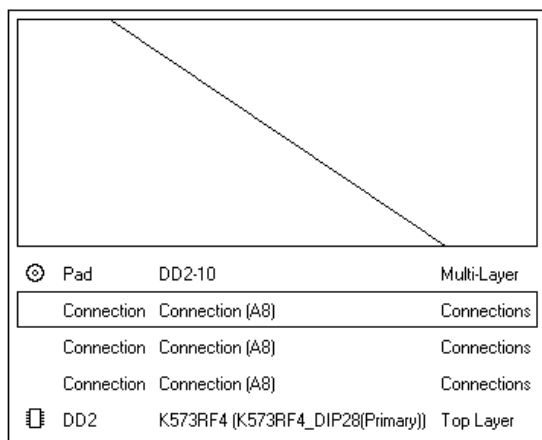


Рис. 6.25. Меню команд автотрассировки

- ❑ **Net Class** — трассировка выбранного класса цепей. Активизируется диалоговое окно **Choose Net Classes to Route** (выбрать классы цепей, подлежащие трассировке) (рис. 6.27, а). Указать курсором один или (удерживая клавишу <Shift>) несколько классов и дать старт трассировке щелчком на кнопке **OK**;



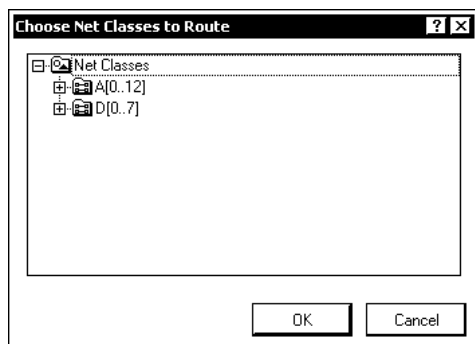
a



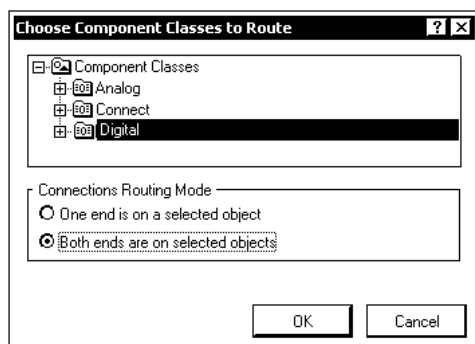
б

Рис. 6.26. Окна выбора объекта для автотрассировки:
a — линии связи (Connection); б — цепи (Net)

- ❑ **Area** — трассировка связей, полностью укладываемых в пределы прямоугольной области, обозначаемой курсором (связи, выходящие за пределы области, не разводятся);



a



б

Рис. 6.27. Окна выбора объекта для автотрассировки:
a — класса цепей; б — компонента

- ❑ **Room** — трассировка связей, укладываемых в область **Room** (связи, выходящие за пределы "комнаты", не разводятся);
- ❑ **Component** — трассировка связей одного выбранного компонента;

- ❑ **Component Class** — трассировка связей компонентов выбранного класса. Выполняется аналогично трассировке класса цепей — при указании курсором на один или несколько классов компонентов в окне (рис. 6.27, б);
- ❑ **Connections on Selected Components** — трассировка связей выбранных компонентов, как между собой, так и с остальными. Компоненты нужно предварительно выделить курсором при нажатой клавише <Shift>;
- ❑ **Connections Between Selected Components** — трассировка связей только между выбранными компонентами. Аналогично предыдущему случаю, компоненты также сначала должны быть выделены;
- ❑ **Fanout** — трассировка веером от тесно расположенных контактов поверхностно-монтируемых (SMD) компонентов. Команда имеет несколько подкоманд:
 - **All** — разводка веером всех связей;
 - **Power Plane Nets** — разводка веером цепей, уходящих в слои питания/"земли", класса Plane;
 - **Signal Nets** — разводка веером сигнальных цепей;
 - **Net** — разводка выбранной цепи;
 - **Connection** — разводка выбранной связи;
 - **Component** — разводка от контактов выбранного компонента;
 - **Selected Components** — разводка от контактов выбранной группы компонентов;
 - **Pad** — разводка от выбранной контактной площадки;
 - **Room** — разводка от SMD-компонентов в пределах области **Room**.

Правила веерной трассировки назначаются в ветви дерева правил проектирования **Design Rules | Routing | Fanout Control** (Правила проектирования | Трассировка | Управление веерной разводкой). По умолчанию назначено четыре правила (рис. 6.28) для SMD-компонентов разных типов:

- ❑ **LCC** (Leadless Chip Component) — безвыводных чипов;
- ❑ **BGA** (Ball Grid Array) — в корпусах с матрицей шариковых выводов;
- ❑ **SOIC** (Small Outline Integrated Circuit) — в малогабаритных корпусах с планарными выводами;
- ❑ **Small** — в малогабаритных корпусах с числом выводов менее пяти.

Область приложения правил назначается с помощью процедуры формирования запросов (**Query**) и отображается в колонке **Scope** таблицы, приведенной на рис. 6.28. Приоритеты правил устанавливаются в зависимости от наличия компонентов соответствующего типа в проекте.

Правило **Fanout Default** можно распространить на все SMD-компоненты проекта. В этом случае данному правилу назначается низший приоритет, а для области распространения формируется запрос **IsSMTComponent**.






Name	Δ	Priority	Enabled	Type	Category	Scope	Attributes
 Fanout_BGA		1	<input checked="" type="checkbox"/>	Fanout Con Routing	IsBGA	IsBGA	Style - Auto Direction - Alternating In and Out Via Grid = 0.025mm
 Fanout_Default		5	<input checked="" type="checkbox"/>	Fanout Con Routing	All	All	Style - Auto Direction - Alternating In and Out Via Grid = 0.025mm
 Fanout_LCC		2	<input checked="" type="checkbox"/>	Fanout Con Routing	IsLCC	IsLCC	Style - Auto Direction - Alternating In and Out Via Grid = 0.025mm
 Fanout_Small		4	<input checked="" type="checkbox"/>	Fanout Con Routing	(CompPinCount < 5)	(CompPinCount < 5)	Style - Auto Direction - Out Then In Via Grid = 0.025mm
 Fanout_SOIC		3	<input checked="" type="checkbox"/>	Fanout Con Routing	IsSOIC	IsSOIC	Style - Auto Direction - Alternating In and Out Via Grid = 0.025mm

Рис. 6.28. Правила и приоритеты веерной автотрассировки

Программа анализирует структуру посадочного места компонента (типы и расположение контактных площадок) и присваивает компоненту внутренний, невидимый пользователю и недоступный для редактирования атрибут вида **IsBGA = true**, **IsSOIC = true** и т. п. В соответствии с логическим значением этого атрибута строится конфигурация веерных отводов от контактов таких компонентов.

Веерную трассировку обычно следует выполнять в первую очередь, чтобы можно было оценить и исправить ее результаты, прежде чем выполнять прочие проходы автотрассировки. Команда **Auto Route | Fanout | Component** позволяет реализовать веерные отводы, заканчивающиеся межслойным переходным отверстием, от компонентов, не соединенных электрическими связями. Если скопировать такую конфигурацию в библиотеку посадочных мест, получатся предварительно разведенные веером SMD-компоненты, которые будут доступны в проекте печатной платы.

Все предварительно разведенные вручную цепи — дифференциальные пары, веерная разводка планарных (SMD) компонентов, другие критические цепи, можно защитить от автотрассировки. Для этого следует активизировать в диалоговом окне (см. рис. 6.22) опцию **Lock All Pre-routes**.

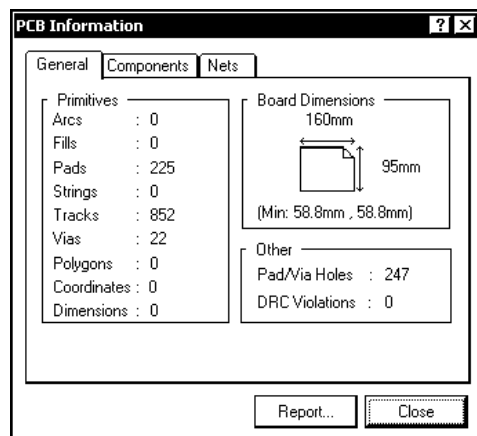
После выполнения всех необходимых настроек дают старт автотрассировке кнопкой **Route All** в окне рис. 6.22. Трассировка сложной платы с сотнями компонентов и плотной компоновкой может длиться до нескольких часов. По ходу выполнения трассировки в плавающую панель **Messages** выводятся со-

общения о завершении проходов трассировки и затраченном на это времени. Трассировку можно приостановить командой **Auto Route | Stop** и возобновить повторением команды **Auto Route | All**.

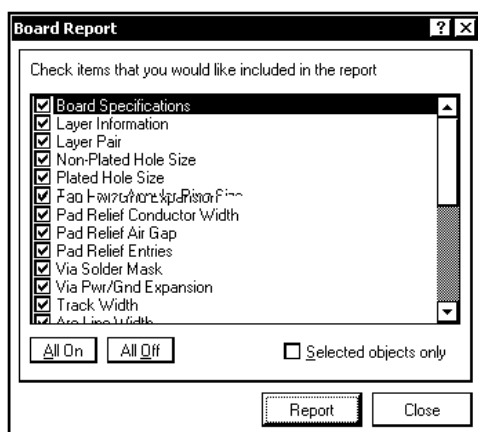
Стопроцентная разводка печати не гарантирована. В случае недоразводки некоторых цепей приходится менять размещение компонентов, настройку правил и выполнять повторную автотрассировку или ручную трассировку неразведенных цепей. Человеку часто удается то, что недоступно программным средствам САПР.

6.8.4. Отчет автотрассировщика

Команда главного меню **Reports | Board Information** активизирует диалоговое окно **PCB Information** (рис. 6.29, а) с тремя панелями-вкладками, в которых приводятся сводка общих характеристик платы, списки компонентов и цепей проекта.



а



б

Рис. 6.29. Окна настройки отчета автотрассировщика:

а — сводка объектов печатной платы; б — настройка содержания отчета

Кнопка **Report** открывает окно следующего уровня с полным списком объектов платы, о которых можно получить отчет (рис. 6.29, б).

Кнопка **Report** в окне на рис. 6.29, б дает старт формированию отчета о параметрах выбранных объектов (рис. 6.30).

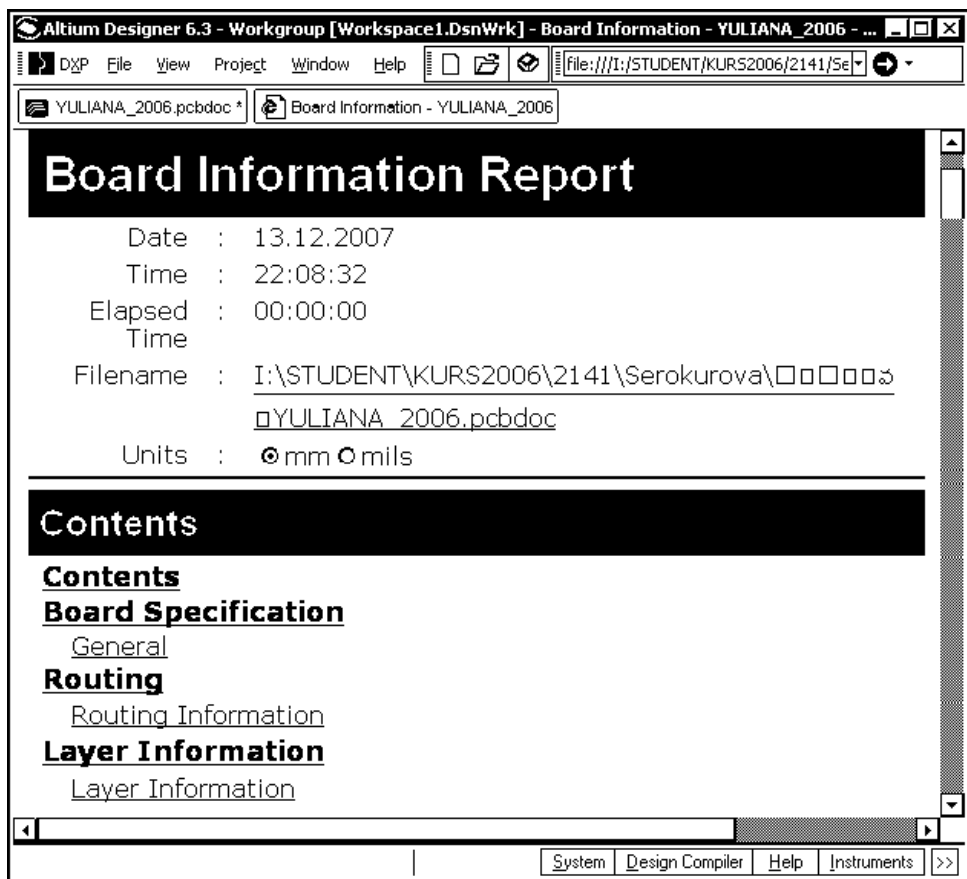


Рис. 6.30. Окно отчета автотрассировщика Situs

Через систему гиперссылок можно просмотреть подробности, касающиеся отдельных структурных компонентов проекта (геометрические размеры платы, параметры слоев, проводников, процент разведенных и неразведенных цепей и др.).

6.9. Верификация PCB-проекта

Команда главного меню **Tools | Design Rule Check** (Инструменты | Контроль правил проекта) запускает проверку выполнения правил проектирования, установленных для текущего открытого проекта. Открывается диалоговое

окно **Design Rule Checker** (рис. 6.31), в полях которого настраивают опции текущей (Online) и пакетной (Batch) проверок.

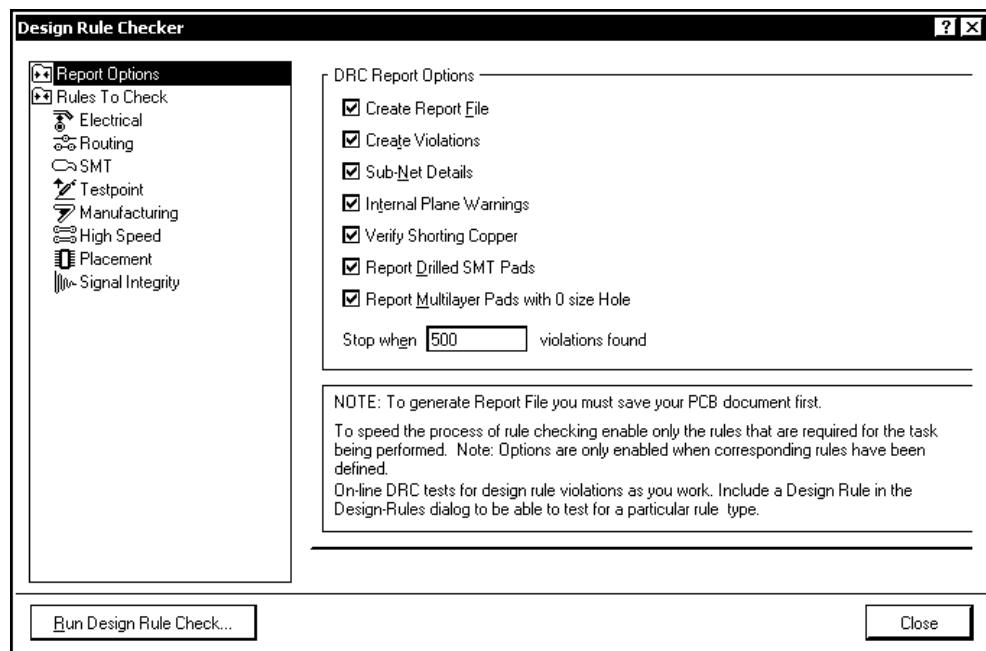


Рис. 6.31. Диалог настройки пакетного DRC-контроля

В правом поле окна **DRC Report Options** устанавливаются области для проверки и формирования отчетов:

- ☐ **Create Report File** — создать файл отчета;
- ☐ **Create Violations** — формировать отчет о нарушениях;
- ☐ **Sub-Net Details** — детальная информация о сегментах цепей;
- ☐ **Internal Plane Warnings** — предупреждения, связанные с внутренними слоями класса Plane;
- ☐ **Verify Shorting Copper** — проверка на короткие замыкания участков металлизации;
- ☐ **Report Drilled SMT Pads** — отчет о просверленных планарных КП;
- ☐ **Report Multilayer Pads with 0 size Hole** — отчет о многослойных КП с нулевым диаметром монтажного отверстия.

В левом поле окна на рис. 6.31 приводится список категорий правил, выполнение которых подлежит проверке:

- ☐ **Electrical** — электрические (зазоры, короткие замыкания и др.);
- ☐ **Routing** — трассировочные (ширина проводников, стиль ПО и др.);
- ☐ **SMT** — правила работы с планарными компонентами;
- ☐ **Testpoint** — контрольные точки;
- ☐ **Manufacturing** — правила, влияющие на возможность изготовления платы (изломы печатных проводников под острым углом, ширина кольца металлизации КП и др.);
- ☐ **High Speed** — правила проектирования высокоскоростных устройств;
- ☐ **Placement** — правила размещения компонентов;
- ☐ **Signal Integrity** — правила контроля целостности сигналов (волновое сопротивление, время распространения сигналов, фронты, выбросы и др.).

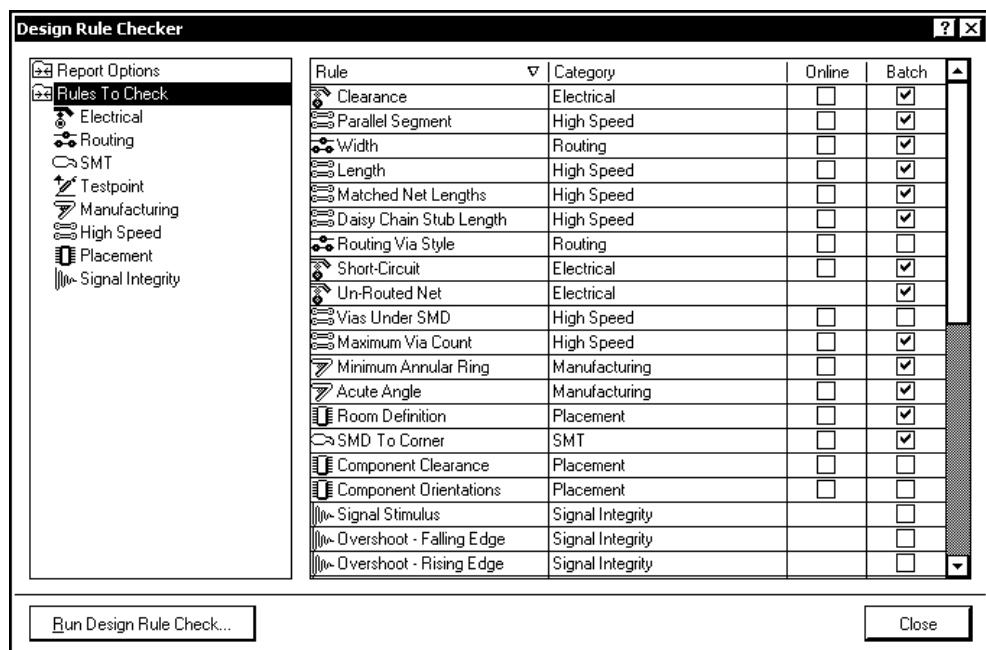


Рис. 6.32. Сводка правил DRC-контроля

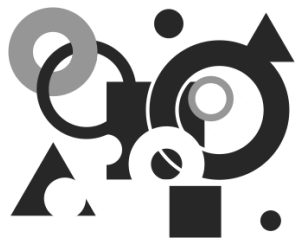
При указании курсором одной из категорий в левом поле окна (рис. 6.31) правое поле видоизменяется: в нем разворачивается список правил выбранной категории.

При указании в левом поле окна строки **Rules to Check** разворачивается полный список правил по всем категориям (рис. 6.32).

В колонках **Online** и **Batch** установкой флажков активизируются правила, текущая и пакетная проверка которых должна выполняться в активном проекте.

Кнопка **Run Design Rule Check** запускает пакетную проверку. В процессе проверки в плавающей панели **Messages** появляются сообщения о ходе выполнения и результатах проверки по всем пунктам назначенных правил. По окончании проверок в главном окне программы выводится итоговый отчет, озаглавленный **Design Rule Verification Report** (отчет о верификации правил проектирования).

Все обнаруженные нарушения отмечаются на плате цветовой подсветкой зеленого цвета. Полученный отчет и графическая иллюстрация позволяют принять решение о необходимости редактирования проекта — изменения размещения компонентов, ручного редактирования трассировки, выполнения повторной автотрассировки, редактирования правил, приоритетов и т. п., что в конечном итоге даст результат, отвечающий техническому заданию.



Проект ПЛИС

Достижения технологии интегральных микросхем обеспечили такой рост степени интеграции, что появилась возможность реализовать в одной микросхеме сложную цифровую систему с так называемой жесткой логикой, ориентированную на выполнение единственной задачи обработки данных. Такой интегральной микросхеме в западной инженерной лексике соответствует аббревиатура ASIC — Application-Specific Integrated Circuit (интегральная схема, ориентированная на конкретное приложение).

Дальнейшее развитие технологии и увеличение степени интеграции привело к разработке микро-ЭВМ и микроконтроллеров, включающих встроенную память, узлы логической, арифметической обработки и синхронизации. Подобные устройства можно запрограммировать для выполнения задач цифрового управления процессами в широком спектре инженерных приложений.

В обоих этих случаях внутренняя структура таких микросхем является жесткой, не подлежащей модификации применительно к приложению. Они нужны только во вводе данных в необходимом формате и выводе результата обработки на исполнительные или индикаторные устройства. Конструкторское проектирование функциональных узлов на основе таких микросхем не отличается от рассмотренного в предыдущих главах книги выполнения проекта печатной платы (PCB Project).

Параллельно с индустрией ASIC-микросхем и микроконтроллеров развивалась индустрия микросхем, функции которых пользователь программирует под конкретное приложение. Внутренняя структура этих микросхем представляет собой матричное поле элементарных логических вентилей или логических блоков, входы и выходы которых выходят на внутренние магистрали и могут коммутироваться под воздействием электрических сигналов,

образуя логическую структуру, обеспечивающую выполнение задачи пользователя. Идеология такого построения логических и вычислительных устройств восходит к началу 60-х гг. XX столетия [3], когда на рынке только появились первые интегральные микросхемы, а микросхем высокой степени интеграции еще не существовало. С развитием технологии микросхем появилась возможность практической реализации этих идей. В западной инженерной лексике интегральные схемы программируемой логики обозначают по-разному, в зависимости от технологии выполнения коммутации на внутренних магистралях микросхемы:

- ❑ CPLD (Complex Programmable Logic Device) — сложные программируемые логические устройства, исторически появившиеся первыми, как правило, с одноразовым программированием по типу "прожига" ПЗУ;
- ❑ EPLD (Erasable Programmable Logic Device) — программируемые логические устройства, в которых действующая программа при перепрограммировании стирается ультрафиолетовым излучением и/или электрическим сигналом;
- ❑ FLASH PLD — микросхемы, внутренняя коммутация которых может многократно перепрограммироваться с использованием flash-технологии;
- ❑ FPGA (Field Programmable Gate Array) — микросхема, внутренняя коммутация в которой перепрограммируется под воздействием электростатического поля на МДП-транзисторы с изолированным затвором, выполненные на пересечениях линий внутренних магистралей.

В нашей стране был принят схожий по смыслу термин ПЛИМ (программируемая логическая матрица), который в последние годы уступил место новому — ПЛИС (программируемая логическая интегральная схема). В этих аббревиатурах отсутствует указание на технологию выполнения внутренней коммутации.

Следует также отметить, что программируемой может быть не только цифровая, но и аналоговая или аналого-цифровая микросхема. Некоторые производители, в частности фирма Cypress Semiconductors (США), выпустили на рынок и такие микросхемы.

В настоящий момент достижения в технологии программируемых интегральных микросхем настолько высоки, что позволяют в одном кристалле ИС реализовать сложные устройства обработки информации, включающие процессорное ядро, постоянную и оперативную память, узлы интерфейсной логики и управления периферийными устройствами.

Разработка проектов на основе ПЛИС невозможна без развитых программных средств САПР, предназначенных для программирования внутренней структуры микросхемы под задачу пользователя. Такие средства были разработаны ведущими производителями ПЛИС — фирмами Altera, Atmel, Actel, Xilinx — каждой под свое семейство ПЛИС. Эти системы проектирования обладают развитыми средствами ввода исходных данных:

- ❑ привычный радиоинженеру графический ввод данных в среде встроенного схемного редактора в виде логической схемы устройства (Graphical Design File);
- ❑ задание временных диаграмм логических сигналов на входах и выходах устройства в среде встроенного редактора "форм волны" (Waveform Editor);
- ❑ описание логики на специально разработанных HDL-языках программирования высокого уровня — так называемых языках описания аппаратуры (Hardware Description Languages);
- ❑ комбинированный ввод: часть данных вводится в виде схемы, часть в виде временных диаграмм, часть — в конструкциях HDL-языка;
- ❑ описание логики в формате специального языка обмена данных с "электронными" САПР общего назначения (P-CAD, OrCAD и др.).

Все эти специализированные средства разработки и программирования внутренней логики ПЛИС [4] не позволяют выполнить сквозной проект радиоэлектронного функционального узла с использованием ПЛИС, поскольку конечным результатом их работы чаще всего являются карта разводки входных и выходных логических сигналов на электрические контакты выбранной микросхемы ПЛИС, результаты моделирования логики и файлы управляющей информации для компьютера или для специализированного программатора, под действием которого выполняется коммутация внутренних логических блоков ПЛИС.

Производители "электронных" САПР общего назначения предпринимают попытки разработать интегрированную среду, обеспечивающую сквозное проектирование радиоэлектронного функционального узла на ПЛИС. В качестве примеров могут быть упомянуты интегрированные САПР Mentor Graphics Leonardo Spectrum, Allegro фирмы Cadence Design Systems (развитие OrCAD).

Такую же интегрированную среду представляет рассматриваемая здесь САПР Altium Designer. Эта среда позволяет образовать иерархию проектов, на верхнем уровне которой находится проект печатной платы (PCB-проект).

В рамках РСВ-проекта определяется "окружение" ПЛИС — компоненты, которые необходимы для нормальной работы проектируемого функционального узла, но не включаются в состав имеющихся на рынке микросхем ПЛИС (кварцевые резонаторы задающих генераторов, резисторы, конденсаторы, микросхемы АЦП, конфигурационные ПЗУ, электрические соединители и т. п.). Поскольку концепция ПЛИС предполагает, что пользователь программирует внутреннюю логику кристалла под свою задачу, библиотечные элементы ПЛИС поставляются с пакетами САПР в виде "болванок": на схемном УГО (рис. 7.1) обозначены только цоколевочные номера и электрический тип контактов микросхемы (входные, выходные, выводы для подключения питания и "земли", входы тактового сигнала, входы для загрузки программы коммутации внутренней логики ПЛИС и т. п.).

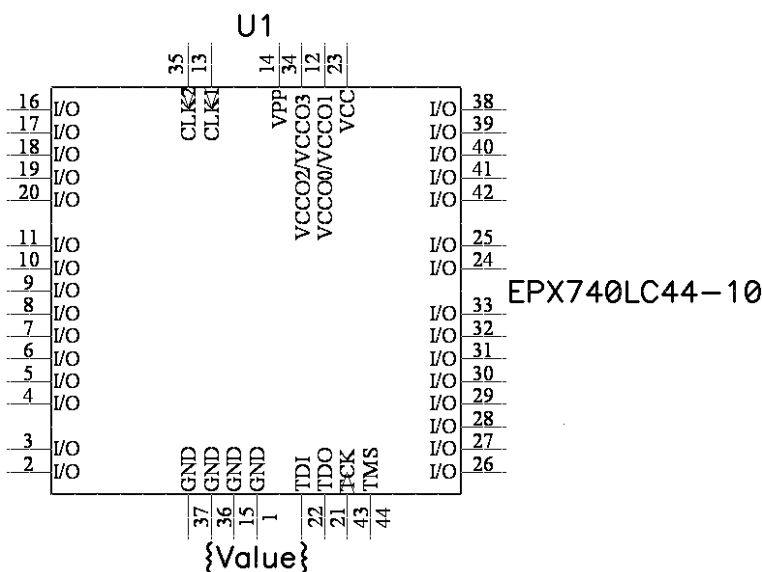


Рис. 7.1. Пример схемного символа ПЛИС фирмы Altera

Таким образом, уже на начальной стадии разработки РСВ-проекта наступает момент, когда следует выполнить подчиненный FPGA-проект внутренней логической структуры ПЛИС. Заложенные в Altium Designer средства выполнения FPGA-проекта позволяют:

- составить файлы исходных данных для проектирования внутренней логики ПЛИС в виде привычной радиоинженеру принципиальной схемы или описать логику на языке VHDL;

- ❑ промоделировать работу логики ПЛИС с помощью встроенного логического анализатора;
- ❑ указать тип конкретной микросхемы одного из известных мировых производителей и ввести специальные файлы привязок (Constraints), устанавливающие связь логической схемы проекта и выбранной ПЛИС;
- ❑ откомпилировать проект как это принято при выполнении проектов остальных типов;
- ❑ конвертировать данные проекта в обменный формат EDIF для передачи в поставляемую фирмой-производителем ПЛИС специализированную среду проектирования внутренней коммутации логических связей на кристалле ПЛИС выбранного семейства (Vendor Tools);
- ❑ запрограммировать в этой специализированной среде проектирования коммутацию внутренних логических связей и вывод входных и выходных логических сигналов на predetermined для этого контакты выбранной микросхемы;
- ❑ установить выбранную микросхему ПЛИС и отладить логику на подключенной к параллельному порту компьютера отладочной панели NanoBoard (поставляется по отдельному заказу);
- ❑ сгенерировать файлы управляющей информации для коммутации в выбранной ПЛИС.

После выполнения рассмотренных операций запрограммированная ПЛИС возвращается в проект верхнего уровня иерархии, включается как обычный компонент в документ схемного редактора Altium Designer, и оставшиеся этапы разработки выполняются в обычном РСВ-проекте.

Отметим, что в иерархию РСВ-проекта может войти несколько FPGA-проектов, если проектируемый функциональный узел выполняется на нескольких ПЛИС. В свою очередь, в иерархию FPGA-проекта могут входить один или несколько "встроенных" проектов программного обеспечения (Embedded Project), если на ПЛИС реализуется процессорное ядро микроконтроллера или микро-ЭВМ, программируемой под задачу пользователя.

Библиотечное обеспечение FPGA-проектов включает библиотеки базовых логических узлов, из которых строится схема проектируемого устройства, и ряд логических прототипов широко распространенных 16- и 32-разрядных микроконтроллеров.

7.1. Формирование нового проекта ПЛИС

Проект функционального узла, выполняемый на ПЛИС в среде Altium Designer, называется FPGA Project. Файлу оболочки проекта назначается имя с предопределенным расширением PrjFpg.

Последовательность образования нового проекта ПЛИС¹:

1. В правом нижнем углу главного окна программы указать команду-вкладку **System** и открыть в контекстном меню плавающую панель **Files**.
2. В поле **New** панели **Files** активизировать команду **Blank Project (FPGA)**.
3. Автоматически открывается плавающая панель **Projects** (рис. 7.2), содержащая предопределенное по умолчанию имя нового проекта `FPGA_Project1.PrjFpg`.

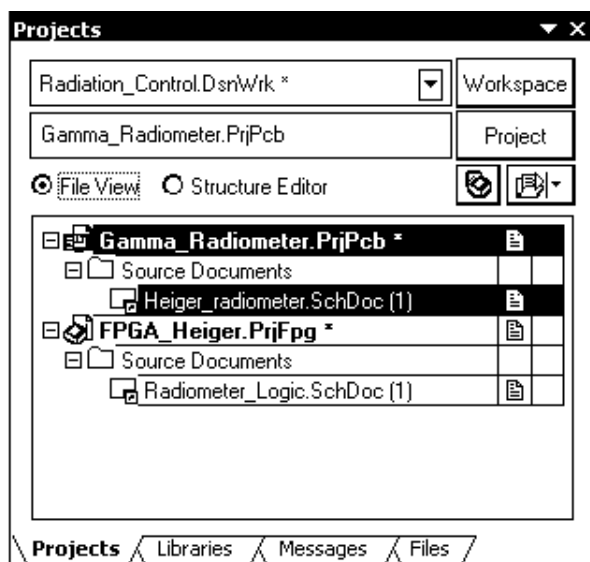


Рис. 7.2. FPGA-проект в панели Projects

Аналогичный результат достигается, если в главном меню программы выполнить цепочку команд **File | New | Project | FPGA Project**.

¹ Документ Getting Started with FPGA Design.pdf. Tutorial TU0116 (v1.5), November 28, 2005.

4. Щелчком правой кнопки мыши на имени проекта открыть контекстное меню, указать в нем команду **Save Project As** и переименовать новый проект, присвоив ему имя, соответствующее функциональному назначению или другим отличительным особенностям разработки.

Далее мы попытаемся осветить основные моменты разработки FPGA-проекта на примере узла цифровой регистрации гамма-радиометра — прибора для измерения интенсивности радиоактивного гамма-излучения на основе газоразрядного счетчика Гейгера–Мюллера. Электрическая структурная схема такого регистратора приведена на рис. 7.3.

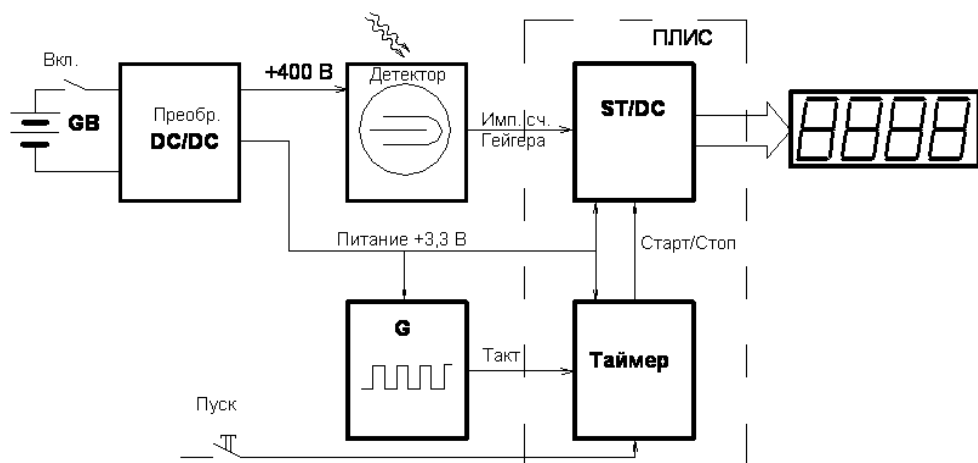


Рис. 7.3. Структурная схема гамма-радиометра

Прибор состоит из:

- источника первичного питания (гальванической или аккумуляторной батареи);
- преобразователя питания, подающего постоянное напряжение 400 В на детектор;
- кварцевого задающего генератора, обеспечивающего тактирование работы всей схемы;
- цифрового таймера, формирующего временные интервалы, в течение которых газоразрядный счетчик Гейгера–Мюллера или детектор другого типа регистрирует определенное число гамма-квантов, достаточное для получения отсчета интенсивности излучения с заданной точностью;

- четырехдекадного двоично-десятичного счетчика импульсов с дешифраторами двоичного кода в коды управления разрядами семисегментного жидкокристаллического индикатора в каждой декаде;
- в зависимости от выбора конкретного семейства и микросхемы ПЛИС для реализации проекта схема может включать конфигурационное ПЗУ, задающее программу коммутации внутренних логических связей ПЛИС, либо эта программа будет вводиться в ПЛИС с внешнего программатора перед установкой микросхемы на печатную плату или с компьютера непосредственно в готовый прибор.

Преобразователь высоковольтного источника вторичного питания, кварцевый задающий генератор, вырабатывающий тактовые импульсы с частотой 32 768 Гц, и конфигурационное ПЗУ (при его наличии) выполняются на отдельных микросхемах и дискретных компонентах и образуют внешнее окружение ПЛИС. Проектирование функциональных узлов на печатных платах с использованием таких компонентов рассматривалось в предыдущих главах книги. В настоящем разделе мы сосредоточим наши усилия непосредственно на проектировании логической структуры цифровой части радиометра и реализации ее в ПЛИС.

Времязадающую часть (таймер) радиометра построим так, чтобы при естественной (фоновой) мощности дозы (интенсивности) гамма-излучения в 12 мкР/час счетчик Гейгера мог уловить 12 квантов излучения в течение нескольких секунд. С точки зрения инженерной психологии и радиационной безопасности время ожидания отсчета мощности дозы может составлять 15...30 с. Более длительное ожидание отсчета утомительно для человека, выполняющего измерение, а то и опасно для здоровья при высоких уровнях интенсивности излучения. В зависимости от марки счетчика Гейгера этот интервал может быть различным. В нашем случае при использовании в приборе четырех счетчиков типа СБМ-21 интервал, необходимый для регистрации 12 квантов, составляет 18 с. При таком условии таймер можно построить на двух двоичных счетчиках, первый из которых имеет коэффициент пересчета (деления частоты) 32 768, а второй — 18. Для этой цели пригодны счетчики с предварительным занесением кода по пусковому импульсу, после чего тактовые импульсы задающего генератора доводят счетчики до переполнения в продолжение 18 с.

Зададимся диапазоном интенсивностей, подлежащих контролю, в пределах 10...9999 мкР/час. При таком диапазоне счетчик-регистратор отсчета составляется из четырех двоично-десятичных декад с дешифраторами двоичного

кода в коды управления четырьмя семисегментными жидкокристаллическими цифровыми индикаторами.

Прежде чем перейти непосредственно к разработке этой части проекта необходимо рассмотреть вопросы обеспечения проектных процессов библиотеками базовых логических элементов и узлов, образующих логику ПЛИС.

7.2. Библиотечное обеспечение проектов ПЛИС

С программным пакетом Altium Designer поставляется обширная библиотека базовых логических элементов FPGA Generic Library.IntLib¹, позволяющая вводить исходные данные для проектирования логической структуры функционального узла на ПЛИС в привычной инженеру-разработчику графической форме.

Библиотека составлена по принципу независимости от конкретных семейств ПЛИС, поставляемых на рынок разными производителями. Это позволяет реализовать FPGA-проект Altium Designer, на выбор, в микросхемах таких фирм-производителей, как Altera, Atmel, Xilinx, Lattice, Sharp. Приведем основные сведения о составе, принципах обозначения и правилах использования элементов библиотеки логических прообразов функциональных блоков ПЛИС в сквозном проектировании узлов РЭС на микросхемах программируемой логики.

Имена логических прототипов кодируют лаконичными аббревиатурами, для понимания которых следует рассмотреть принятые в библиотеке Altium Designer соглашения об именах и правила обозначения базовых логических элементов проекта ПЛИС.

Имя библиотечного элемента — функционального прототипа логической ячейки ПЛИС — образуется из обозначения функционального класса, типа (Type), разрядности (Bit-Size), функциональных особенностей (Function), дополнительных признаков (State — направление счета или сдвига в регистрах, предел счета для программируемых делителей частоты и др.) и признака однолинейного (Bus — в виде линии групповой связи) или многолинейного (Single — для каждой линии свой проводник) изображения входных и вы-

¹ Документ FPGA Generic Library Guide.pdf. Core Reference CR0118 (v1.0) January 29, 2004.

ходных линий логического элемента. Приведенный формат имени включает обязательные и необязательные поля, в зависимости от функционального класса и других особенностей конкретного библиотечного элемента.

Все логические прототипы сгруппированы в 18 функциональных классов, в целом совпадающих с классами широко распространенных микросхем малой и средней степени интеграции, — логические примитивы (LogicalPrimitives — схемы И, ИЛИ, НЕ и др.), триггеры (Flip-Flop), делители частоты (Clock Divider), в том числе с программируемым коэффициентом деления, счетчики (Counter), элементы арифметических устройств — сумматоры, перемножители (Arithmetic Function) и т. д. Кроме этих функциональных классов, характерных для интегральных микросхем, имеются еще три вида логических прототипов, отвечающих специфике ПЛИС:

- ❑ Bus Joiner — шинные соединители, облегчающие соединение линий групповой связи при формировании логической схемы проекта;
- ❑ JTAG — логический прототип элемента JTAG-интерфейса для ввода программы коммутации внутренней структуры ПЛИС;
- ❑ Wired Function — прототип набора резисторов, предназначенных для "оттяжки" логических цепей к "земле" (PULLDOWN) или к напряжению питания (PULLUP).

Приведем примеры обозначения логических прототипов¹, которые в дальнейшем будут использованы в логической схеме ПЛИС, на примере которой мы попытаемся осветить основные моменты разработки FPGA-проекта. Обозначения даются в виде металингвистических формул, определяющих формат имени. В обозначениях конкретных логических узлов в поля металингвистической формулы вносятся алфавитно-цифровые условные обозначения соответствующих функциональных признаков, в результате чего имена логических компонентов представляют цепочки алфавитно-цифровых символов.

Поскольку логические прототипы функциональных ячеек ПЛИС не имеют конструктивного воплощения и не подлежат приобретению как комплектующие изделия, будем считать, что при выполнении FPGA-проекта логическая структура ПЛИС является внутренним документом проекта и может не выполняться в виде конструкторского документа по ЕСКД. Это избавляет от необходимости формировать УГО логических прототипов по ЕСКД и позволяет пользоваться их изображениями, поставляемыми в библиотеке Altium Designer.

¹ Документ FPGA Generic Library Guide.pdf. Core Reference CR0118 (v1.0) January 29, 2004.

□ Триггерный двоичный счетчик:

C<Type><Bit-Size><Function|{Function}>[Direrction][Version],

где

C — функциональный класс счетчиков (Counter);

Type — тип счетчика:

B — каскадируемый синхронный двоичный счетчик;

D — каскадируемый синхронный двоично-десятичный счетчик;

J — кольцевой счетчик (счетчик Джонсона);

R — асинхронный двоичный счетчик с переносом по отрицательно-му фронту;

Bit-Size — разрядность:

2, 4, 8 или 16 — для типов B и R;

4 — для типа D;

2, 4, 5, 8, 16 — для типа J;

Function — функция:

C — асинхронный сброс (Clear);

R — синхронный сброс (Reset);

P — асинхронная предустановка (Asynchronous Preset);

L — синхронная загрузка кода (Synchronous Load);

E — разрешение счета (Clock Enable).

Direction — направление счета (сложение/вычитание):

D — реверсивный счетчик.

Version — версия выполнения графического изображения линий данных:

S — многолинейное изображение линий входных и выходных данных: на каждый логический вывод своя линия связи (Single pin version);

B — однолинейное, "шинное" изображение линий входных и выходных данных (Bus pin version).

Таким образом, например, 4-разрядный двоично-десятичный счетчик с синхронным сбросом и разрешением счета, с многолинейным изображением линий входных и выходных данных имеет в библиотеке обозначение CD4RES.

□ Делитель частоты тактовых импульсов (Clock divider).

CDIV<Num>[Prog Bit-Size][Duty Cycle],

где

CDIV — функциональный класс делителей;

Num — коэффициент деления частоты:

2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 16, 20, 24, 32, 64, 128, 256;

Prog Bit-Size — разрядность кода коэффициента деления для программируемых делителей:

N_8 — 8-разрядный параллельный код;

N_16 — 16-разрядный код;

N_32 — 32-разрядный код;

Duty Cycle — доля времени активного положительного сигнала на выходе, в процентах к длительности цикла (для непрограммируемых делителей).

□ Дешифраторы двоичного кода (Decoder).

D<Type>[Function][Version],

где

D — функциональный класс дешифраторов;

Type — тип преобразования:

4_10 — 4-разрядный двоичный код на 10 выходных линий унитарного кода;

7SEG — 4-разрядный двоичный код в коды управления семисегментными цифровыми индикаторами (высокий активный уровень);

7SEGN — 4-разрядный двоичный код в коды управления семисегментными цифровыми индикаторами (низкий активный уровень);

n_m — *n*-разрядный двоичный код в *m* выходных линий унитарного кода:

доступны три варианта: 2_4, 3_8 и 4_16;

Function:

E — разрешение декодирования (только для вариантов 2_4, 3_8, 4_16);

Version — вариант выполнения входных и выходных линий:

S — многолинейное изображение линий данных (Single pin version);

B — однолинейное, "шинное" изображение линий данных (Bus pin version).

Таким образом, дешифратор двоично-десятичного кода в коды управления семисегментным цифровым индикатором, с высоким активным уровнем выходных сигналов, который мы в дальнейшем будем использовать в примере FPGA-проекта, обозначается как D7SEG.

□ Базовые логические ячейки (Logic Primitive).

<Type><Bit-Size>[Function][Version],

где

Type — тип логической ячейки:

AND — ячейка И;

NAND — ячейка И-НЕ;

OR — ячейка ИЛИ;

NOR — ячейка ИЛИ-НЕ;

XOR — ячейка "исключающее ИЛИ";

XNOR — ячейка "исключающее ИЛИ-НЕ";

INV — ячейка НЕ (инвертор);

TCZO — логическое дополнение до нуля/единицы;

SOP — сумма логических произведений;

Bit-Size — число входных логических линий (разрядность слова):

2, 3, 4, 5, 6, 7, 8, 9, 12, 16, 32 — для ячеек AND, NAND, OR, NOR, XOR и XNOR;

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 16, 32 — для инверторов;

m_n — обозначение числа *m* *n*-входовых ячеек AND (И), для которых определяется сумма произведений: допустимые значения 2_2, 2_3, 2_4 и 4_2;

Function — функциональные особенности (только для ячеек AND, NAND, OR, NOR, XOR и XNOR):

Nm — число логических входов активного низкого уровня (применимо только к ячейкам разрядности 2, 3, 4 и 5, причем число *m* меньше или равно разрядности ячейки);

D — наличие прямого и инверсного выходов (только у ячеек И и ИЛИ с разрядностью 2, 3 и 4);

Version — версия выполнения логических линий:

S (Single pin version) — многолинейное исполнение (для каждого сигнала своя линия);

B (Bus version) — исполнение в виде линии групповой связи (шины).

Например, трехходовая логическая ячейка И с одним входом активного низкого уровня и многолинейным исполнением входных линий обозначается именем AND3N1S.

□ Порт подключения шестнадцатеричного числа (Numeric Connector).

NUM<Hex Value>,

где Hex Value — постоянный четырехразрядный двоичный (он же шестнадцатеричный) код в пределах 0...F. Предназначается для ввода предустановок в двоичные счетчики и регистры.

□ Соединители и разветвители шин (Bus Joiner).

J<Bit><Port>[Bus-Num]_<Bit><Port>[Bus-Num][Pin-Type],

где

J — функциональный класс соединителей/разветвителей;

Bit — разрядность шины или одной ветви слева и справа:

2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 16, 32;

Port — способ изображения входных и выходных линий:

B — однолинейное (шинное) изображение (Bus);

S — многолинейное изображение (Single pin);

Bus-Num — число входящих и выходящих ветвей-шин:

(1), 2, 4, 8;

Pin-Type — тип входящих и выходящих связей:

X — для двунаправленных линий (INOUT). Если данное обозначение отсутствует, по умолчанию принято, что контакты соединителя имеют тип "входной" (IN) слева и "выходной" (OUT) справа.

Отметим, что допустимы не все возможные, а только 52 комбинации разрядностей и способов группирования входящих и выходящих ветвей.

Например, соединитель, принимающий слева 4 шины по 4 входных линии в каждой, объединяющий их и отдающий вправо 16-линейную шину, обозначается именем J4B4_16B.

Таблицы коммутации входных и выходных линий для каждого из вариантов шинных соединителей и разветвителей приведены в руководстве по работе с библиотекой FPGA Generic Library Guide.pdf.

К сожалению, ограниченный объем книги не позволяет привести более обширные сведения по обозначению элементов библиотеки логических прото-

типов: полный объем руководства FPGA Generic Library Guide.pdf. составляет 390 страниц.

Аналогично тому, как схемные символы библиотек компонентов, предназначенных для выполнения РСВ-проекта, связаны с конструктивным образом (топологическим посадочным местом) соответствующих компонентов, за графическим представлением каждого логического прототипа стоит внутреннее представление модели его логики в конструкциях языка EDIF. При формировании логической структуры ПЛИС система проектирования конвертирует графический схемный образ в формат внутреннего представления на языке EDIF, которое только и позволяет в конечном итоге запрограммировать внутреннюю коммутацию логических связей ПЛИС.

Библиотечное обеспечение не ограничивается библиотекой базовых логических прототипов (Generic components). Для реализации в проектах ПЛИС различных видов микроконтроллеров в поставке пакета имеется библиотека FPGA Processors.IntLib — логических прототипов процессорного ядра нескольких широко распространенных микропроцессоров, в том числе прототипы, совмещенные по системе команд с микропроцессорами семейства 165x фирмы Microchip, 8051 фирмы Intel, Z80 фирмы Zilog и еще нескольких типов.

Библиотека ...\\Altium Designer\\Library\\Fpga\\FPGA NanoBoard Port-Plugin.IntLib содержит схемные образы и EDIF-модели элементов обмена информацией и управления отладочной панелью NanoBoard.

Библиотека ...\\Altium Designer\\Library\\Fpga\\FPGA Peripherals.IntLib содержит логические прототипы контроллеров периферийных устройств различного назначения.

Проектирование сложных функциональных узлов, в том числе микроконтроллеров с развитой периферией, на основе этих прототипов требует разработки иерархических проектов, имеющих на верхнем уровне иерархии рассмотренный ранее РСВ-проект, на следующем уровне — один или несколько проектов ПЛИС (FPGA Project) и на нижнем уровне — один или несколько проектов встроенного программного обеспечения Embedded Project (файл оболочки проекта *.PrjEmb). В данной книге невозможно охватить все вопросы проектирования микроконтроллеров. Далее приведен пример проекта функционального узла с жесткой логикой, для реализации которого потребуются только базовые логические прототипы.

7.3. Формирование логической структуры ПЛИС

Рассмотрим вариант графического ввода логической схемы, подлежащей выполнению в ПЛИС.

7.3.1. Конфигурация рабочего пространства схемного редактора

1. Командой главного меню **File | New | Schematic** открыть новый документ схемного редактора Schdoc1.SchDoc. Имя нового файла отображается в плавающей панели **Projects** (см. рис. 7.2) в ветви дерева документов **Free Documents**.
2. Командой главного меню **File | Save As** сохранить новый документ в памяти компьютера под именем Radiometer_Logic.SchDoc.
3. Щелчком правой кнопки мыши в плавающей панели **Projects** на имени проекта FPGA_Heiger.PrjFpg открыть контекстное меню и командой **Add Existing to Project** включить схемный документ в структуру проекта. Аналогичного результата можно достичь "перетаскиванием" (Drag and Drop) имени схемного документа в дерево документов проекта панели **Projects**.
4. Щелчком правой кнопки мыши в поле графического окна схемного редактора вызвать контекстное меню и выполнить цепочку команд **Options | Document Options**. Откроется диалоговое окно **Document Options** с тремя вкладками — **Sheet Options**, **Parameters** и **Units**. Отконфигурировать рабочее пространство схемного редактора:
 - поскольку УГО библиотечных логических прототипов построены в дюймовой метрике, установить дюймовую систему единиц измерения — на вкладке **Units** выбрать опцию **Use Imperial Unit System**;
 - на вкладке **Sheet Options** установить:
 - ◇ размер листа, достаточный для размещения логической схемы ПЛИС (для нашего примера хватит стандартного формата A2);
 - ◇ шаг сетки захвата (**Snap**) 100 mil (миллидюймов); в метрической системе это 2,54 мм;

- ◇ шаг видимой сетки (**Visible**) также 100 mil;
- ◇ шаг "электрической" сетки (**Electrical Grid Range**) 40 mil (в метрической системе это 1,02 мм).

Несмотря на то, что логическая схема ПЛИС является промежуточным документом проекта, она должна найти свое место в структуре комплекта конструкторских документов РСВ-проекта верхнего уровня иерархии, возможно, как дополнительный документ. Поэтому целесообразно подключить к схеме стандартную форматку и составить реквизиты документа по ЕСКД, как это описано в *разд. 3.1*.

7.3.2. Вызов библиотечных компонентов

Порядок действий таков.

1. Щелчком левой кнопки мыши в поле **System** главного окна программы вызвать всплывающее меню и указать в нем команду **Libraries**. Откроется одноименная плавающая панель **Libraries** (рис. 7.4).
2. Кнопкой **Install** запустить процедуру поиска и включения библиотек в рабочую среду проекта. Найти и подключить библиотеку логических прототипов ...\\Altium Designer\\Library\\FPGA\\FPGA Generic.IntLib.

В полях плавающей панели **Libraries** отображаются имя библиотеки, список прототипов и графическое обозначение выбранного прототипа.

3. Щелчком на кнопке **Place<имя>** активизировать операцию переноса выбранного прототипа на поле схемы. Панель **Libraries** временно гасится программой, и вся площадь графического окна схемного редактора оказывается доступной для размещения УГО логического прототипа.
4. Клавишей табуляции <Tab> открыть диалоговое окно управления свойствами компонента **Component Properties** и назначить позиционное обозначение для выбранного компонента (**Designator**). Щелчком на кнопке **OK** закрыть диалог управления свойствами.
5. Переместить прототип курсором мыши и зафиксировать необходимое число его копий щелчками левой кнопки мыши. Программа автоматически инкрементирует позиционные обозначения.

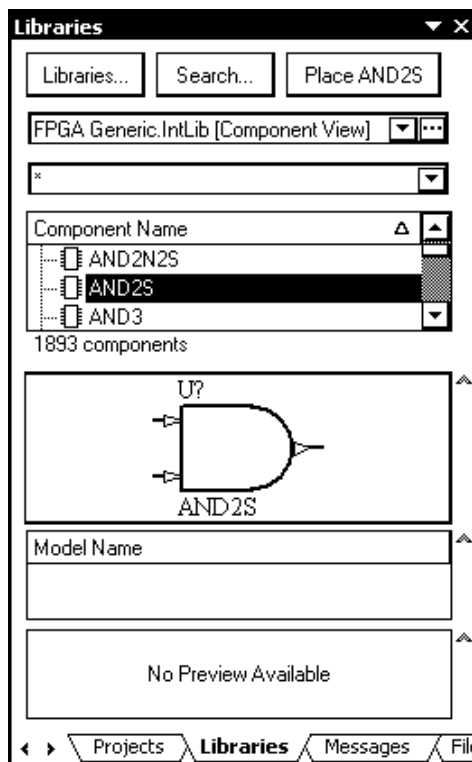


Рис. 7.4. Вызов библиотечного логического прототипа из панели **Libraries**

Действуя таким образом, вызвать из библиотеки и разместить на поле схемы в графическом окне программы элементы логической схемы, реализуемой в ПЛИС:

- ☐ цепочку D-триггеров U1, U4, U7, U8, U11, образующих синхронный формирователь импульсов управления логикой регистратора гамма-квантов;
- ☐ два двоичных счетчика U6 и U17 типов, соответственно, CB16CES и CL8CLEDB, образующих делитель частоты таймера;
- ☐ два 4-разрядных порта для занесения шестнадцатеричного кода предустановки $ED_{\text{HEX}} = 237_{\text{DEC}}$ во второй счетчик таймера;
- ☐ соединитель шин J2B4_8B для подключения линий, несущих код предустановки ко входам счетчика U17 таймера;

- ❑ четыре каскадируемых двоично-десятичных счетчика CD4CES с асинхронным сбросом и входом разрешения счета, образующих регистрирующую часть радиометра;
- ❑ одноразрядный инвертор U2 типа INV;
- ❑ четыре дешифратора типа D7SEGS для преобразования двоичного кода в коды управления сегментами цифрового индикатора;
- ❑ шестнадцативходовый логический вентиль AND16S, выполняющий функцию дешифратора кода 4000_{HEX} на выходах счетчика U6;
- ❑ двухвходовый логический вентиль AND2S в цепь управления регистрацией отсчета.

7.3.3. Линии электрической связи

Линии электрической связи схемы выполняются по тем же правилам с помощью тех же команд, что и в схеме PCB-проекта:

- ❑ **Place Wire** — команда размещения индивидуальных линий связи;
- ❑ **Place Bus** — команда размещения линий групповой связи.

Связность электрической цепи можно обеспечить двумя способами:

- ❑ непосредственным прочерчиванием линии электрической связи от контакта до контакта компонентов (логических прототипов);
- ❑ присвоением единой метки цепи (**Net Label**) разбросанным по полю чертежа схемы сегментам одной и той же линии связи.

Все это относится как к индивидуальным связям, так и к линиям групповой связи (шинам). Для линий групповой связи формат метки включает имя линии связи и обозначение разрядности — числа линий связи, объединенных в групповую, например, DATA[7..0] или DATA[0..7]. Индивидуальные линии, входящие и отходящие от групповой линии, обозначаются как DATA0, DATA1 и т. д.

В отличие от схемных документов PCB-проекта для слияния и разветвления линий групповой связи в FPGA-проекте можно использовать рассмотренные ранее шинные соединители и разветвители.

Шинные соединители при включении в схему не отличаются от обычных компонентов. На рис. 7.5 приведены примеры допустимого обозначения входных и выходных линий при использовании шинных соединителей/разветвителей.

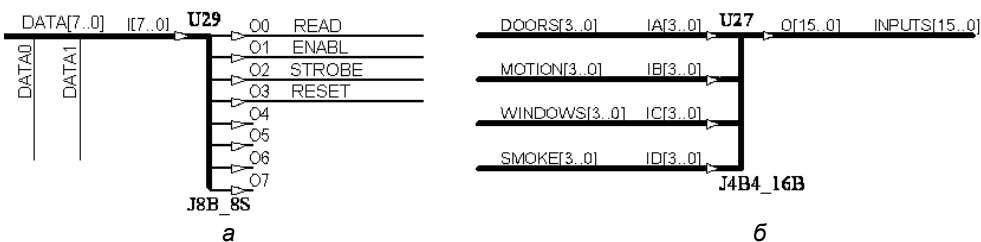


Рис. 7.5. Примеры соединителей и разветвителей линий групповой связи:
 а — разветвитель; б — соединитель четырех линий в одну

В схемных документах PCB-проекта признаки связности цепей, образующих линию групповой связи, назначают путем присоединения единой метки цепи каждой электрической связи схемы во всех точках входа и выхода ее из групповой линии. При использовании шинных соединителей имена входных и выходных линий, как отдельных, так и групповых, могут различаться. На рис. 7.5, а индивидуальные линии, образующие линию групповой связи с именем DATA[7..0], обозначены по общим правилам, принятым в схемных документах, метками DATA0, DATA1 и т. д. Входные контакты шинного разветвителя U29 промаркированы в "шинном" формате индексом I[7..0]. Выходные контакты разветвителя имеют индексы O0, O1...O7, а подключенные к ним индивидуальные линии связи могут обозначаться произвольными именами, отличающимися от имен связей исходной групповой линии.

На рис. 7.5, б четыре 4-разрядных линии групповой связи объединяются в одну 16-разрядную линию групповой связи. Входные контакты шинного соединителя U27 объединены в группы по четыре контакта и обозначены индексами IA[3..0], IB[3..0], IC[3..0], ID[3..0], а выходные контакты — индексом O[15..0]. Вторая буква в индексах входных линий определяет старшинство разрядов входных линий при объединении их в выходную линию групповой связи INPUTS[15..0]: разряды 3...0 входной линии IA передаются в старшие разряды 15...12 выходной линии; разряды 3...0 входной линии ID передаются в младшие разряды 3...0 выходной линии.

В результате проектирования получаем логическую схему устройства цифровой регистрации мощности дозы гамма-радиометра (рис. 7.6), состоящую из трех узлов: таймера, схем управления и регистрации.

Схема управления выполнена на триггерах U1, U4, U7, U8, U11. На триггерах U1, U4, U7, U8 построен синхронный формирователь последовательности импульсов Clear (сброса счетчиков U6, U17, U5, U9, U14, U16), Load (предустановки счетчика U17) и импульса Begin_18 (начала 18-секундного интервала отсчета) (рис. 7.7).

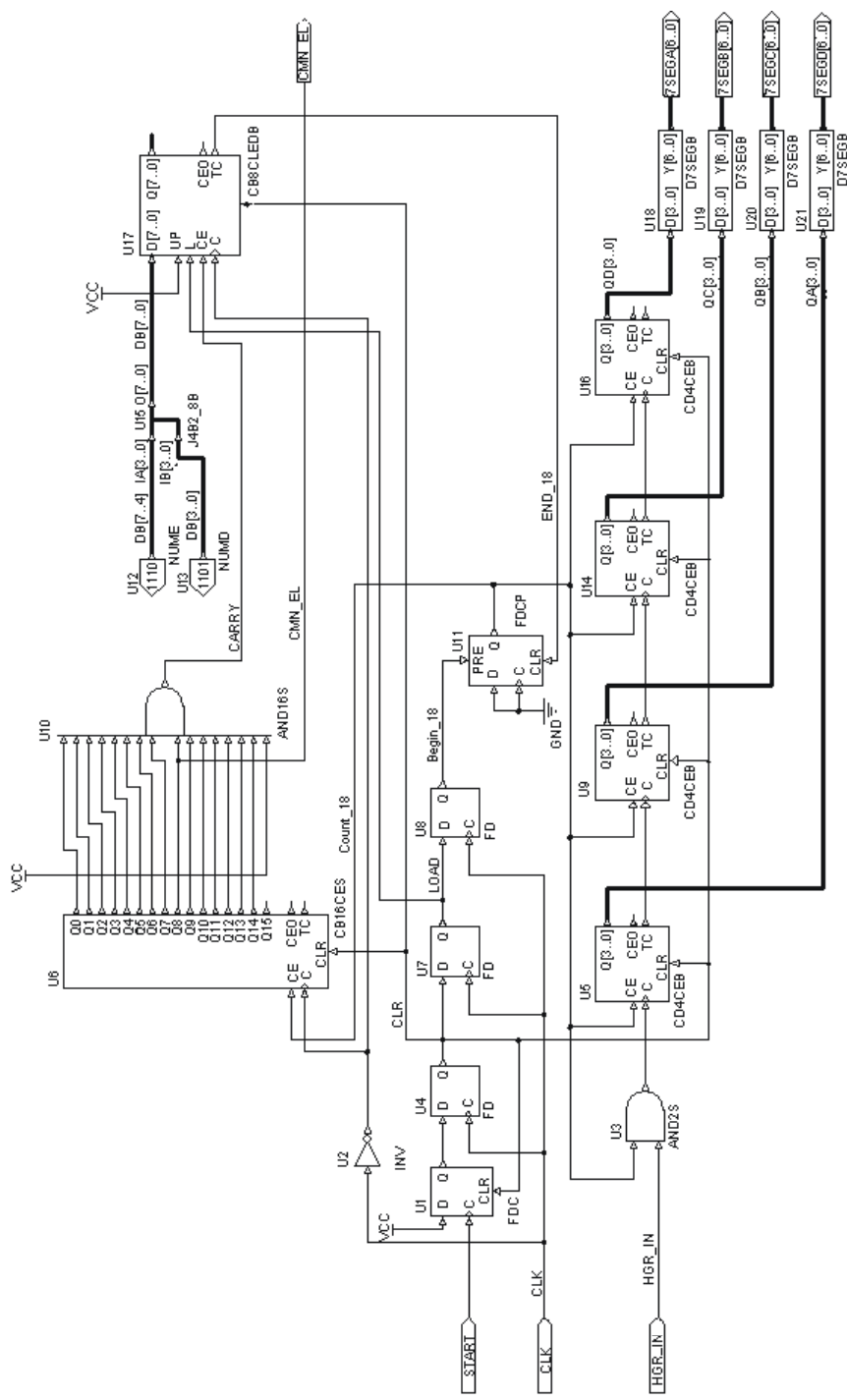


Рис. 7.6. Схема логического ядра радиометра, реализуемого в ПЛИС

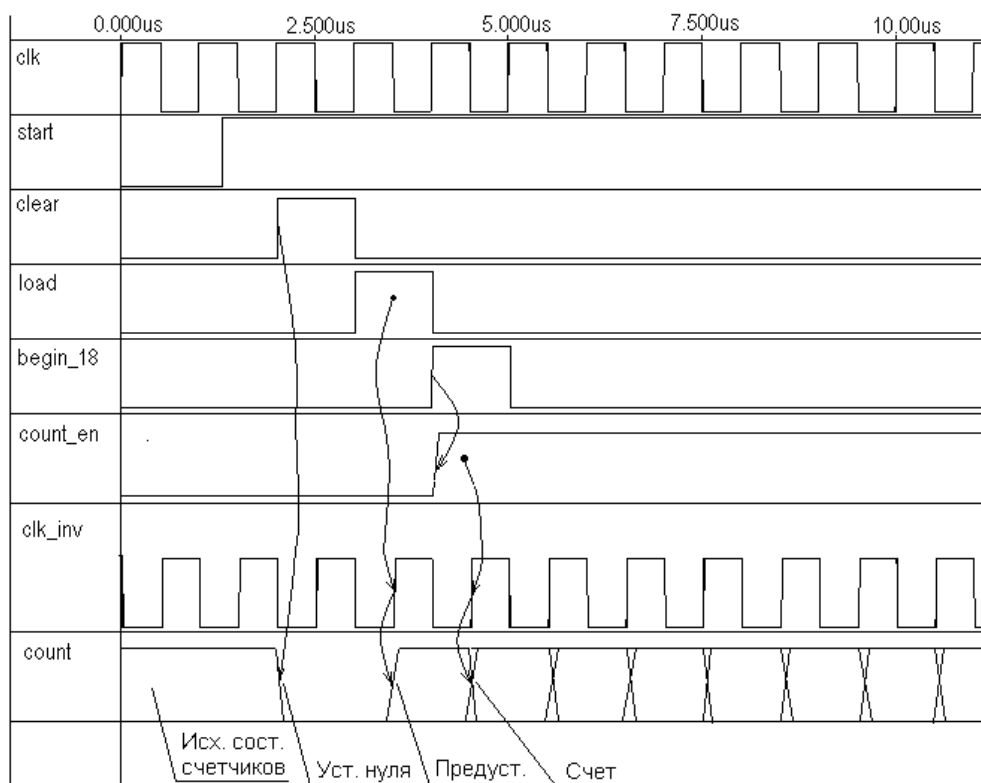


Рис. 7.7. Временные диаграммы напряжений в узлах логического ядра

Импульс *Begin_18* поступает на вход предустановки *PRE* триггера *U11* и устанавливает его в состояние "единица". Положительный потенциал с выхода *Q* триггера *U11* подается на входы разрешения счета счетчиков таймера *U6*, *U17* и счетчиков *U5*, *U9*, *U14*, *U16*, регистрирующих число импульсов от датчиков гамма-излучения (четырёх газоразрядных счетчиков Гейгера–Мюллера), поступающих на ПЛИС через порт *HGR_IN*.

Таймер, отсчитывающий время набора импульсов отсчета мощности дозы, состоит из двух двоичных счетчиков с асинхронным сбросом — 16-разрядного счетчика *U6* и 8-разрядного счетчика с занесением кодов предустановки *U17*.

Первый счетчик делит частоту импульсов тактового генератора, поступающих с выхода инвертора *U2*, с 32 768 до 1 Гц. Полная емкость 16-разрядного

счетчика составляет 65 536. Счетчик считает тактовые импульсы до заполнения единицами 15 младших разрядов — до достижения кодом в разрядах счетчика значения $FFFF_{\text{HEX}} = 32767_{\text{DEC}}$. На это уходит время 1 с. Код $FFFF_{\text{HEX}}$ дешифруется логической ячейкой U10 AND16S.

Выходной логический сигнал высокого уровня с вентиля U10 поступает на вход разрешения счета CE второго счетчика таймера, U17 и удерживается там в течение последнего, 32768-го тактового интервала первой секунды отсчета, после чего по положительному перепаду напряжения тактового сигнала на входе С счетчика U17 в этот счетчик заносится первая единица. Для достижения коэффициента пересчета 18 в счетчик U17 в момент запуска измерения с двух шестнадцатеричных портов U12, U13 заносится код предустановки $ED_{\text{HEX}} = 237_{\text{DEC}} = 256 - 18$.

В результате счетчик U17 отсчитывает до переполнения 18 интервалов по 1 с, после чего импульсом переноса со своего выхода ТС сбрасывает в ноль триггер управления U11, и на этом время регистрации отсчета мощности дозы заканчивается.

Число гамма-квантов излучения определяется путем подсчета импульсов датчика излучения, поступающих с входного порта HGR_IN через вентиль U3 в течение 18 с.

Технические характеристики применяемых в разрабатываемом регистраторе счетчиков Гейгера–Мюллера типа СБМ-21 таковы, что при фоновом значении интенсивности гамма-излучения 12 мкР/час четыре счетчика, работающие на общую нагрузку, захватывают 12 квантов излучения в течение 18 с. Таким образом, при использовании четырехдекадного счетчика-регистратора гамма-квантов пределы измерения мощности дозы составляют 10...9999 мкР/час. Меньшие значения интенсивности радиоактивного излучения в природе не встречаются, а интенсивности выше 10 мР/час маловероятны в бытовых условиях, для которых предназначен наш регистратор.

7.4. Выбор путей реализации проекта

Следующий этап выполнения проекта после составления логической схемы ПЛИС — выбор возможностей его реализации. В простейшем случае для этого можно задать тип микросхемы ПЛИС, указать разводку логических сигналов проекта на выводы этой микросхемы, провести трассировку внутренней структуры кристалла ПЛИС и вернуть трассированную ПЛИС в кор-

новой проект печатной платы. В Altium Designer допускаются более широкие возможности реализации одной и той же логики ПЛИС, например реализации ее в разных типах микросхем ПЛИС, в варианте, предназначенном для отладки на входящей в комплект с САПР отладочной панели Nano Board или пользовательском отладочном стенде и, наконец, в окончательном исполнении на печатной плате функционального узла, предназначенного для промышленного выпуска. Это свойство FPGA-проекта называется переносимостью проектов (Design Portability)¹.

Переносимость проекта ПЛИС достигается путем формирования так называемых файлов привязки (Constraint Files), в которых фиксируются варианты выбора типа микросхем ПЛИС, составляется описание внешних портов схемы, их электрических свойств подключения к внешним выводам выбранных микросхем, в соответствии с числом возможных реализаций проекта. Это текстовые файлы, которые включаются в логическую структуру проекта и отображаются в ветви **Settings** дерева проекта в плавающей панели **Projects**.

7.4.1. Создание нового файла привязок

Для создания нового файла привязок нужно выполнить команду главного меню **Project | Add New to Project | Constraint File**. Оболочка нового файла привязок отображается в главном окне программы.

Строки файла привязок (листинг 7.1), называемые группами, начинаются с идентификатора группы `Record`. Идентификатору группы можно присвоить одно из двух значений: `FileHeader` или `Constraint`. Идентификатор `Id` определяет тип и версию файла привязок.

Листинг 7.1

```
;  
.....  
;Constraints File  
  
;   Device      :  
;   Board       :  
;   Project     :
```

¹ Документ AR0124 Design Portability Configurations and Constraints.pdf Article AR0124 (v.1.1) July 06, 2005.

```
; .....  
;   Created   11.11.2008  
;  
; .....  
Record=FileHeader | Id=DXP Constraints v1.0  
;
```

Далее следует стадия редактирования файла привязок — наполнение его оболочки информацией о типе выбранной микросхемы ПЛИС, о направлении сигналов с внешних портов логической схемы, выполняемой на выбранной ПЛИС, на выводы микросхемы. Эта информация используется в ассоциированных с Atium Designer специализированных САПР фирм — производителей микросхем ПЛИС (Vendor Tools) для синтеза проекта (преобразования исходной логической схемы в комбинацию элементарных логических ячеек и логических блоков) и программирования ПЛИС (коммутации внутренних логических связей).

7.4.2. Выбор микросхемы ПЛИС

Последовательность действий такова.

1. Активизировать команду главного меню программы **Design | Add/Modify Constraint** и указать подкоманду **Part** (рис. 7.8). Откроется диалоговое окно выбора микросхемы **Choose Physical Device** (рис. 7.9).

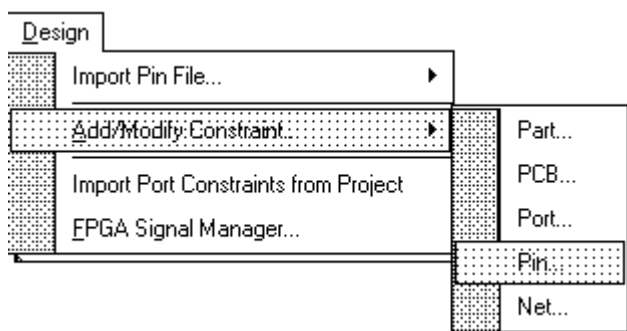


Рис. 7.8. Команды формирования групп файла привязок

- В поле **Vendors** отображается список фирм — производителей микросхем программируемой логики, использование которых в проектах ПЛИС поддерживается в Altium Designer. В наших целях выбираем микросхемы фирмы Altera.
- В поле **FPGA Families** выводится список семейств микросхем ПЛИС выбранного производителя. Выбор семейства ПЛИС определяется такими факторами, как объем и сложность логической схемы проекта, быстродействие, электрическая совместимость выводов ПЛИС с технологическими стандартами микросхем внешней "обвязки" на печатной плате РСВ-проекта и др. Выбираем из этих соображений для рассматриваемого примера микросхемы семейства MAX 7000S.

Choose Physical Device

Vendors

Actel

ALTERA

AMCC

Lattice

Sharp

XILINX

FPGA Families

Cyclone

Cyclone2

MAX2

MAX3000A

MAX7000AE

MAX7000B

MAX7000S

NIOS2

Stratix

Stratix2

StratixGX

Temperature Grades

Commercial Grade

Industrial Grade

Speed Grades

10 ns pin-to-pin delay

15 ns pin-to-pin delay

5.0 ns pin-to-pin delay

6.0 ns pin-to-pin delay

7.5 ns pin-to-pin delay

ALTERA

MAX

ON BOARD

MAX7000S

MAX7000S

5V MAX 7000S In-System Programmable CPLD

Base Device	EPM70325	EPM70645	EPM71285	EPM71605	EPM71925	EPM72565
Macrocells	32	64	128	160	192	256
I/O's	32	64	96	100	120	160

J-Lead Chip Carrier (L)

PLCC44	16.5 x 16.5 mm	32	32	-	-	-	-
PLCC84	29.3 x 29.3 mm	-	64	64	60	-	-

Thin Quad Flat Pack (T)

TQFP44	10 x 10 mm	32	32	-	-	-	-
TQFP100	14 x 14 mm	-	64	80	80	-	-

Quad Flat Pack - Option 1 (Q)

PQFP100	20 x 14 mm	-	-	80	-	-	-
PQFP160	28 x 28 mm	-	-	96	100	120	-
PQFP208	28 x 28 mm	-	-	-	-	-	160

Power Quad Flat Pack (R)

RQFP208	28 x 28 mm	-	-	-	-	-	160
---------	------------	---	---	---	---	---	-----

17:1

BANK A

IO1

IO2

IO3

IO4

IO5

IO6

IO7

IO8

IO9

IO10

IO11

IO12

IO13

IO14

IO15

IO16

IO17

IO18

IO19

IO20

IO21

IO22

IO23

IO24

IO25

IO26

IO27

IO28

IO29

IO30

IO31

IO32

IO33

IO34

IO35

IO36

IO37

IO38

IO39

IO40

IO41

IO42

IO43

IO44

IO45

IO46

IO47

IO48

IO49

IO50

IO51

IO52

IO53

IO54

IO55

IO56

IO57

IO58

IO59

IO60

IO61

IO62

IO63

IO64

IO65

IO66

IO67

IO68

IO69

IO70

IO71

IO72

IO73

IO74

IO75

IO76

IO77

IO78

IO79

IO80

IO81

IO82

IO83

IO84

IO85

IO86

IO87

IO88

IO89

IO90

IO91

IO92

IO93

IO94

IO95

IO96

IO97

IO98

IO99

IO100

IO101

IO102

IO103

IO104

IO105

IO106

IO107

IO108

IO109

IO110

IO111

IO112

IO113

IO114

IO115

IO116

IO117

IO118

IO119

IO120

IO121

IO122

IO123

IO124

IO125

IO126

IO127

IO128

IO129

IO130

IO131

IO132

IO133

IO134

IO135

IO136

IO137

IO138

IO139

IO140

IO141

IO142

IO143

IO144

IO145

IO146

IO147

IO148

IO149

IO150

IO151

IO152

IO153

IO154

IO155

IO156

IO157

IO158

IO159

IO160

IO161

IO162

IO163

IO164

IO165

IO166

IO167

IO168

IO169

IO170

IO171

IO172

IO173

IO174

IO175

IO176

IO177

IO178

IO179

IO180

IO181

IO182

IO183

IO184

IO185

IO186

IO187

IO188

IO189

IO190

IO191

IO192

IO193

IO194

IO195

IO196

IO197

IO198

IO199

IO200

IO201

IO202

IO203

IO204

IO205

IO206

IO207

IO208

IO209

IO210

IO211

IO212

IO213

IO214

IO215

IO216

IO217

IO218

IO219

IO220

IO221

IO222

IO223

IO224

IO225

IO226

IO227

IO228

IO229

IO230

IO231

IO232

IO233

IO234

IO235

IO236

IO237

IO238

IO239

IO240

IO241

IO242

IO243

IO244

IO245

IO246

IO247

IO248

IO249

IO250

IO251

IO252

IO253

IO254

IO255

IO256

IO257

IO258

IO259

IO260

IO261

IO262

IO263

IO264

IO265

IO266

IO267

IO268

IO269

IO270

IO271

IO272

IO273

IO274

IO275

IO276

IO277

IO278

IO279

IO280

IO281

IO282

IO283

IO284

IO285

IO286

IO287

IO288

IO289

IO290

IO291

IO292

IO293

IO294

IO295

IO296

IO297

IO298

IO299

IO300

IO301

IO302

IO303

IO304

IO305

IO306

IO307

IO308

IO309

IO310

IO311

IO312

IO313

IO314

IO315

IO316

IO317

IO318

IO319

IO320

IO321

IO322

IO323

IO324

IO325

IO326

IO327

IO328

IO329

IO330

IO331

IO332

IO333

IO334

IO335

IO336

IO337

IO338

IO339

IO340

IO341

IO342

IO343

IO344

IO345

IO346

IO347

IO348

IO349

IO350

IO351

IO352

IO353

IO354

IO355

IO356

IO357

IO358

IO359

IO360

IO361

IO362

IO363

IO364

IO365

IO366

IO367

IO368

IO369

IO370

IO371

IO372

IO373

IO374

IO375

IO376

IO377

IO378

IO379

IO380

IO381

IO382

IO383

IO384

IO385

IO386

IO387

IO388

IO389

IO390

IO391

IO392

IO393

IO394

IO395

IO396

IO397

IO398

IO399

IO400

IO401

IO402

IO403

IO404

IO405

IO406

IO407

IO408

IO409

IO410

IO411

IO412

IO413

IO414

IO415

IO416

IO417

IO418

IO419

IO420

IO421

IO422

IO423

IO424

IO425

IO426

IO427

IO428

IO429

IO430

IO431

IO432

IO433

IO434

IO435

IO436

IO437

IO438

IO439

IO440

IO441

IO442

IO443

IO444

IO445

IO446

IO447

IO448

IO449

IO450

IO451

IO452

IO453

IO454

IO455

IO456

IO457

IO458

IO459

IO460

IO461

IO462

IO463

IO464

IO465

IO466

IO467

IO468

IO469

IO470

IO471

IO472

IO473

IO474

IO475

IO476

IO477

IO478

IO479

IO480

IO481

IO482

IO483

IO484

IO485

IO486

IO487

IO488

IO489

IO490

IO491

IO492

IO493

IO494

IO495

IO496

IO497

IO498

IO499

IO500

IO501

IO502

IO503

IO504

IO505

IO506

IO507

IO508

IO509

IO510

IO511

IO512

IO513

IO514

IO515

IO516

IO517

IO518

IO519

IO520

IO521

IO522

IO523

IO524

IO525

IO526

IO527

IO528

IO529

IO530

IO531

IO532

IO533

IO534

IO535

IO536

IO537

IO538

IO539

IO540

IO541

IO542

IO543

IO544

IO545

IO546

IO547

IO548

IO549

IO550

IO551

IO552

IO553

IO554

IO555

IO556

IO557

IO558

IO559

IO560

IO561

IO562

IO563

IO564

IO565

IO566

IO567

IO568

IO569

IO570

IO571

IO572

IO573

IO574

IO575

IO576

IO577

IO578

IO579

IO580

IO581

IO582

IO583

IO584

IO585

IO586

IO587

IO588

IO589

IO590

IO591

IO592

IO593

IO594

IO595

IO596

IO597

IO598

IO599

IO600

IO601

IO602

IO603

IO604

IO605

IO606

IO607

IO608

IO609

IO610

IO611

IO612

IO613

IO614

IO615

IO616

IO617

IO618

IO619

IO620

IO621

IO622

IO623

IO624

IO625

IO626

IO627

IO628

IO629

IO630

IO631

IO632

IO633

IO634

IO635

IO636

IO637

IO638

IO639

IO640

IO641

IO642

IO643

IO644

IO645

IO646

IO647

IO648

IO649

IO650

IO651

IO652

IO653

IO654

IO655

IO656

IO657

IO658

IO659

IO660

IO661

IO662

IO663

IO664

IO665

IO666

IO667

IO668

IO669

IO670

IO671

IO672

IO673

IO674

IO675

IO676

IO677

IO678

IO679

IO680

IO681

IO682

IO683

IO684

IO685

IO686

IO687

IO688

IO689

IO690

IO691

IO692

IO693

IO694

IO695

IO696

IO697

IO698

IO699

IO700

IO701

IO702

IO703

IO704

IO705

IO706

IO707

IO708

IO709

IO710

IO711

IO712

IO713

IO714

IO715

IO716

IO717

IO718

IO719

IO720

IO721

IO722

IO723

IO724

IO725

IO726

IO727

IO728

IO729

IO730

IO731

IO732

IO733

IO734

IO735

IO736

IO737

IO738

IO739

IO740

IO741

IO742

IO743

IO744

IO745

IO746

IO747

IO748

IO749

IO750

IO751

IO752

IO753

IO754

IO755

IO756

IO757

IO758

IO759

IO760

IO761

IO762

IO763

IO764

IO765

IO766

IO767

IO768

IO769

IO770

IO771

IO772

IO773

IO774

IO775

IO776

IO777

IO778

IO779

IO780

IO781

IO782

IO783

IO784

IO785

IO786

IO787

IO788

IO789

IO790

IO791

IO792

IO793

IO794

IO795

IO796

IO797

IO798

IO799

IO800

IO801

IO802

IO803

IO804

IO805

IO806

IO807

IO808

IO809

IO810

IO811

IO812

IO813

IO814

IO815

IO816

IO817

IO818

IO819

IO820

IO821

IO822

IO823

IO824

IO825

IO826

IO827

IO828

IO829

IO830

IO831

IO832

IO833

IO834

IO835

IO836

IO837

IO838

IO839

IO840

IO841

IO842

IO843

IO844

IO845

IO846

IO847

IO848

IO849

IO850

IO851

IO852

IO853

IO854

IO855

IO856

IO857

IO858

IO859

IO860

IO861

IO862

IO863

IO864

IO865

IO866

IO867

IO868

IO869

IO870

IO871

IO872

IO873

IO874

IO875

IO876

IO877

IO878

IO879

IO880

IO881

IO882

IO883

IO884

IO885

IO886

IO887

IO888

IO889

IO890

IO891

IO892

IO893

IO894

IO895

IO896

IO897

IO898

IO899

IO900

IO901

IO902

IO903

IO904

IO905

IO906

IO907

IO908

IO909

IO910

IO911

IO912

IO913

IO914

IO915

IO916

IO917

IO918

IO919

IO920

IO921

IO922

IO923

IO924

IO925

IO926

IO927

IO928

IO929

IO930

IO931

IO932

IO933

IO934

IO935

IO936

IO937

IO938

IO939

IO940

IO941

IO942

IO943

IO944

IO945

IO946

IO947

IO948

IO949

IO950

IO951

IO952

IO953

IO954

IO955

IO956

IO957

IO958

IO959

IO960

IO961

IO962

IO963

IO964

IO965

IO966

IO967

IO968

IO969

IO970

IO971

IO972

IO973

IO974

IO975

IO976

IO977

IO978

IO979

IO980

IO981

IO982

IO983

IO984

IO985

IO986

IO987

IO988

IO989

IO990

IO991

IO992

IO993

IO994

IO995

IO996

IO997

IO998

IO999

IO1000

IO1001

IO1002

IO1003

IO1004

IO1005

IO1006

IO1007

IO1008

IO1009

IO1010

IO1011

IO1012

IO1013

IO1014

IO1015

IO1016

IO1017

IO1018

IO1019

IO1020

IO1021

IO1022

IO1023

IO1024

IO1025

IO1026

IO1027

IO1028

IO1029

IO1030

IO1031

IO1032

IO1033

IO1034

IO1035

IO1036

IO1037

IO1038

IO1039

IO1040

IO1041

IO1042

IO1043

IO1044

IO1045

IO1046

IO1047

IO1048

IO1049

IO1050

IO1051

IO1052

IO1053

IO1054

IO1055

IO1056

IO1057

IO1058

IO1059

IO1060

IO1061

IO1062

IO1063

IO1064

IO1065

IO1066

IO1067

IO1068

IO1069

IO1070

IO1071

IO1072

IO1073

IO1074

IO107

4. В центральном поле окна отображается список микросхем выбранного семейства, поддерживаемых программой, с их краткими конструктивно-технологическими и электрическими характеристиками:
 - **5V MAX 7000S In-System Programmable CPLD** — программируемые в аппаратуре пользователя ПЛИС (CPLD — Complex Programmable Logical Device) MAX 7000S с напряжением питания 5 В;
 - **Base Device** — перечень базовых микросхем, поддерживаемых программой;
 - **Macrocells** — число логических макроячеек на кристалле ПЛИС;
 - **I/O's** — число программируемых пользователем двунаправленных логических входов-выходов ПЛИС;
 - перечисление типов корпуса, в которых выполняются ПЛИС выбранного семейства.
5. В поле **Temperature Grades** возможен выбор варианта исполнения ПЛИС по температурному режиму — коммерческого (Commercial) или промышленного (Industrial) стандарта.
6. В поле **Speed Grades** выбрать характеристику быстродействия — указать время задержки распространения логического сигнала от контакта до контакта ПЛИС.
7. Указать курсором мыши ячейку таблицы в центральном поле диалогового окна, рис. 7.9, на пересечении столбца с именем ПЛИС и строки с обозначением типа корпуса. Ячейка выделяется цветовой подсветкой, а в двух нижележащих полях отображаются обозначение и краткий список технических характеристик выбранной ПЛИС, а также перечень технологических стандартов, электрически совместимых с выбранной ПЛИС.

В результате в файл привязок добавляется следующая группа:

Record=Constraint | TargetKind=Part | TargetId=EPM7064STC100-10

Поля данной группы обозначают тип объекта-мишени (TargetKind), для которого определяется привязка — компонент (Part) и идентификатор объекта-мишени (TargetId), обозначающий тип выбранной микросхемы ПЛИС.

7.4.3. Привязка портов

Последовательность действий такова.

1. Активизировать команду главного меню программы **Design | Add/Modify Constraint** (см. рис. 7.8) и указать подкоманду **Port**. Откроется диалоговое окно редактирования привязок порта (рис. 7.10).
2. В поле **Target** ввести имя порта. Для входа внешнего тактового импульса это порт **CLK**. В дальнейшем, по мере формирования привязки других портов их имена образуют в поле **Target** список, разворачивающийся (Dropdown List) по щелчку левой кнопки мыши на кнопке, расположенной в правом конце поля.
3. В поле **Constraint Kind** (Тип привязки) развернуть щелчком мыши список предопределенных типов привязок и, в нашем случае, выбрать тип **FPGA_CLOCK_PIN**, означающий, что на тактовый вход микросхемы ПЛИС будет подведен внешний "глобальный" тактовый сигнал.
4. В поле **Constraint Value** (Значение параметра привязки) выбрать из разворачивающегося списка логическое значение **True** (Истинно).

В результате этих действий в файл привязок добавляется группа:

```
Record=Constraint | TargetKind=Port | TargetId=CLK |  
FPGA_CLOCK_PIN=True
```

5. Повторяя рассмотренную последовательность действий, добавить в группу привязки порта CLK новые поля:
 - **FPGA_PINNUM** — номер контакта в цоколевке ПЛИС, на который выводится сигнал данного порта (см. рис. 7.1);

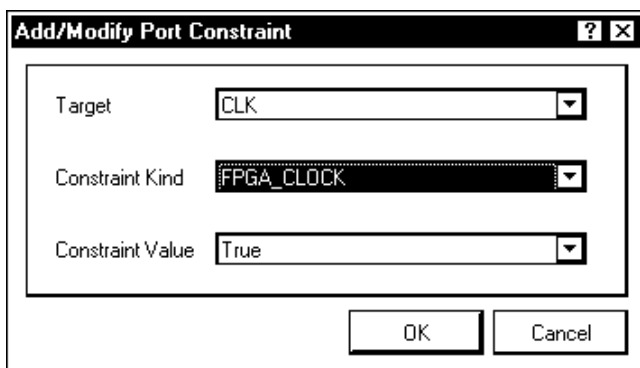


Рис. 7.10. Окно назначения привязок портов

- **FPGA_CLOCK_FREQUENCY** — частота тактовых импульсов, на которой должна работать выбранная ПЛИС;
 - **FPGA_DRIVE** — токовая нагрузка на данный контакт ПЛИС;
 - **FPGA_IOSTANDARD** — технологический стандарт, электрически совместимый со стандартом входных/выходных цепей выбранной ПЛИС: всего возможен выбор одного из более чем полусотни стандартов, но реальный выбор должен быть ограничен перечнем, приводимым в поле **Supported I/O Standards** диалогового окна выбора ПЛИС (см. рис. 7.9);
 - **FPGA_SLEW** — скорость нарастания сигнала на входных/выходных контактах микросхемы ПЛИС; допустимы два значения — Fast и Slow.
6. Переходя к другим портам схемы, составить группы привязок и для этих портов.

Листинг 7.2 иллюстрирует пример файла привязок.

Листинг 7.2

```
;.....
;Constraints File
; Device : Altera 7000s EPM7064STC100-10
; Board : User-defined PCB
; Project : Any
; Created 12.11.2008
;.....
Record=FileHeader | Id=DXP Constraints v1.0
Record=Constraint | TargetKind=Part | TargetId=EPM7064STC100-10
;.....
;....Global clock signal....
Record=Constraint | TargetKind=Port | TargetId=CLK |
FPGA_CLOCK_PIN=True
;....Input from Heiger counter....
Record=Constraint | TargetKind=Port | TargetId=HGR_IN |
FPGA_PINNUM=99
;....Start measurement....
Record=Constraint | TargetKind=Port | TargetId=START |
FPGA_PINNUM=100
```

```

;....Indicator outputs....

Record=Constraint | TargetKind=Port | TargetId=CMN_EL |
FPGA_PINNUM=37

;....First (most-significant) decade....

Record=Constraint | TargetKind=Port | TargetId=7SEGA[6..0] |
FPGA_PINNUM=6,8,9,10,12,13,14 |
FPGA_SLEW=SLOW,SLOW,SLOW,SLOW,SLOW,SLOW,SLOW,SLOW

;....Second decade....

Record=Constraint | TargetKind=Port | TargetId=7SEGB[6..0] |
FPGA_PINNUM=16,17,19,20,21,23,25 |
FPGA_SLEW=SLOW,SLOW,SLOW,SLOW,SLOW,SLOW,SLOW,SLOW

;....Third decade....

Record=Constraint | TargetKind=Port | TargetId=7SEGC[6..0] |
FPGA_PINNUM=29,30,31,32,33,35,36 |
FPGA_SLEW=SLOW,SLOW,SLOW,SLOW,SLOW,SLOW,SLOW,SLOW

;....Fourth decade....

Record=Constraint | TargetKind=Port | TargetId=7SEGD[6..0] |
FPGA_PINNUM=40,41,42,44,45,46,47 |
FPGA_SLEW=SLOW,SLOW,SLOW,SLOW,SLOW,SLOW,SLOW,SLOW

```

Отметим особенности формата записей в группах привязки портов, выполненных в схеме в "шинном" варианте. Для таких портов в поле **Target** диалогового окна (рис. 7.10) имя порта записывается так же, как в схеме, например 7SEGA[6..0]. В поле **FPGA_PINNUM** указываются через запятую номера всех контактов ПЛИС, на которые поступают сигнальные линии данного порта. Если указывается скорость нарастания сигнала на выводах, то в поле **FPGA_SLEW** так же через запятую значение **Fast** или **Slow** повторяется столько раз, сколько сигнальных линий объединяет "шинный" порт.

В качестве альтернативного варианта можно на данной стадии не назначать привязку портов к контактам ПЛИС, а предоставить выполнение этой привязки средствам САПР производителя ПЛИС (Vendor Tools) и после внутренней трассировки ПЛИС импортировать результат в файл привязок проекта.

7.4.4. Формирование конфигурации проекта

Для обеспечения возможности множественной реализации формируется набор конфигураций проекта. Конфигурация представляет собой именованный

список файлов привязки проекта к микросхемам ПЛИС различных типов, привязки к вариантам реализации на отладочной панели NanoBoard, пользовательском отладочном стенде, наконец, на печатной плате функционального узла, предназначенного для промышленного выпуска.

Для формирования конфигурации следует:

1. Активизировать команду главного меню программы **Project | Configuration Manager**. Откроется диалоговое окно **Configuration Manager For <имя проекта>** (рис. 7.11).

Центральное поле окна разделено на две части — **Constraint Files** и **Configurations**. В поле **Constraint Files** отображается список имен файлов привязки, входящих в структуру текущего активного проекта. Поле **Configurations** пусто.

2. Щелчком на кнопке **Add** (Добавить) против надписи **Configurations** активизировать образование новой конфигурации. Откроется диалоговое окно назначения имени новой конфигурации (поверх основного окна на рис. 7.11).
3. Ввести в поле **New Configuration Name** имя новой конфигурации и щелчком на кнопке **OK** завершить операцию. В поле **Configuration** появится подзаголовок, обозначающий новую конфигурацию.

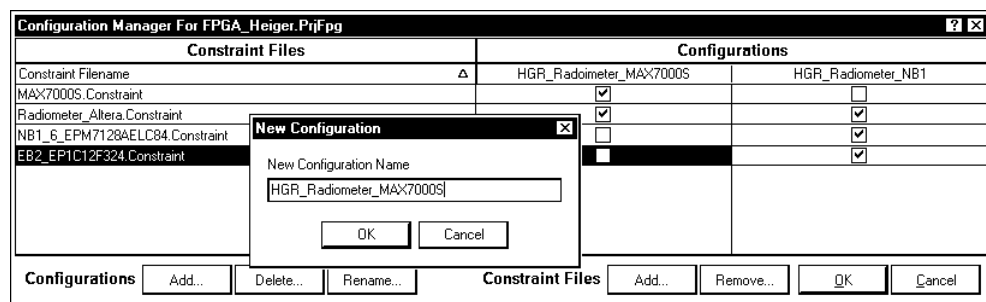


Рис. 7.11. Окно настройки конфигураций FPGA-проекта

4. Повторяя последние два действия, образовать необходимое количество новых конфигураций. Для каждой новой конфигурации в поле-таблице **Constraints** образуеться новая колонка.
5. При необходимости добавить к списку файлов привязки новые файлы щелчком на кнопке **Add** против надписи **Constraint Files** открыть стан-

дартный диалог поиска файлов. Имена найденных файлов подключаются к списку в поле **Constraint Files**.

Конфигурация составляется из одного или нескольких файлов привязки, относящихся к одному из возможных исполнений проекта. Для составления конфигурации нужно установить флажки-птички (Ticks) в колонках конфигураций на пересечении со строками файлов привязки. Ничто не мешает включить в одну конфигурацию несколько файлов привязки, однако программа контролирует, не включены ли в одну конфигурацию конфликтующие объекты — две микросхемы разных типов, две разные отладочные панели и т. п. При обнаружении таких конфликтов в плавающую панель **Messages** выводятся сообщения о фатальной ошибке, и конфигурация не составляется.

7.5. Выполнение проекта в оболочке Devices View

Следующий этап разработки FPGA-проекта — программирование ПЛИС — в свою очередь, делится на фазы компиляции (Compile) и синтеза проекта (Synthesize), в результате чего генерируется исходный файл для инструментов размещения логики проекта на выбранной ПЛИС и трассировки (коммутации) ее внутренних логических связей (Vendor Tools). Затем следует фаза "построения" (Build) — собственно работа инструментов размещения и трассировки, главным результатом чего является формирование файла двоичных данных для программирования ПЛИС (программы коммутации) и, наконец, фаза загрузки программы коммутации в "физическую" ПЛИС на отладочной панели или в аппаратуре пользователя (Program FPGA).

Все эти фазы выполняются в оболочке **Devices View**, которая активизируется командой главного меню **View | Devices View** или кнопкой **Devices** в линейке инструментов главного окна программы¹.

Для обеспечения процесса размещения логики проекта в выбранную ПЛИС и трассировки ее внутренних логических связей на компьютере необходимо установить "инструменты производителя" (Vendor Tools). В частности для микросхем фирмы Altera, используемых в рассматриваемом нами примере, это САПР Altera Quartus II.

¹ Документ AP0103 Processing the Captured FPGA Design.pdf. Application Note AP0103 (v.1.2) December 12, 2005.

Проектирование в оболочке **Devices View** может выполняться в "живой" (**Live**) или "неживой" (**Not Live**) модах.

Проектирование выполняется в режиме **Live**, если в комплект инструментов Altium Designer входит и подключена к компьютеру отладочная панель NanoBoard или пользовательская отладочная панель. В обоих случаях на отладочную панель должна быть установлена микросхема ПЛИС, на которой реализуется проект. В этой моде выполняются все четыре рассмотренные фазы разработки. Когда загрузка файла программирования выполнена, возможна отладка жесткой логики проекта и (или) встроенного в ПЛИС программного ядра с помощью виртуальных инструментов отладки Nexus Core Components, загружаемых в ПЛИС как и файл программирования коммутации.

Проектирование в режиме **Not Live** выполняется в случаях если:

- ☐ аппаратная отладочная панель отсутствует;
- ☐ на существующей отладочной панели нет возможности установить требуемую микросхему ПЛИС;
- ☐ цель разработчика — убедиться, возможно ли размещение логики проекта в выбранной им микросхеме ПЛИС или потребуется другая микросхема;
- ☐ ставится задача выполнения проекта в нескольких вариантах — на ПЛИС разных типов.

Во всех этих случаях выполняются три из четырех рассмотренных фаз — за исключением загрузки файла программирования коммутации ПЛИС. Этот файл или файлы (по числу вариантов выбора ПЛИС) формируются при выполнении фазы **Build**, сохраняются в памяти компьютера и могут быть загружены с компьютера в выбранные ПЛИС с помощью специального программатора или непосредственно в плату функционального узла в аппаратуре пользователя.

7.5.1. Работа в моде Live

Для программирования в режиме **Live** необходимо, чтобы к компьютеру была подключена отладочная панель NanoBoard NB-1¹ или отладочная панель пользователя с установленной в ней микросхемой ПЛИС, на которой выполняется разработка.

¹ Документ TR0102 NanoBoard Technical Reference manual-NB 1.pdf. Technical Reference TR0102 (v.1.2) February 8, 2006.

1. Активизировать команду главного меню **Tools | Preferences**, после чего в левом поле диалогового окна **Preferences** выбрать ветвь **FPGA | Devices View** и в правой панели окна установить активность опции **Live**.
2. Выполнить команду главного меню **View | Devices View**; откроется оболочка **Devices** (рис. 7.12).

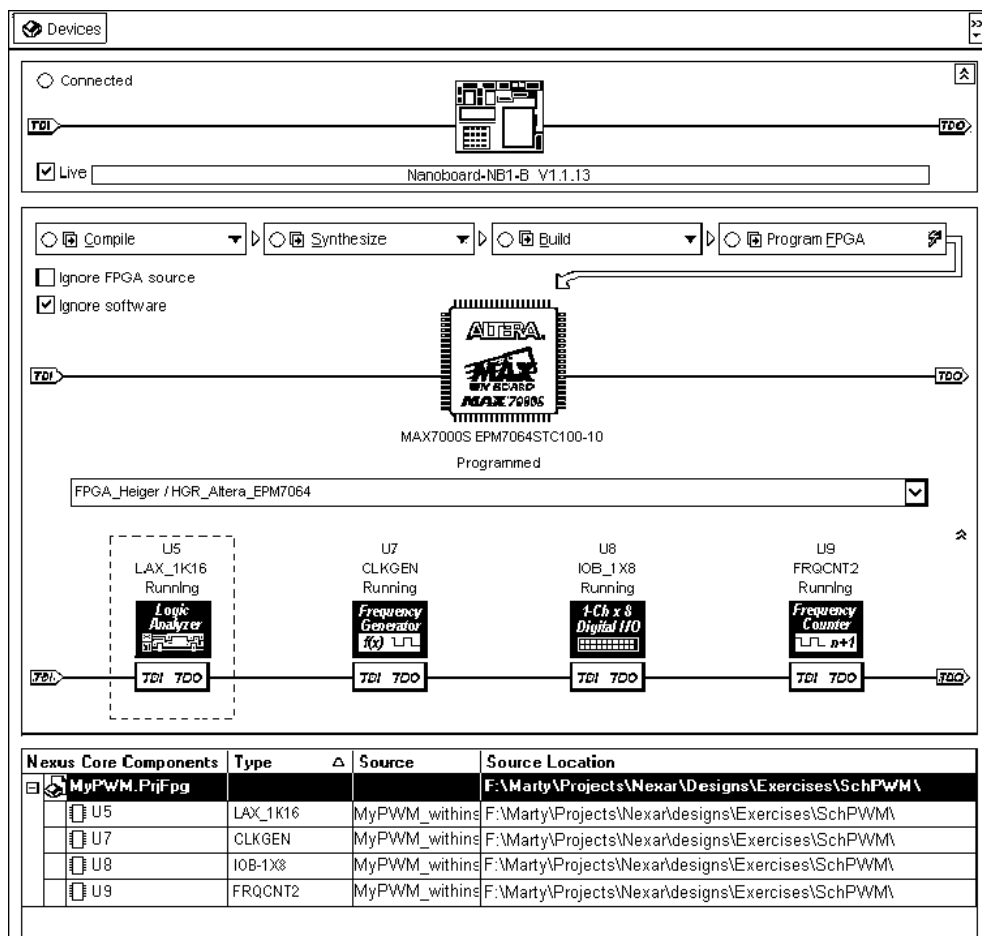


Рис. 7.12. Окно подсистемы Devices View
в режиме Live

Оболочка разделена по горизонтали на четыре области (Regions). В трех первых областях отображаются действия, выполняемые в трех потоках,

на которые делится процесс программирования ПЛИС. В лексике Altium Designer эти потоки названы цепями (Chains):

- **NanoBoard Controllers Chain** — цепь контроллеров отладочных панелей;
- **Hard Devices Chain** — цепь "твердых" приборов — микросхем ПЛИС;
- **Soft Devices Chain** — цепь программных моделей процессорного ядра микроконтроллеров, реализуемых в проекте;
- **Nexus Core Component listing** — список виртуальных инструментов в стандарте Nexus 5001¹, предназначенных для контроля и отладки ПЛИС после ее программирования.

При активизации оболочки **Devices** автоматически детектируется наличие включенных отладочных панелей и установленных на этих отладочных панелях микросхем ПЛИС. В поле **NanoBoard Controllers Chain** графически отображается цепь отладочных панелей, соединенных между собой и с компьютером через входные и выходные линии JTAG-интерфейса. Факт подключения отладочных панелей отображается также кнопкой-индикатором зеленого цвета и надписью **Connected** в левой части поля **NanoBoard Controllers Chain**. Там же, в левой части этого поля, отображается активность моды **Live**.

В области **Hard Devices Chain** графически отображается состав микросхем ПЛИС, установленных на всех включенных отладочных панелях, обозначенных в файлах привязки (**Constraint Files**) и включенных в конфигурации проекта (**Configurations**). Каждая микросхема отображается условным значком (Icon) и расположенным под значком обозначением типа микросхемы.

В верхней части области располагаются четыре поля управления фазами обработки проекта — компиляции, синтеза, построения (внутренней структуры ПЛИС) и загрузки программы коммутации в ПЛИС.

В поле, расположенном ниже цепочки микросхем ПЛИС, раскрывается выпадающий (Drop-Down) список, из которого можно выбрать проекты и конфигурации, нацеленные на выполнение на выбранной микросхеме (микросхемах).

¹ Документ AR0130 PC to NanoBoard Communications.pdf. Article AR0130 (v.1.1) May 27, 2005.

3. При необходимости поиска других микросхем для реализации текущего активного проекта можно активизировать такой поиск. Для этого щелчком правой кнопки мыши в поле **Hard Devices Chain** выбрать функцию **Add** (рис. 7.13), после чего щелчком левой кнопки мыши на стрелочке в поле **Add** раскрыть выпадающее командное меню, содержащее список имеющихся в проекте конфигураций и команды просмотра (**Browse**) и поиска (**Search**), поддерживаемых программой ПЛИС.

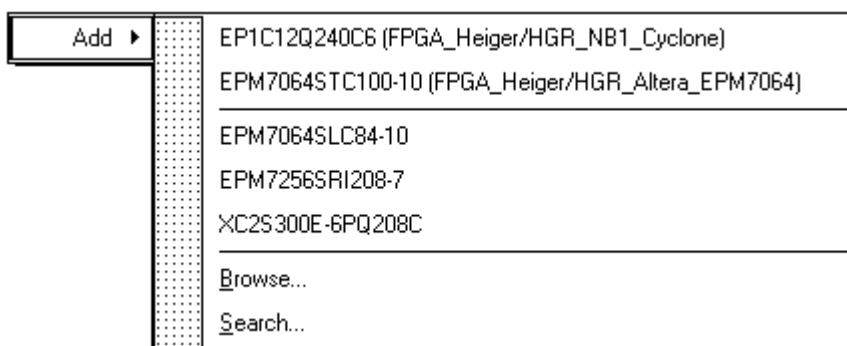


Рис. 7.13. Команды просмотра конфигураций и поиска ПЛИС

4. Команда **Browse** открывает рассмотренное ранее диалоговое окно выбора **Choose Physical Device** (см. рис. 7.9), в котором можно изменить сделанный ранее выбор ПЛИС.

При активизации команды **Search** откроется диалоговое окно поиска **Find Physical Device** (рис. 7.14). В поле **Search Criteria** необходимо назначить критерии поиска. Упомянем здесь только те из них, которые могут интересовать нас с точки зрения рассматриваемого примера проекта ПЛИС:

- **Device Type** — тип ПЛИС: выбираем CPLD;
- **Vendor** — производитель: выбираем фирму Altera;
- **Min. IO Pin Count** — минимальное число входных/выходных контактов;
- **Min. Macrocells** — минимальное число макроячеек в кристалле ПЛИС;
- **With IO Standard** — технологический стандарт входных/выходных линий ПЛИС;
- **Min. Global Clock Resources** — минимальное число глобальных тактовых входов.

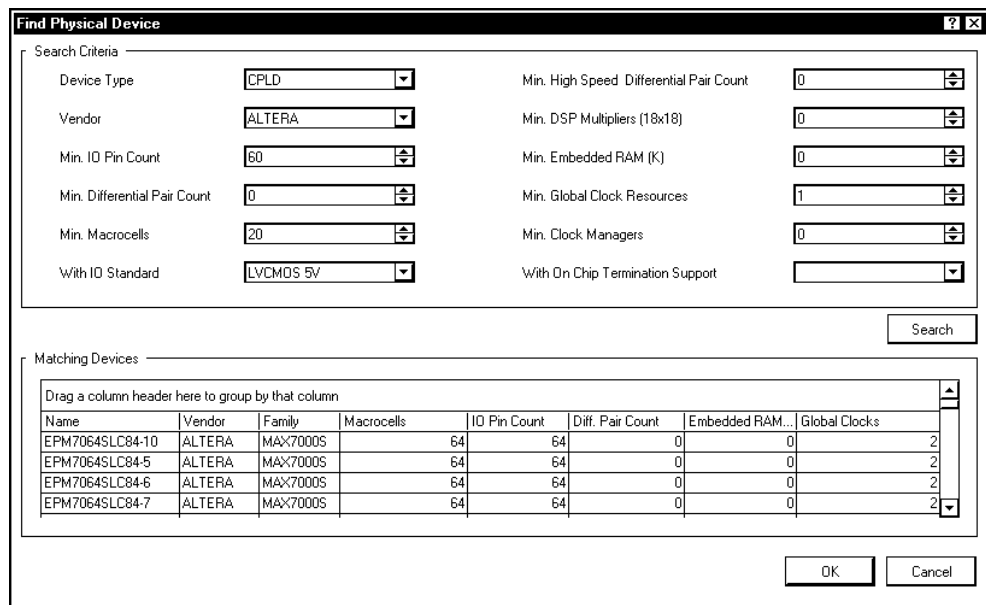


Рис. 7.14. Окно формирования критериев и управления поиском ПЛИС

- Щелчком на кнопке **Search** запустить процедуру поиска. Результаты поиска отображаются в виде списка в поле **Matching Devices** — приборы, отвечающие назначенным критериям. Выбрать курсором строку в этом списке и щелчком на кнопке **OK** подключить новую микросхему к набору ранее выбранных. Она ставится последовательно в существующую цепь и отображается в области **Hard Devices Chain**.
- Если вновь выбранная ПЛИС не соответствует ни одной из установленных конфигураций проекта, то в нижележащем поле выводится красным цветом сообщение: **[No Compatible Project Configurations Found]**. Для того чтобы вновь выбранную ПЛИС можно было использовать, следует вернуться к формированию новых файлов привязки и составить из них новую конфигурацию.
- Для исключения неподходящей ПЛИС из цепи щелчком правой кнопки мыши на ее значке нужно вызвать контекстное меню и указать в нем команду **Remove**.

В области **Soft Devices Chain** отображается цепочка моделей процессорного ядра программируемых контроллеров в стандарте Nexus 5001, если предполагается их реализация на выбранной ПЛИС. Они также связаны в последо-

вательную цепь линиями JTAG-интерфейса. Если проект не предполагает наличия моделей процессорного ядра, область остается пустой.

В области **Nexus Core Components Listing** отображается дерево текущего открытого проекта, вложенных проектов и все входящие в них компоненты стандарта Nexus 5001 — модели процессорного ядра микроконтроллеров, блоки оперативной памяти (RAMXXX), виртуальные инструменты отладки.

Обмен данными между компьютером и отладочной панелью NanoBoard, формируемыми в трех потоках процесса обработки проекта, происходит по линиям JTAG-интерфейса. Эти данные мультиплексируются контроллером отладочной панели, а при поступлении в компьютер распределяются по цепям **NanoBoard Controller Chain**, **Hard Devices Chain** и **Soft Devices Chain**.

7.5.2. Работа в моде *Not Live*

В моду **Not Live** можно перейти из моды **Live** непосредственно в оболочке **Devices View**, сняв активность опции **Live** в области **NanoBoard Controllers Chain** (см. рис. 7.12), либо по команде **Tools | Preferences | FPGA | Devices View** в главном меню программы.

Диалоговое окно оболочки **Devices View** приобретает вид, показанный на рис. 7.15. В область контроллеров отладочных панелей выводится сообщение об отсутствии подключения к "физическим" приборам **Not Connected To Any Physical Devices**.

В области **Hard Devices Chain** активными остаются только три поля управления процессом — **Compile**, **Synthesize** и **Build**. Поле **Program FPGA** затенено и недоступно для работы, поскольку в моде **Not Live** не предполагается загрузка кода программирования в ПЛИС.

Поскольку в моде **Not Live** ПЛИС, на которой выполняется проект, не детектируется автоматически, она должна быть включена в оболочку **Devices View** по команде **Add**, активизируемой щелчком правой кнопки мыши в области **Hard Devices Chain**. В контекстном меню показаны микросхемы ПЛИС, для которых в структуре проекта назначены конфигурации (см. рис. 7.13). При указании курсором на одну из них со щелчком левой кнопки мыши выбранная микросхема включается в оболочку **Devices View** и отображается в виде значка в поле **Hard Devices Chain** (рис. 7.15).

Поскольку физическое подключение ПЛИС к аппаратным средствам программирования и отладки отсутствует, линии ее интерфейса неактивны (затенены).

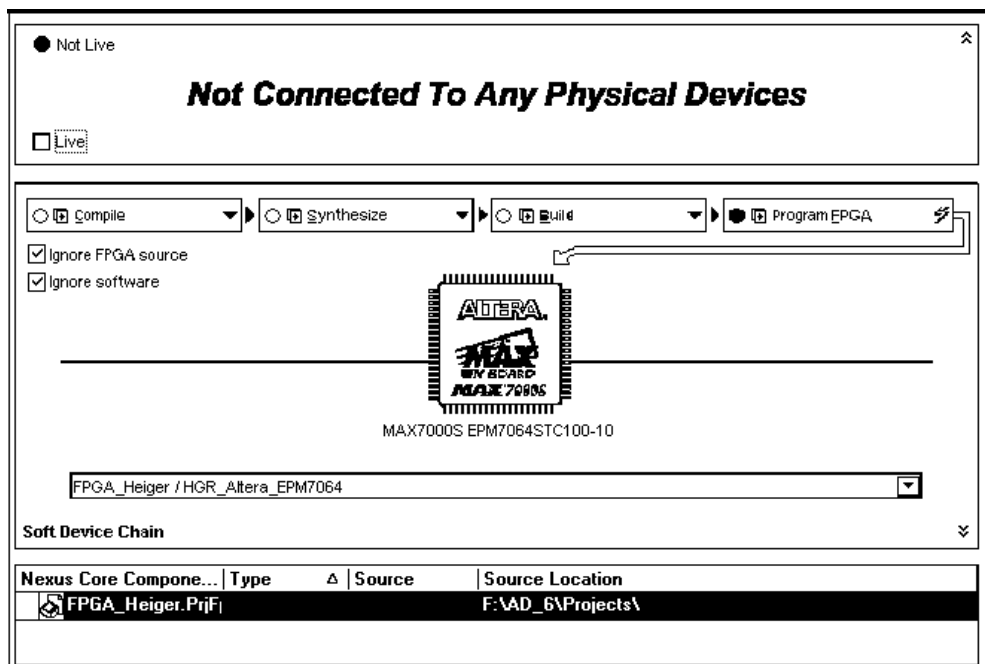


Рис. 7.15. Окно подсистемы **Devices View** в режиме **Not Live**

В моде **Not Live** также доступны средства поиска, выбора и замены ПЛИС, рассмотренные в предыдущем разделе (см. рис. 7.13 и 7.14).

При необходимости реализации проекта в нескольких вариантах требуемые микросхемы ПЛИС отбирают средствами поиска и составляют в цепочку в области **Hard Devices Chain**. Командой главного меню **Tools | Devices List** можно активизировать окно просмотра **Design Workspace — Physical devices List**, в котором отображается список выбранных микросхем ПЛИС с краткой технической характеристикой каждой из них. Если для каждой микросхемы в проекте определена конфигурация, командой **Compile All Bit Files** можно запустить формирование двоичных файлов программирования для всех ПЛИС, составляющих цепочку.

7.6. Программирование ПЛИС

Как уже было сказано, завершающая стадия выполнения проекта ПЛИС включает четыре фазы: компиляцию, синтез, построение и загрузку програм-

мы коммутации в выбранную ПЛИС. Первые три фазы могут выполняться как в режиме **Live**, так и в режиме **Not Live**. Управляют этими фазами с помощью команд, выведенных на поля-кнопки в области **Hard Devices Chain** оболочки **Devices View** (см. рис. 7.12, 7.15).

7.6.1. Компиляция

Компиляция проекта запускается щелчком на кнопке **Compile**. В процессе компиляции выполняется проверка исходных документов проекта на отсутствие электрических, графических ошибок и ошибок кодировки. Кроме того, если проект содержит Nexus-компоненты (процессорное ядро микроконтроллера, блоки памяти, виртуальные инструменты отладки), для каждого такого компонента генерируется шестнадцатеричный файл, включаемый в структуру проекта. Входящие в структуру Nexus-компоненты выстраиваются в цепочку в области **Soft Devices Chain** оболочки **Devices View**.

Если компиляция прошла успешно, информация о структуре проекта образует файл структуры проекта с именем проекта и расширением `PrjFpgStructure`. Этот файл включается в дерево структуры проекта и отображается в плавающей панели **Projects**.

7.6.2. Синтез

Назначение данной фазы — подготовка исходных данных для инструментов размещения и трассировки внутренних связей ПЛИС от производителя микросхем (Vendor Tools). На этой стадии выясняется, возможен синтез или нет.

Если синтез проходит успешно, конечным результатом является файл описания всей структуры проекта (Netlist) на языке обменного формата EDIF (Electronic Design Interchange Format). Синтез выполняется в несколько шагов:

1. Синтезируются модели ядра логических блоков, входящих в состав проекта ПЛИС. Эти модели содержат компилированные шестнадцатеричные данные из вложенных проектов.
2. Для каждого листа схемы проекта формируется промежуточное описание в конструкциях языка описания аппаратуры VHDL. Эти описания конвертируются и объединяются в файл верхнего уровня на языке EDIF.

3. Для конкретной выбранной микросхемы ПЛИС программа отыскивает в библиотечном каталоге ...\\Altium Designer\\Library\\EDIF\\ и копирует в выходной каталог ...\\Output2\\ файлы синтезированных EDIF-моделей компонентов схемы и функциональных узлов внутренней структуры ПЛИС.

Допускаются как системные модели, так и модели, синтезированные пользователем. Размещение каталога пользовательских моделей определяется в диалоге настройки **Tools | FPGA Preferences** на странице **FPGA – Synthesis** диалогового окна.

7.6.3. Построение

В фазе **Build** из оболочки **Devices View** запускают инструменты размещения и трассировки логики проекта на кристалле ПЛИС (Vendor Tools). На этой фазе можно решить задачи разного рода:

- ☐ проверить, размещается ли логика проекта в выбранной ПЛИС;
- ☐ получить данные о подключении внешних логических сигналов к выводам выбранной ПЛИС для импорта их в Altium Designer и формирования в его среде файлов привязки.

При выполнении фазы построения в полном объеме конечным результатом является файл программирования ПЛИС, который в итоге загружается в выбранную микросхему и обеспечивает ее функционирование в соответствии с замыслом разработчика проекта. Это достигается за пять шагов, названия которых разворачиваются по щелчку левой кнопки мыши на кнопке **Build**:

- ☐ **Translate Design** — конверсия EDIF-файла проекта верхнего уровня и присоединенных файлов синтезированных моделей, полученных с фазы синтеза, в формат внутренней базы данных (Native Generic Database) специализированной САПР производителя ПЛИС (Vendor Tools);
- ☐ **Map Design to FPGA** — формирование логики проекта из логических примитивов внутренней структуры ПЛИС;
- ☐ **Place and Route** — использование данных из описания нижнего уровня, полученного на предыдущем шаге для размещения необходимой логики на кристалле ПЛИС и последующей коммутации внутренних логических связей;

- ❑ **Timing Analysis** — временной анализ проекта в соответствии с настройками временных параметров, назначенными в файлах привязки;
- ❑ **Make Bit File** — формирование двоичного файла программирования, предназначенного для загрузки в выбранную ПЛИС.

По окончании фазы построения (Build) открывается диалоговое окно, в котором представлены сведения о занятых ресурсах кристалла ПЛИС при размещении в нем логики проекта (рис. 7.16).

Results Summary

Device Resources - Usage summary

4-Input LUTs - Logic	3,245 / 6,144	52%
4-Input LUTs - Total	3,489 / 6,144	56%
Block RAMs	4 / 16	25%
DLLs	1 / 4	25%
Global Clock I/O Buffers	1 / 4	25%
Global Clocks	3 / 4	75%
I/O Pins	26 / 142	18%
Slice Flip Flops	1,235 / 6,144	20%
Slices with only related logic	2,143 / 2,143	100%
Slices with unrelated logic	0 / 2,143	0%
Slices	2,143 / 3,072	69%
Tbuts	8 / 3,200	1%

Design Statistics - Timing summary

JTAG_NEXUS_TCK (Constraint = 1.000 MHz)	54.708 MHz
Maximum Frequency	10.485 MHz
Maximum combinatorial path delay	5.603 ns
Minimum input required time before clock	16.470 ns
Maximum output delay after clock	28.257 ns

☒ Show Results Summary dialog
 Note: The Results Summary also appears in the Output panel

Print

Copy

Report

Close

Рис. 7.16. Сводка данных об используемых ресурсах ПЛИС

В нижней части окна размещаются сведения о тактовых частотах и временных задержках логических сигналов. Та же самая информация выводится в плавающую панель **Output**.

Для микросхем ПЛИС фирмы Xilinx к этим шагам добавляется еще один — **Make PROM File** — формирование файла программирования конфигурационного ПЗУ, с которого в ПЛИС при включении питания функционального узла загружается программа коммутации внутренних логических связей.

7.6.4. Загрузка программы коммутации ПЛИС

Данная фаза процесса выполняется только в том случае, если к компьютеру подключена отладочная панель NanoBoard или пользовательская отладочная панель с установленной на ней микросхемой ПЛИС.

Второе условие, при котором выполняется данная фаза, — успешное завершение всех предыдущих фаз процесса.

Файл программирования загружается в ПЛИС через мультиплексируемый последовательный канал JTAG-интерфейса, действующий под управлением контроллера отладочной панели NanoBoard. Ход программирования ПЛИС отображается в линейке статуса. По окончании программирования сообщение **Reset** под значком ПЛИС в области **Hard Devices Chain** меняется на **Programmed**.

Отметим еще ряд особенностей выполнения стадий работы в оболочке **Devices View**.

□ Кнопки запуска стадий процесса multifunctional:

- щелчком кнопки мыши на текстовой надписи запускается только текущая стадия процесса;
- щелчком на значке в левой части поля кнопки запускаются все предыдущие и текущая стадия;
- щелчком на "сигнальной лампочке" в левом конце или на стрелке в правом конце поля кнопки разворачивается список шагов текущей стадии процесса с цветовой индикацией, отображающей ход выполнения каждого шага, при этом можно активизировать диалог настройки опций для каждого шага.

□ Цвет "сигнальной лампочки" в левом конце поля кнопки отображает ход и успех или неудачу при выполнении текущей стадии или отдельных ее шагов:

- серый — **Not Available** — стадия недоступна для исполнения;
- красный — **Missing** — данная стадия или ее шаг ранее не выполнялись (в том числе перед первым исполнением стадии или шага);

- желтый — **Out of Date** — исходный файл был изменен, поэтому требуется повторное выполнение стадии или шага, чтобы согласовать результаты стадий;
 - синий — **Running** — происходит выполнение текущей стадии или шага;
 - оранжевый — **Cancelled** — выполнение стадии или шага остановлено пользователем;
 - сиреневый — **Failed** — при выполнении стадии или шага произошла ошибка;
 - зеленый — **Up to Date** — выполнение стадии или шага прошло успешно, все результаты согласуются.
- ❑ Выполнение процесса на стадии **Build** можно остановить нажатием клавиши <Esc> на клавиатуре. На стадиях **Compile** и **Synthesize** имеется свой диалог прогресса, в котором выполнение стадии можно прервать. Стадию загрузки файла программирования в ПЛИС (**Program FPGA**) нельзя остановить, она выполняется до конца.
- ❑ Наблюдать за ходом выполнения стадий формирования логической структуры и программирования ПЛИС структуры в текущем времени можно по сообщениям в плавающей панели сообщений (**Messages**).
- ❑ Микросхеме ПЛИС можно присвоить индивидуальный идентификационный код. Для этого следует двойным щелчком левой кнопки мыши на имени проекта в поле **Nexus Core Components** активизировать диалог настройки опций FPGA-проекта и открыть вкладку **Options** (рис. 7.17).

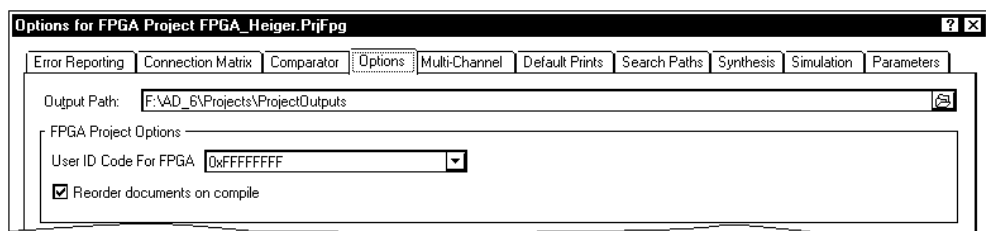


Рис. 7.17. Присвоение микросхеме ПЛИС индивидуального идентификационного кода

Ввести в поле **User ID For FPGA** шестнадцатеричный код идентификации. Этот код появляется в панели **Devices View** под изображением значка микросхемы. Такая кодировка полезна в случаях, когда на печатной плате PCB-проекта верхнего уровня иерархии стоит несколько однотипных микросхем ПЛИС.

7.7. Встроенные инструменты контроля и отладки

При отладке изготовленных радиоэлектронных функциональных узлов обычно применяют электро- и радиоизмерительные приборы для исследования сигналов в разных точках схемы. Существуют контрольные стенды с матричным расположением множества щупов, снимающих сигналы в сотнях точек печатного узла, сканированием этих щупов и отображением сигналов на мониторе логического анализатора (bed-of-nails testing: дословно — тестирование на кровати с гвоздями). Применить такую практику контроля к интегральным микросхемам чрезвычайно затруднительно из-за их малых размеров, наличия защиты кристалла и отсутствия доступа ко многим узлам электрической схемы.

Средства Altium Designer позволяют организовать такой контроль за счет использования ресурсов кристалла ПЛИС, не задействованных при реализации логической схемы проектируемого устройства. Библиотека **FPGA Instruments.IntLib** содержит набор встраиваемых в кристалл инструментов, виртуальные измерительные щупы которых подключаются к узлам электрической схемы, реализуемой в кристалле ПЛИС, и обеспечивают наблюдение сигналов в этих узлах. Для этого необходимо извлечь из библиотеки **FPGA Instruments.IntLib** и ввести эти инструменты в схему как обычные компоненты, присоединив их электрические выводы к необходимым узлам схемы. В оболочке **Devices View** в **Live**-моде виртуальные инструменты наблюдаются в области **Soft Devices Chain** и включаются там в цепочку с другими программно-эмулируемыми компонентами. При размещении и трассировке логической схемы в кристалле ПЛИС для их реализации выделяется некоторая часть площади кристалла. Настройка и управление работой встроенных виртуальных инструментов активизируется двойным щелчком левой кнопки мыши на значке инструмента в поле **Soft Devices Chain** оболочки **Devices View**. На экране монитора компьютера открывается вид передней панели, напоминающий переднюю панель современных цифровых измерительных

приборов, набираемых в стойку (рис. 7.18). Отсюда общее название этого набора инструментов — Instrument Rack.



Рис. 7.18. Панель встроенного логического анализатора

Набор встраиваемых инструментов включает:

- ❑ CLKGEN — генератор тактовой частоты, значение которой устанавливается щелчком мыши на одной из экранных кнопок передней панели или из диалогового окна настройки;
- ❑ FRQCNT2 — двухканальный цифровой частотомер, измеряющий частоту сигнала, период или число импульсов за заданный временной интервал;
- ❑ IOB_x — модули входных/выходных буферов¹, через которые можно наблюдать или активизировать логические сигналы в узлах схемы устройства, реализованного в ПЛИС. Варианты исполнения — от одного до четырех каналов 8- или 16-разрядного параллельного кода. Логические сигналы на входах отображаются в виде шестнадцатеричного числа и линейкой светодиодов. Выходные сигналы отображаются в виде двоичного и шестнадцатеричного числа в каждом канале;
- ❑ LAX_x — логический анализатор, подключаемый к необходимым узлам схемы, конфигурируемый в диалоге настройки и представляющий на экране монитора временные диаграммы дискретных (цифровых) или аналоговых сигналов в этих узлах².

Отличие встроенных инструментов Altium Designer от эмулируемых виртуальных инструментов отладки состоит в том, что это реальные измерительные инструменты, встроенные в кристалл. Информация, которую получает пользователь, снимается с узлов физического прибора ПЛИС.

¹ Документ CR0102 IOB_x Digital I/O Module.pdf Core reference (v.1.2) December 12, 2005.

² Документ CR0103 LAX_x Logic Analyzer.pdf Core reference CR0103 (v.1.2) December 12, 2005.

Кроме этих встраиваемых инструментов Altium Designer располагает виртуальными инструментами управления функциями контроллера отладочной панели NanoBoard NB-1 и средствами отладки программного ядра микроконтроллеров, встраиваемых в ПЛИС. Эти инструменты активизируют двойным щелчком левой кнопки мыши на значке контроллера отладочной панели в области **NanoBoard Controllers Chain** или на значке эмулируемого компонента в зоне **Soft Devices Chain** оболочки **Devices View**.

На отладочной панели NanoBoard имеются средства поддержки встроенных инструментов, обеспечивающие их связь с компьютером, на котором ведется разработка проекта. Все виртуальные инструменты связываются с компьютером через программно эмулируемую JTAG-цепочку, работающую в Nexus-стандарте и отображаемую в зоне **Soft Devices Chain** оболочки **Devices View**. Для того чтобы иметь возможность работы со встроенными инструментами и программно эмулируемыми компонентами FPGA-проекта следует разместить на листе схемы верхнего уровня иерархии два компонента — NEXUS_JTAG_PORT и NEXUS_JTAG_CONNECTOR, хранящиеся в библиотеках, соответственно, FPGA Generic.IntLib и FPGA NanoBoard Port-Plugin.IntLib (рис. 7.19).

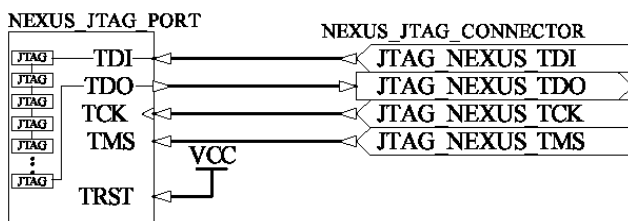
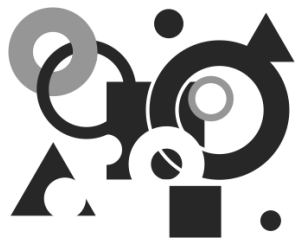


Рис. 7.19. Линии JTAG-интерфейса ПЛИС

Для физической реализации этого канала выбранная ПЛИС должна располагать необходимым резервом аппаратных ресурсов, в том числе четырьмя выводами для портов JTAG-коннектора.



Объединение проекта ПЛИС с проектом печатной платы

В результате выполнения проекта ПЛИС мы получаем микросхему, запрограммированную на отладочной панели или готовую к программированию на печатной плате функционального узла, разрабатываемого в рамках РСВ-проекта. Технология сквозного проектирования предполагает параллельное выполнение FPGA-проекта логического ядра и РСВ-проекта функционального узла. В конце концов наступает момент, когда необходимо объединить оба проекта. Это достигается, если логическое ядро функционального узла, выполненное в ПЛИС, оформляется как схемный документ и включается в электрическую принципиальную схему РСВ-проекта на правах иерархического схемного компонента. Связность проектов обеспечивается через имена цепей: необходимо, чтобы входные и выходные порты микросхемы ПЛИС и метки цепей, подключаемых к схемному элементу ПЛИС в РСВ-проекте верхнего уровня, имели одинаковые имена. Необходима также возможность передачи изменений, вносимых как в FPGA-, так и в РСВ-проект, из одного проекта в другой и обратно (синхронизация проектов).

8.1. Мастер объединения проектов

Связь проектов FPGA и РСВ и их синхронизация легче всего достигаются, если использовать функции мастера **FPGA to PCB Project Wizard**. В его диалоговых окнах-страницах можно выполнить такие шаги FPGA-проекта, как выбор микросхемы ПЛИС, формирование файла привязок и конфигурации проекта, назначение логических сигналов проекта на выводы выбранной ПЛИС. Все это было описано в предыдущих разделах книги, будем считать,

что данные шаги выполнены, и нет необходимости повторять их при рассмотрении функций мастера.

8.1.1. Выбор конфигурации проекта

Для использования функций мастера следует:

1. В среде открытого схемного документа FPGA-проекта выполнить команду главного меню **Tools | FPGA to PCB Project Wizard** и щелчком мышью на кнопке **Next** перейти на страницу определения конфигурации проекта (рис. 8.1).

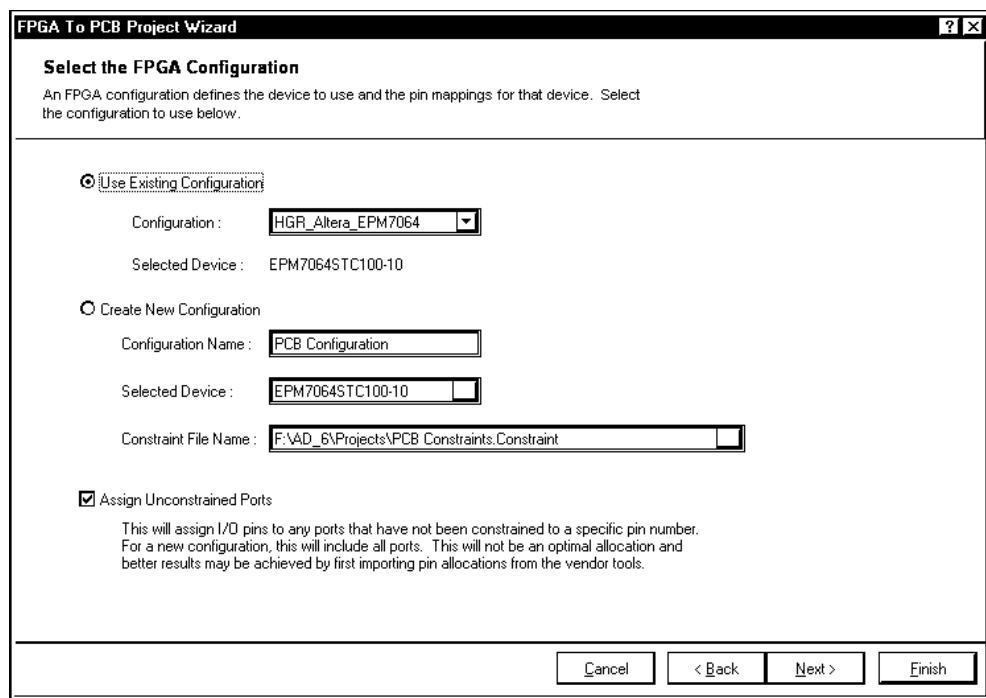


Рис. 8.1. Окно мастера объединения проектов

2. На странице **Select the FPGA Configuration** нажать кнопку выбора **Use Existing Configuration** (Использовать существующую конфигурацию). Имя текущей активной конфигурации проекта отображается в поле выбора **Configuration**.

3. В поле **Selected Device** отображается название выбранной ПЛИС.

Если конфигурация ранее не назначалась, ее можно образовать, нажав кнопку **Create New Configuration** (Создать новую конфигурацию).

4. В полях **Configuration Name** и **Constraint File Name** можно оставить значения по умолчанию или назначить новые имена конфигурации и файла привязок.
5. В поле **Selected Device** (Выбранный прибор) можно открыть диалог поиска микросхемы ПЛИС, рассмотренный в *разд. 7.4.2* (см. рис. 7.9).

8.1.2. Привязка портов логической схемы к выводам ПЛИС

Привязка портов логической схемы нашего проекта к выводам выбранной ПЛИС была выполнена ранее, на стадии формирования файла привязок. Эти привязки передаются в САПР производителя ПЛИС (Vendor Tools), где используются для генерации файлов программирования ПЛИС. В среде мастера объединения проектов эти привязки требуются при формировании схемного элемента, включаемого в PCB-проект. Включение можно выполнить одним из следующих способов.

- ❑ Если к моменту работы по объединению FPGA- и PCB-проектов уже выполнены синтез и построение FPGA-проекта в среде САПР производителя ПЛИС (Vendor Place and Route Tools), то привязки можно импортировать из среды этой специализированной САПР командой главного меню программы **Tools | Import FPGA Pin File** (Инструменты | Импортировать файл выводов ПЛИС).
- ❑ В открытом схемном документе активизировать команду главного меню программы **Tools | FPGA Signal Manager** (Менеджер сигналов ПЛИС). Откроется диалоговое окно с таблицей, в которой отображен набор привязок, выполненных при формировании файла привязок проекта (рис. 8.2). При этом назначения для групповых (шинных) портов в данную таблицу не передаются. Недостающие привязки можно задать вручную для каждой индивидуальной логической линии в колонках таблицы окна (см. рис. 8.2).

Недостающие привязки можно также назначить щелчком мышью на кнопке **Assign Unconstrained Signals**. В обоих случаях эти привязки не совпадают с ранее назначенными в файле привязок. Таким образом, данная функция позволяет назначить привязки на "голый" проект.

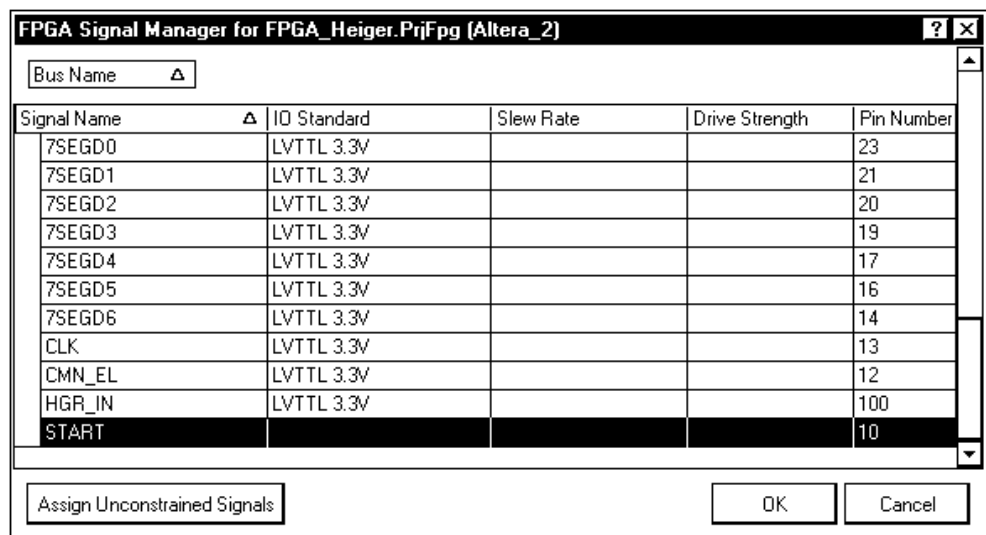


Рис. 8.2. Окно "сигнального менеджера"

- Аналогично предыдущему случаю, автоматическую привязку портов логической схемы проекта к выводам выбранной ПЛИС можно выполнить в среде "мастера" щелчком на кнопке **Assign Unconstrained Ports** (см. рис. 8.1). Если привязка ранее не назначалась, операция затронет все порты логической схемы. Если же привязки назначены, активизировать данную функцию *не следует*, поскольку при ее выполнении программа игнорирует привязки индивидуальных линий, собранных в шинные порты, и добавляет в группы файла привязок свои назначения, не удаляя исходные. В результате в группах привязки шинных портов оказываются два одноименных поля `FPGA_PINNUM` с разными назначениями контактов ПЛИС. Кроме того, порт входного тактового сигнала назначается не на предопределенный вывод глобального тактового сигнала ПЛИС, а на один из пользовательских контактов. Все это нарушает первоначальный замысел автора разработки и приводит к ошибкам компиляции проекта.
- Самый трудоемкий способ — назначить привязки вручную в готовом автоматически сгенерированном схемном документе. Применение этого способа нежелательно, поскольку кроме большого объема работы по назначению привязок каждого сигнала необходим еще дополнительный шаг синхронизации проектов для передачи выполненных назначений в файл привязок проекта.

8.1.3. Объявление имени РСВ-проекта

Поскольку в нашем проекте микросхема ПЛИС выбрана и не должна меняться, файл привязок сформирован и конфигурация определена, щелчком на кнопке **Next** переходим на следующую страницу мастера. На этой странице выполняется единственное действие — подтверждается существующее или назначается новое имя РСВ-проекта верхнего уровня (рис. 8.3).

PCB Project File Name :

F:\AD_6\Projects\Radiometer.PrjPcb

Рис. 8.3. Назначение имени РСВ-проекта

При назначении нового имени РСВ-проекта программа выдает диалоговое окно с предложением выбрать формат файла проекта — Protel PCB или P-CAD PCB. Определить формат проекта и щелчком на кнопке **Next** перейти на следующую страницу мастера.

8.1.4. Генерация схемного листа проекта

На странице **Configure the Main Sheet** (Конфигурировать главный лист), (рис. 8.4) строится лист схемного документа РСВ-проекта, содержащий изображение символа ПЛИС с присвоенными выводам микросхемы обозначениями цепей и изображениями портов логической схемы проекта.

Последовательность действий такова.

1. Назначить размерность для графических документов РСВ-проекта — метрическую (**Metric**) или дюймовую (**Imperial**) систему единиц измерения.
2. В поле **Main Sheet File Name** выбрать из выпадающего списка существующих или ввести взамен назначенного программой по умолчанию имя файла схемного листа, включаемого в структуру РСВ-проекта.
3. В поле **Component Designator** ввести позиционное обозначение для формируемого схемного документа.
4. Оставить активными опции **Use Standard Sheet Size Where Possible** (Использовать лист стандартного размера там, где возможно) и **Connect Power Pins via Power Ports** (Соединить контакты питания (и "земли") через силовые порты схемы).

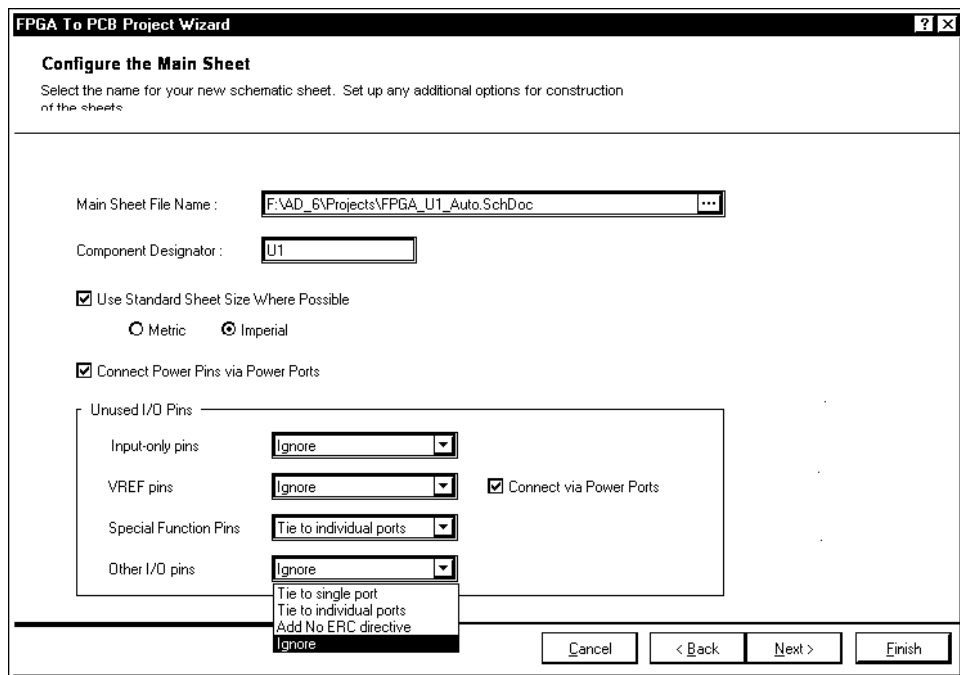


Рис. 8.4. Настройка конфигурации иерархического схемного символа ПЛИС

5. В поле **Unused I/O Pins** указать, как следует обходиться с неподключенными входными и выходными контактами ПЛИС. Возможны следующие варианты выбора (см. рис. 8.4):

- **Tie to single port** — связать все выводы с одним общим портом;
- **Tie to individual ports** — связать каждый вывод с индивидуальным портом;
- **Place No ERC directive** — снабдить вывод директивой No ERC (не применять электрический контроль): в этом случае при компиляции не будут выводиться сообщения об ошибках;
- **Ignore** — игнорировать, т. е. никуда не присоединять.

Применительно к выводам, предназначенным для выполнения специальных функций (**Special Function Pins**), — входам тактовых сигналов, начальной установки, выводам встроенного JTAG-контроллера — назначим присоединение к индивидуальным портам (**Tie to individual ports**).

Для остальных неиспользуемых выводов зададим признак **Ignore**.

После завершения рассмотренных настроек щелчком на кнопке **Next** переходим на следующую страницу мастера объединения FPGA- и PCB-проектов **Configure Sheet Symbol Sheet** (рис. 8.5).

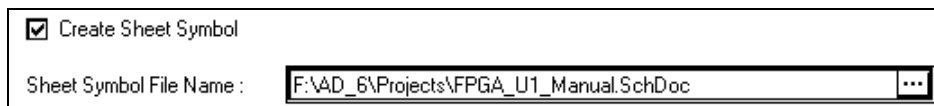


Рис. 8.5. Назначение имени иерархического схемного символа ПЛИС

На этой странице следует выполнить единственную настройку — назначить имя схемного документа для иерархического компонента PCB-проекта:

1. Активизировать опцию **Create Sheet Symbol** (создать символ схемного элемента).
2. В поле **Sheet Symbol File Name** выбрать из выпадающего списка одно из существующих имен или ввести с клавиатуры новое имя файла схемного документа иерархического компонента.

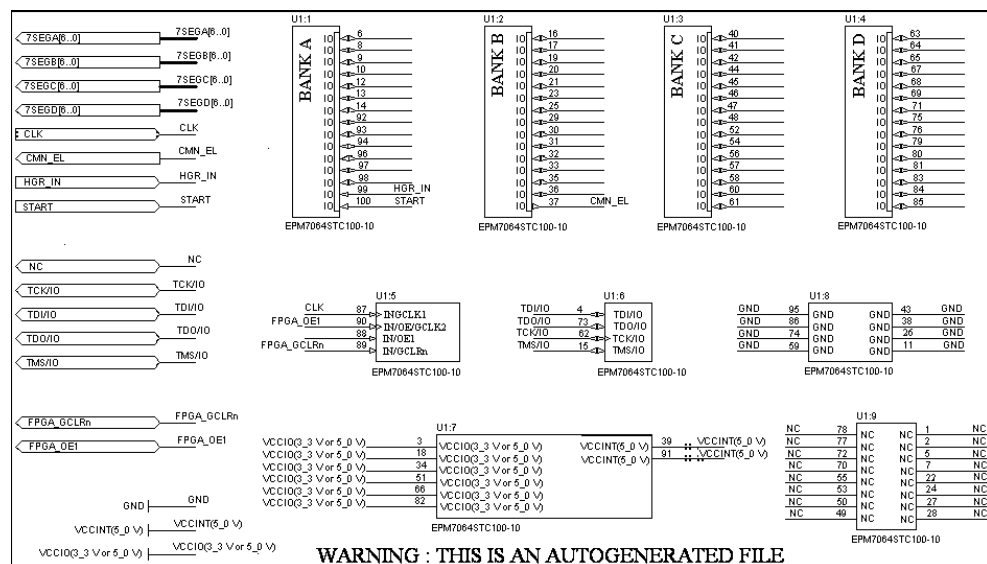


Рис. 8.6. Иерархический схемный компонент ПЛИС

Закончив все настройки, завершить работу мастера объединения проектов щелчком на кнопке **Finish**.

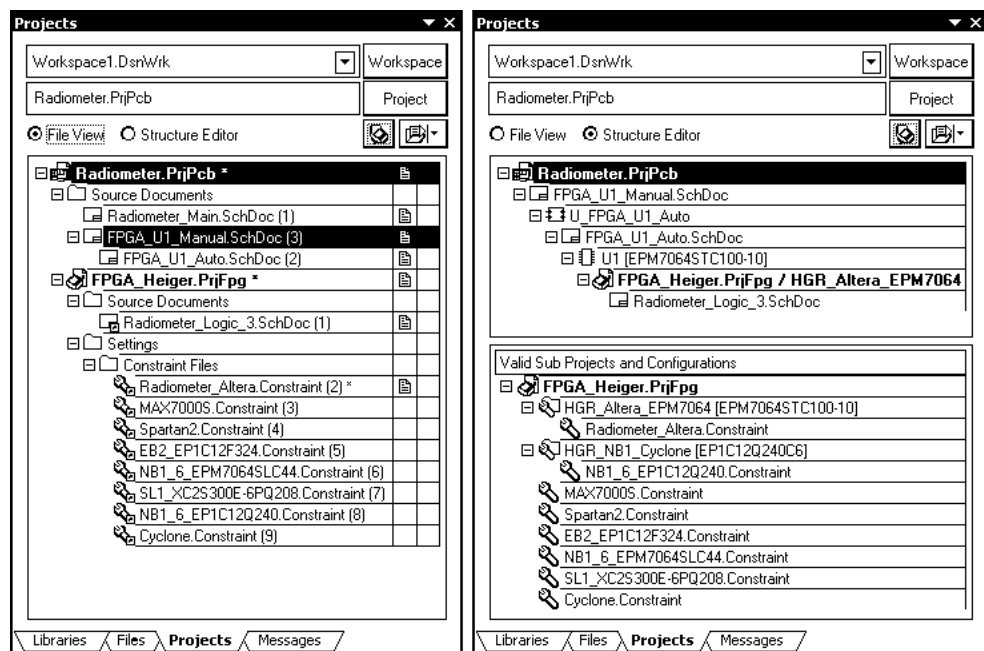
В результате получаем два документа, образующих иерархию:

- ❑ лист схемы с изображением схемного символа ПЛИС, внешних портов логической схемы FPGA-проекта и портов питания (рис. 8.6);
- ❑ лист схемного документа с изображением иерархического схемного компонента, в котором сигнальные порты логической схемы FPGA-проекта заменены на соединители схемных листов — элементы Sheet Connector с теми же именами.

Кроме того, в файл привязок FPGA-проекта добавляется строка:

Record=Constraint | TargetKind=PCB | TargetId=Radiometer_3.PrjPcb

Эту строку следует удалить из файла привязок FPGA-проекта и перенести в файл привязок PCB-проекта.



a

б

Рис. 8.7. Панель **Projects** после объединения FPGA- и PCB-проектов:

a — в режиме просмотра документов проекта;

б — в режиме редактора структуры проектов

Далее щелчком правой кнопки мыши на имени PCB-проекта в плавающей панели **Projects** открыть контекстное меню и выполнить команду компиляции проекта. Созданные схемные документы включаются в иерархическую структуру PCB-проекта (рис. 8.7, а).

Для того чтобы убедиться, осуществилась ли связь PCB- и FPGA-проектов, следует активизировать в плавающей панели **Projects** опцию **Structure Editor**. При успешном объединении в верхнем поле панели FPGA-проект отображается в виде узла иерархической структуры PCB-проекта, а в нижнем поле **Valid Sub Projects and Configurations** (рис. 8.7) отображается структура документов FPGA-проекта.

Не забудьте сохранить в памяти компьютера образованные схемные документы и файл PCB-проекта.

8.2. Синхронизация FPGA- и PCB-проектов

Для физического объединения FPGA- и PCB-проектов необходимо, чтобы к моменту объединения была разработана PCB-часть сквозного проекта, — составлена и откомпилирована электрическая принципиальная схема, включающая иерархический схемный элемент или элементы всех FPGA-проектов, входящих в состав функционального узла, и все схемное окружение ПЛИС, входящее в схему, разработан или открыт ранее сформированный документ редактора печатной платы и на него должны быть "выгружены" компоненты, образующие электрическую схему.

Синхронизация проектов заключается в поддержании соответствия имен цепей и назначения выводов ПЛИС в FPGA- и PCB-проектах и в значительной степени обеспечивается внутренними механизмами DXF платформы. Тем не менее разработчику проекта необходимо понимать действие этих механизмов, чтобы при неумелом вмешательстве не разрушить синхронность проектов, а в некоторых случаях вносить коррективы, призванные восстановить синхронность после редактирования документов выполненного проекта.

Для обнаружения текущего состояния синхронизации проектов следует командой главного меню **Project | FPGA Workspace Map** (Проект | Карта рабочего пространства ПЛИС) открыть одноименную панель **FPGA Workspace Map** (рис. 8.8), в которой отображается состав и состояние синхронизации частей текущего активного сквозного проекта или всех проектов, отображаемых в плавающей панели **Projects**.

- ❑ В крайнем левом поле отображается позиционное обозначение ПЛИС в документе печатной платы РСВ-проекта, имя документа печатной платы текущего открытого проекта, значок и обозначение типа ПЛИС, используемой в проекте.

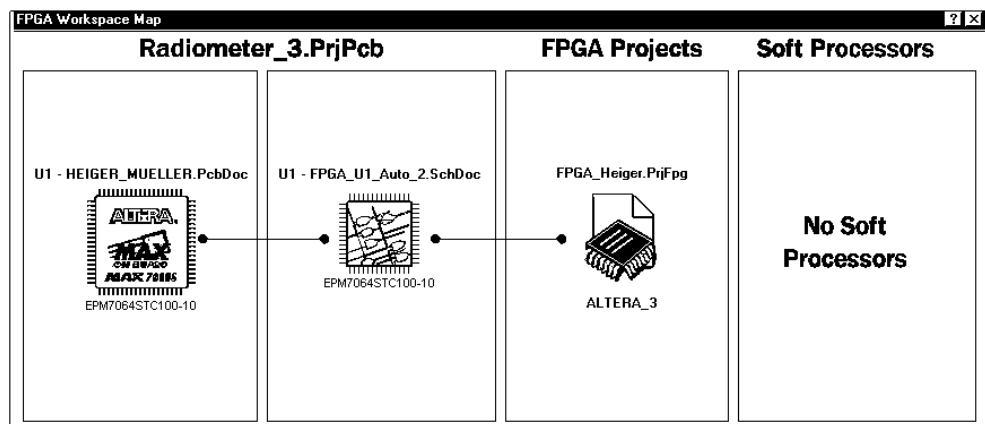


Рис. 8.8. Панель **FPGA Workspace Map** — карты рабочего пространства

- ❑ Во втором слева поле выводится позиционное обозначение ПЛИС в схемном документе РСВ-проекта, имя схемного документа, значок и обозначение типа ПЛИС.

Над двумя этими полями размещается заголовок с именем текущего открытого РСВ-проекта или заголовок **PCB Projects**, если проектов несколько.

- ❑ В следующем поле под заголовком **FPGA Projects** отображается имя текущего открытого FPGA-проекта (FPGA_Heiger.PrjFpg) или всех открытых проектов, значок, обозначающий ПЛИС и имя конфигурации проекта (в нашем примере ALTERA_3).
- ❑ В крайнем правом поле под заголовком **Soft Processors** выдаются сведения о проектах встроенного программного обеспечения.

Текущее состояние связности и синхронизации проектов показывают линии, тянущиеся из поля в поле, — от одного проекта к другому. Если связь существует и синхронность достигнута, цвет линий зеленый, если нет — красный.

Для достижения синхронности следует активизировать диалог щелчком левой кнопки мыши на красной линии связи проектов. Если это линия связи

схемного документа, открывается диалоговое окно синхронизации проектов **Synchronize U1 and FPGA_Heiger.PrjFpg** в заголовке окна на рис. 8.9 отображено позиционное обозначение ПЛИС в схемном документе PCB-проекта и имя FPGA-проекта).

В верхнем поле окна **Matched Signals** размещается таблица связанных сигналов (у которых совпадает имя цепи в FPGA- и PCB-проектах) с обозначением номеров выводов ПЛИС и их электрических типов в FPGA- и PCB-проектах. Ячейки таблицы, в которых эти обозначения совпадают, залиты зеленым фоном, а те, в которых обозначения не совпадают — красным фоном — признак отсутствия синхронизации. В этом случае следует выяснить источник несоответствия и устранить его щелчком мыши на кнопке **Update to PCB** или **Update to FPGA**. Чтобы внесенные изменения вступили в силу, следует выполнить предлагаемую программой стандартную ECO-процедуру.

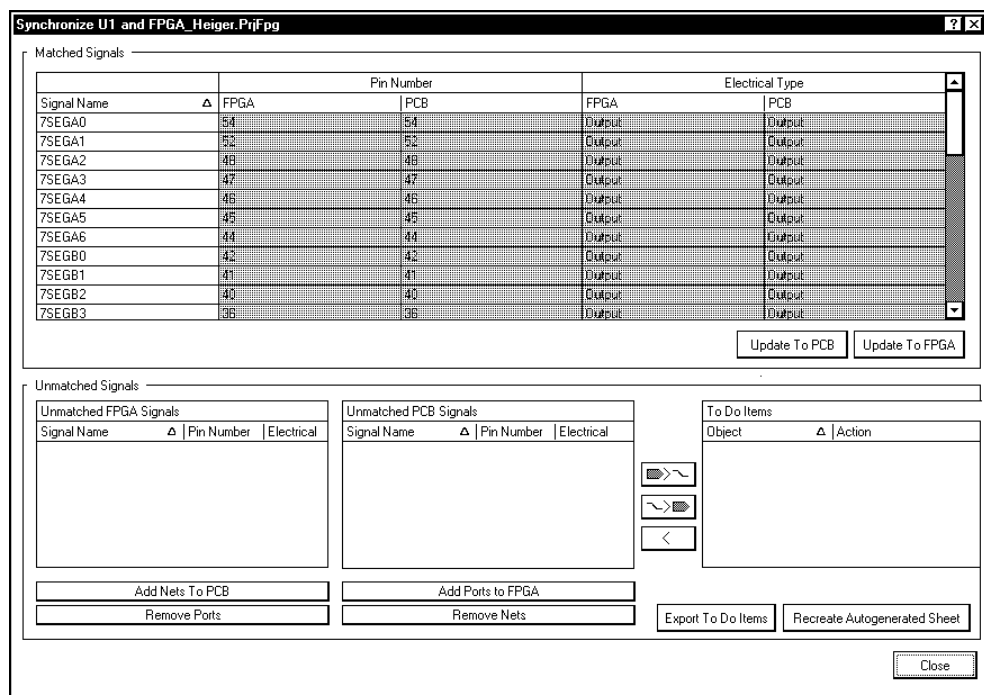


Рис. 8.9. Диалоговое окно синхронизации FPGA- и PCB-проектов

В нижних полях окна (рис. 8.9) отображаются списки несвязанных сигналов (**Unmatched Signals**) — цепей FPGA-проекта, для которых не выполнена

привязка логики ПЛИС к выводам микросхемы (**Unmatched FPGA Signals**) и цепей PCB-проекта, для которых не определены порты в FPGA-проекте. Эти несоответствия устраняют вручную.

Рассмотрим последовательность действий.

1. Кнопками **Add Nets to PCB** или **Add Ports to FPGA** сформировать список имен несвязанных сигналов из одного или другого поля в поле **To-Do Items**.
2. Кнопкой **Export To-Do Items** экспортировать строки этого списка в плавающую панель **To Do**¹; само по себе это не приводит к изменениям в проекте, поскольку таблица в панели **To Do** служит лишь для напоминания разработчику проекта о необходимости выполнить доработки.
3. Кнопкой **Close** закрыть диалог.
4. В открытом схемном документе внести изменения, чтобы обеспечить связность документов FPGA- и PCB-проектов — изменить имена цепей, портов, соединителей листов схемы и т. п.
5. Откомпилировать измененные схемные документы.
6. Выполнить предлагаемую программой ECO-процедуру.
7. Вернуться в оболочку **FPGA Workspace Map** (рис. 8.8) и убедиться, что синхронизация достигнута (по зеленому цвету линий связи проектов).

Кроме рассмотренного способа автоматизированного установления связи и синхронизации FPGA- и PCB-проектов в Altium Designer можно проделать данную работу вручную. Это трудоемкий процесс, включающий практически тот же объем операций, что и при автоматизированном способе, и требующий от разработчика проектов предельного внимания.

8.3. Поддержка связности при модификации проектов

В процессе отладки, подготовки производства, опытной эксплуатации головных образцов изделия PCB-проект может подвергаться модификации различного рода. Это может быть оптимизация трассировки печатного монтажа,

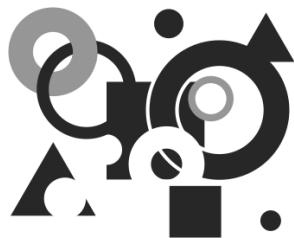
¹ Документ TR0104 Altium Designer Workspace Panels Reference.pdf. Technical Reference TR0104 (v.2.0) June 16, 2006.

требующая переноса электрических связей с одних выводов ПЛИС на другие, изменение электрического типа выводов и технологических стандартов микросхем окружения, соединенных на печатной плате с микросхемой ПЛИС, модификация внутренней логики ПЛИС и связанная с этим необходимость замены одного типа микросхемы на другой и т. п. Altium Designer предоставляет разработчику возможность выполнить любое из упомянутых действий и сохранить при этом синхронизацию FPGA- и PCB-проектов.

К моменту, когда модификация PCB-документов закончена, изменения переданы в FPGA-проект и восстановлена синхронизация проектов, выполненные изменения зафиксированы пока что только в файле привязок проекта. Чтобы перепрограммировать коммутацию внутренних логических связей ПЛИС, следует заново выполнить компиляцию логики, трассировку внутренних соединений и формирование файлов программирования в среде САПР производителя ПЛИС (Vendor Tools).

* * *

В главах 7 и 8 мы рассмотрели лишь ключевые вопросы сквозного проектирования радиоэлектронного функционального узла на основе микросхем программируемой логики (ПЛИС). За рамками изложения остался ряд важных вопросов выполнения проекта на ПЛИС. Прежде всего, это представление логической структуры функционального узла в конструкциях HDL-языков описания аппаратуры, позволяющее добиться более эффективного использования физических ресурсов кристалла ПЛИС при размещении логики проекта [4]. Очень кратко описана стадия обработки проекта в среде специализированных САПР производителя ПЛИС (Vendor Tools). В самых общих словах удалось коснуться вопросов отладки логики проекта на штатной отладочной панели NanoBoard NB-1 и программирования ПЛИС в аппаратуре пользователя.



Схемотехническое моделирование

При традиционных "ручных" способах ведения проекта неизбежна стадия физического моделирования (макетирования), позволяющая выявить ошибки и выполнить настройку действующего макета функционального узла перед передачей его в производство. Современные САПР дают возможность, если не отказаться от этого этапа разработки вовсе, то, во всяком случае, существенно ускорить и сократить затраты на верификацию проекта за счет развитых подсистем моделирования функций, заложенных в проект разработчиком, а также оценки паразитных эффектов в схеме проектируемого изделия.

Подсистема схемотехнического моделирования аналого-цифровых устройств Altium Designer унаследована от САПР узлов РЭС на печатных платах Protel 99 SE и носит название Mixed Signal Circuit Simulator (MSCS), раскрывающее функциональные возможности — моделирование смешанных (аналого-цифровых) цепей передачи сигналов.

Моделирование основано на широко распространенной SPICE-технологии. SPICE — Simulation Program [with] Integrated Circuits Emphasis — "Программа моделирования с акцентом на интегральные цепи (микросхемы)". Ее применение началось в 70-е годы XX в. на больших ЭВМ IBM-360/370. В настоящее время SPICE-технология схемотехнического моделирования реализована в ряде широко распространенных САПР: специализированных, таких как Electronics Workbench, Micro-CAP, MultiSim, и интегрированных — OrCAD (под именем OrCAD PSpice), Protel 99, P-CAD 2000/2002, Protel DXP и др. Полное описание всех деталей SPICE-технологии достаточно объемно [5], [6] и др., поэтому мы ограничимся здесь рассмотрением на нескольких примерах особенностей конкретной реализации ее в Altium Designer и сравнением функций моделирования с функциями систем MicroSim Design Center (OrCAD Pspice), Micro-CAP и Protel 99 (P-CAD 2000–2006).

9.1. Моделирование аналоговых функциональных узлов

9.1.1. Язык моделирования

В подсистеме моделирования Altium Designer (MSCS) электрическая принципиальная схема проектируемого узла описывается на расширенной версии языка Berkeley SPICE3f5/XSpice, содержащего конструкции, известные по системе моделирования PSpice (хотя и не полностью тождественные им), а также предназначенные для описания моделей цифровых компонентов. Язык описания цифровых компонентов Digital SimCode был разработан для системы Protel 99 SE и составляет ее характерную особенность. Формат данных, выполненных в конструкциях языка Digital SimCode, несовместим с другими системами моделирования. На этом языке составляют описания "поведенческих" (behavioral), управляемых событиями (event-driven) моделей цифровых компонентов. Такая структура языка позволяет выполнять моделирование смешанных аналого-цифровых устройств в едином задании, не вставляя вручную аналого-цифровые конверторы на стыке аналоговой и цифровой частей, и не перегружает вычислительные ресурсы компьютера.

Использование поведенческих моделей дает возможность не ограничивать в подсистеме MSCS сложность схемы и глубину вложения как аналоговых, так и цифровых компонентов. Возможно моделирование многолистовых схем. Единственное ограничение — объем оперативной памяти компьютера.

9.1.2. Модели компонентов

Для обеспечения возможности схемотехнического моделирования в библиотеках компонентной базы интегрированных САПР устанавливается связь CAD-описаний компонентов с их SPICE/XSpice-моделями. Принятая в Altium Designer структура библиотек компонентной базы предполагает наличие моделей четырех видов. Модели компонентов представляют собой текстовые или скомпилированные двоичные файлы, описывающие действие одного или группы однородных компонентов. Последнее относится к сериям цифровых интегральных микросхем, объединенных общей технологией изготовления и имеющих сходную структуру и значения электрических параметров.

Файлы моделей имеют предопределенные расширения:

- mdl — модели дискретных аналоговых компонентов;
- ckt — макромодели аналоговых интегральных микросхем;
- txt — модели цифровых компонентов на языке Digital SimCode;
- scb — компилированные наборы цифровых моделей.

В структуре библиотек Altium Designer основная нагрузка по обеспечению связи CAD-образа и моделей компонента ложится на библиотеку схемных образов. Связь CAD-описаний компонентов с их SPICE- либо Digital SimCode-моделями устанавливается при формировании или редактировании библиотек схемных образов компонентной базы *.SchLib.

Процесс подключения моделей к схемному образу (Schematic Symbol) компонента рассматривался в *разд. 2.4*. Напомним, что к моделям относится несколько различных представлений компонента — SPICE- или SimCode-модель для схемотехнического моделирования, топологическое посадочное место (Footprint), трехмерное графическое изображение (PCB 3D-model), Signal Integrity-модель для оценки взаимных наводок и искажений импульсных сигналов при передаче их по линиям печатного монтажа.

Схемные символы с присоединенными к ним SPICE-моделями готовы для использования в схемотехническом моделировании. Для обеспечения сквозного проектирования и переносимости библиотек с компьютера на компьютер схемные символы с присоединенными моделями компилируются в интегрированную библиотеку.

В поставку САПР Altium Designer входит более 160 интегрированных библиотек, содержащих несколько тысяч библиотечных образов компонентной базы, поддержанных SPICE/XSpice- или Digital SimCode-моделями¹. Все это компоненты, разработанные и поставляемые на рынок иностранными производителями.

К сожалению, отечественная компонентная база значительно беднее той, которая поставляется по импорту. Кроме того, среди существующих библиотек отечественной компонентной базы за редчайшими исключениями отсутствуют библиотеки, связанные со SPICE-моделями или моделями других видов, которые позволяли бы выполнить схемотехническое моделирование разрабатываемых функциональных узлов.

¹ Документ Simulation Models and Analyses reference. Technical Reference TR0113 (v. 1.4) June 09, 2006.

При необходимости можно самостоятельно разработать библиотеки отечественной компонентной базы, связанные со схемотехническими моделями. Для этого существуют три способа:

1. Написать тексты моделей в синтаксисе входного языка PSpice, описанного в литературе ([5] и др.), или языка или Digital SimCode и установить связь этих моделей с CAD-описаниями библиотек отечественных компонентов.
2. В среде Altium Designer воспользоваться функциями мастера **SPICE Model Wizard**¹, позволяющего в полуавтоматическом режиме сформировать SPICE-модели на основании справочных технических характеристик компонентов или результатов измерений соответствующих параметров на реальных компонентах. Аналогичными средствами формирования моделей располагают и другие известные САПР.
3. Воспользоваться тем фактом, что большинство отечественных электронных компонентов — диодов, транзисторов и, в особенности, интегральных микросхем — аналоги импортных компонентов. Пользуясь справочниками аналогов отечественной и импортной компонентной базы, найти в библиотеках, поставляемых с конкретной САПР, модели подходящих аналогов и подключить их к CAD-образам соответствующих отечественных компонентов.

В последнем случае нужно быть особенно внимательным. Необходимо следить, чтобы порядок следования (нумерация) электрических выводов в CAD-описаниях схемных символов совпадал с порядком обозначения этих выводов в текстах модельных файлов. Если этого не будет, модель формально присоединится к схемному компоненту, но программа моделирования обнаружит несоответствия, и исполнение задачи моделирования остановится по фатальной ошибке.

Приведем пример схемного компонента отечественного быстродействующего операционного усилителя KP140УД11 из пользовательской библиотеки Simlib.SchLib с подключенной SPICE-моделью зарубежного аналога — операционного усилителя LM318S8 фирмы Linear technology. Компоненты из этой библиотеки будут также использованы в приводимых далее примерах выполнения задач схемотехнического моделирования.

Для просмотра интересующих нас сведений о модели компонента необходимо в открытом схемном документе Altium Designer указать курсором на схемный символ и двойным щелчком левой кнопки мыши вызвать диалог

¹ Документ Spice Model Creation from User Data. Application Note AP0141 (v. 1.0) April 06, 2006.

управления свойствами схемного компонента (см. рис. 4.4). После этого двойным щелчком на имени присоединенной SPICE-модели в поле **Models for DAnn KR140UD11** открыть диалоговое окно **Sim Model — General / Spice Subcircuit** (рис. 9.1).

Диалоговое окно имеет три вкладки в верхнем обрамлении и три — в нижнем. На нижней вкладке **Netlist Template** размещается текстовая строка-шаблон, поля которой соответствуют формату записи обозначения компонента в текстовом файле задания на моделирование:

```
@DESIGNATOR %1 %2 %3 %4 %5 @MODEL
```

На нижней вкладке **Netlist Preview** (Предварительный просмотр в списке цепей) размещается текстовая строка, отображающая реальную строку описания компонента в текстовом файле задания на моделирование, составленную по приведенному шаблону:

```
XDA4 <3> <2> <7> <4> <6> LM318S8
```

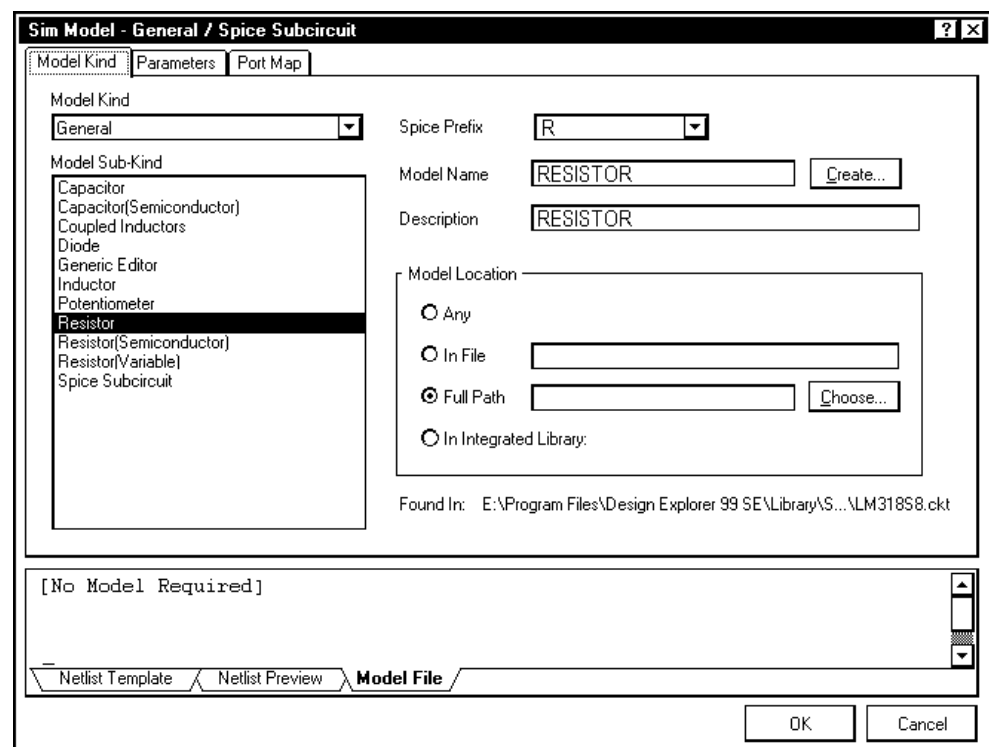


Рис. 9.1. Диалог присвоения SPICE-модели

В поле XDA4 позиционному обозначению компонента в схеме (DA4) предшествует буква X, служащая обозначением типа SPICE-модели компонента — макромодель аналоговой интегральной микросхемы. Далее в угловые скобки заключены обозначения сигнальных выводов и выводов питания микросхемы. Завершается строка именем модели операционного усилителя LM318S8 — зарубежного аналога отечественной микросхемы КР140УД11.

На вкладке **Model File** размещается текст файла-модели (рис. 9.2). В соответствии со стандартом, принятым в SPICE-технологии, текст описания модели начинается с заголовка, из которого ясен порядок следования обозначений выводов микросхемы в строке текстового файла задания: вход неинвертирующий, вход инвертирующий, питание (плюс), питание (минус), выход. Строкам заголовка предшествует символ *, служащий признаком строки-комментария, не подлежащего обработке программой. Далее следует тело модели, текст которого начинается директивой обращения к макромодели, SUBCKT и заканчивается директивой завершения описания .ENDS (листинг 9.1).

Листинг 9.1

```

*////////////////////////////////////
* LM318S8 High Speed Operational Amplifier //
*////////////////////////////////////
*
* connections:  non-inverting input
*               |  inverting input
*               | |  positive power supply
*               | | |  negative power supply
*               | | | |  output
*               | | | | |
*               | | | | |
.SUBCKT LM318S8 3 2 7 4 6
.
.
.
.
.
.ENDS LM318

```

На верхней вкладке **Model Kind** (Тип модели) производится выбор типа модели, поиск и присоединение нужной модели к схемному образу компонента.

На вкладке **Port Map** (Карта портов) отображается условно-графическое обозначение (УГО) схемного символа и таблица соответствия обозначений электрических выводов в библиотеке схемных компонентов и в файле модели (рис. 9.2).

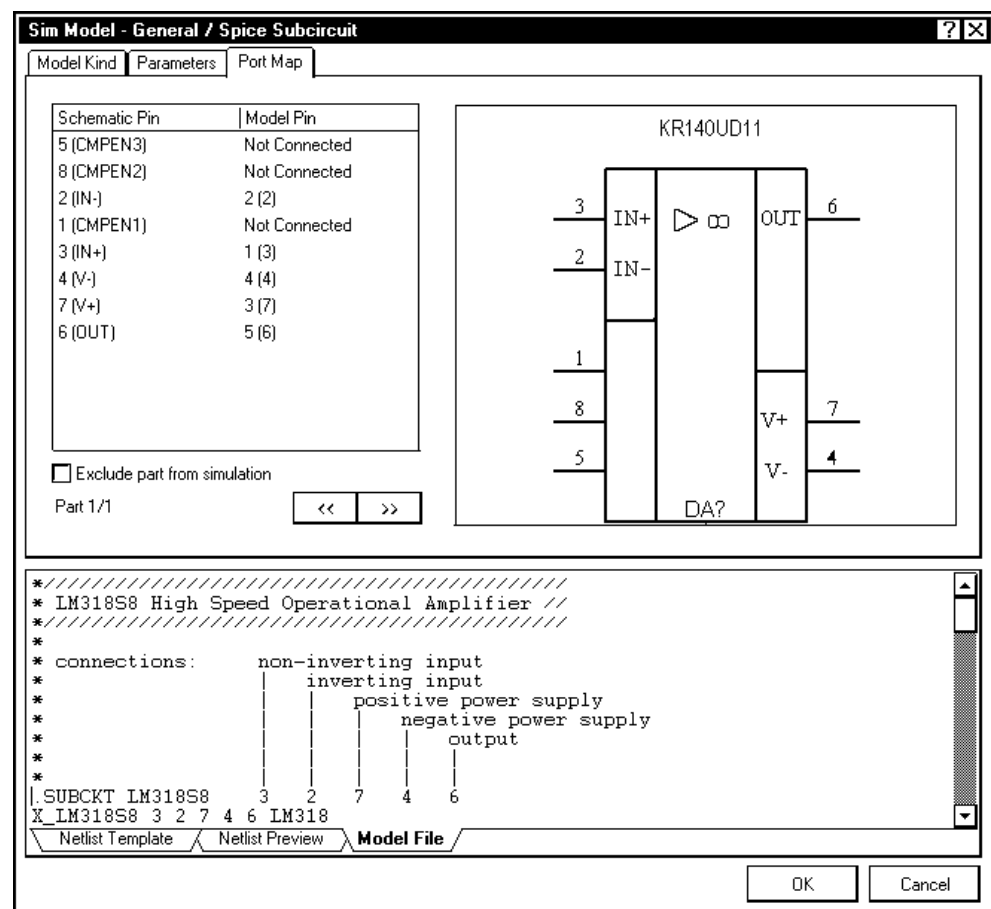


Рис. 9.2. Диалог выбора типа SPICE-модели

Пассивные компоненты (резисторы, конденсаторы, катушки индуктивности) также могут поддерживаться соответствующими SPICE-моделями, определяющими нелинейные свойства таких компонентов, зависимость их основного электрического параметра от температуры и т. п.

Приведем пример SPICE-модели резистора, сопротивление которого зависит от температуры:

```
.model R_5_50 RES (R=1 DEV/GAUSS 5% TC1=50e-6 TC2=5e-6)
```

Расшифруем смысл полей приведенной строки описания модели:

- ☐ `.model` — директива PSpice, служащая для включения модели в текст задания на моделирование;
- ☐ `R_5_50` — произвольное имя модели;
- ☐ `RES` — обязательное имя типа компонента (резистор);
- ☐ `R` — масштабный коэффициент, используемый вычислительным ядром PSpice (по умолчанию $R=1$);
- ☐ `DEV/GAUSS 5%` — вид распределения отклонений значения сопротивления от номинала (гауссов закон с предельным отклонением $\pm 5\%$);
- ☐ `TC1=50e-6` — линейный ТКС $= 50 \cdot 10^{-6}$;
- ☐ `TC2=5e-6` — квадратичный ТКС $= 5 \cdot 10^{-6}$.

Назначение связи с файлом модели для пассивных компонентов не обязательно. Программа использует для них, а также для целого ряда других компонентов (источников напряжений и токов и др.) внутренние встроенные модели, недоступные для редактирования пользователем. Для таких компонентов предопределен или должен быть назначен на вкладке **Model Kind** тип модели **Model Kind** и подтип **Model Sub-Kind**.

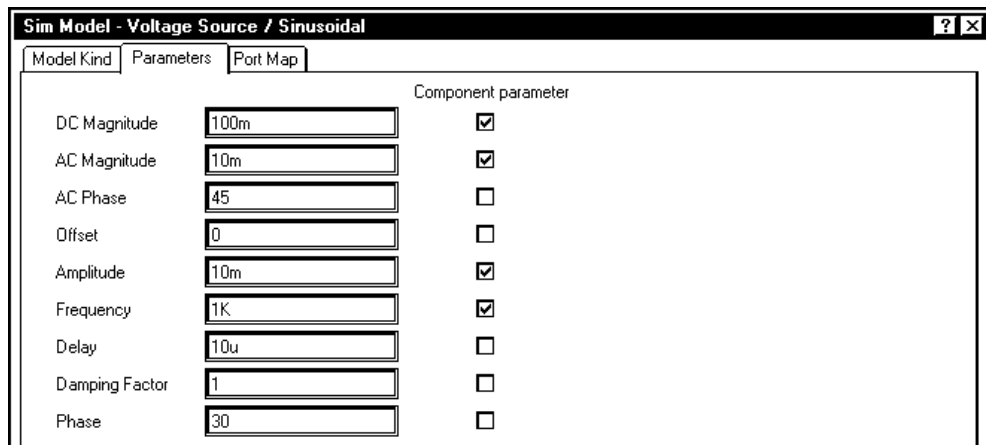


Рис. 9.3. Диалог назначения параметров SPICE-модели

На вкладке **Parameters** размещается список параметров встроенной математической модели, значения которых должны быть указаны задаче моделирования (рис. 9.3).

Программа автоматически присваивает будущему обозначению компонента в задаче моделирования префикс, служащий для распознавания программой вида компонента и запуска соответствующей подпрограммы вычислительно-го ядра для моделирования поведения данного компонента в работающей схеме, а также предопределенное имя модели. Приведем список наиболее часто встречающихся в задачах моделирования префиксов (табл. 9.1).

Таблица 9.1. SPICE-префиксы моделей компонентов

Префикс	Вид компонента	Порядок обозначения выводов
A	Макромодель цифрового компонента на языке Digital SimCode	В порядке следования в файле модели *.txt или *.scb
C	Конденсатор	(+)узел — (–)узел
D	Диод	Анод — катод
E	Источник напряжения, управляемый напряжением	(+)узел — (–)узел
G	Источник тока, управляемый напряжением	То же
I	Независимый источник тока	То же
K	Взаимная индуктивность	L1 — L2 – <коэфф. связи>
L	Индуктивность	(+)узел — (–)узел
Q	Биполярный транзистор	Коллектор — база — эмиттер
R	Резистор	(+)узел — (–)узел
V	Независимый источник напряжения	То же
X	Макромодель аналоговой микросхемы	В порядке перечисления в директиве .SUBCKT

В правой колонке табл. 9.1 приведен порядок обозначения выводов компонентов в тексте задания на моделирование. Полный перечень SPICE-префиксов и предопределенных имен моделей приведен в [5], [6] и в упоминавшемся ранее документе *Simulation Models and Analyses reference. Technical Reference TR0113 (v.1.4) June 09, 2006* и составляет несколько десятков позиций. Предоставляем читателю самостоятельно ознакомиться с этим списком. В нашем же изложении достаточно обозначений, приведенных в табл. 9.1.

9.1.3. Подготовка схемы к моделированию

Электрическая принципиальная схема, подлежащая моделированию, формируется по обычным правилам, рассмотренным в *разд. 4.1*. Компоненты, образующие схему, следует выбирать из библиотек, в которых установлена связь схемных символов с соответствующими SPICE-моделями. Отличие от схем, не подлежащих моделированию, состоит только в том, что в схему на равных правах с другими компонентами необходимо включить источники питания и входных сигналов из библиотеки ...\\Altium Designer\\Library\\Simulation\\Simulation Sources.IntLib.

9.1.4. Задание на моделирование

Составление и выполнение задачи моделирования можно активизировать из открытого схемного документа командой главного меню программы **Design | Simulate | Mixed Sim** (Разработка | Моделировать | Смешанное моделирование), однако для лучшей управляемости процесса целесообразно командой главного меню **View | Toolbars | Mixed Sim** (Вид | Линейки инструментов | Смешанное моделирование) вызвать и зафиксировать в линейке инструментов специальное кнопочное меню **Mixed Sim** с тремя кнопками-командами управления (рис. 9.4).

В открытом схемном документе щелчком на правой кнопке **Generate XSpice Netlist** (генерировать XSpice-описание схемы) запускаем процедуру формирования описания схемы на входном языке системы моделирования.

Отметим, что системы моделирования, использующие SPICE-технологии, не русифицированы — в обозначениях компонентов и их параметров, других текстовых строках файлов задач моделирования присутствуют только латинские буквы.

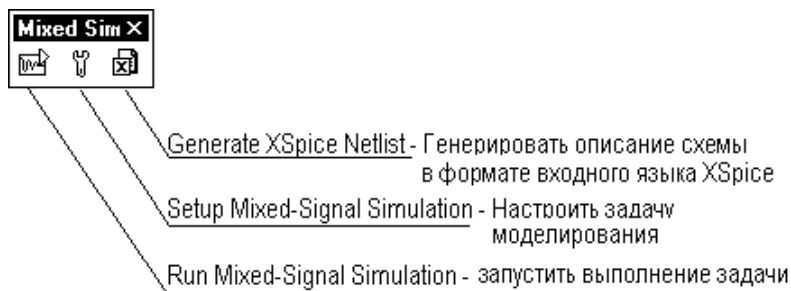


Рис. 9.4. Меню команд управления моделированием

Описание схемы на входном языке системы моделирования представляет собой текстовый файл, содержащий описание компонентов и узлов цепи, к которым подключены выводы компонентов.

Описания компонентов имеют одинаковую структуру для электрорадиокомпонентов цепи, источников входных сигналов и источников электропитания. Описание компонента имеет структуру:

```
<имя компонента><имена узлов подключения>
+ [ <имя модели> ] <числовые значения параметров>
```

Знак "плюс" с необязательными последующими пробелами в начале строки означает перенос описания компонента с предыдущей строки.

Программа не различает строчные и прописные буквы.

Узлу "Общий" ("земля") обязательно должно быть присвоено имя 0 (ноль).

Имя компонента должно начинаться с SPICE-префикса, определяющего вид компонента. За префиксом обычно следует позиционное обозначение компонента по принципиальной схеме. Если буква-префикс совпадает с первой буквой в позиционном обозначении, она не дублируется в обозначении компонента.

Имена узлов цепи программа считывает из схемы. Это могут быть как метки цепей **Net Label**, так и имена, назначенные схемным редактором по умолчанию. В именах допустимы латинские буквы A...Z, цифры от 0 до 9 и знаки \$, _, *, /, %.

После обозначения имени и узлов подключения указываются числовые значения параметров компонентов. Обозначение единиц измерения электрических величин не обязательно. По умолчанию программа принимает основную величину: В, А, Ом и т. д.

Принятые в технике приставки к основной единице измерения физической величины, обозначающие кратность тысяче, миллиону и т. д. в лексике языка моделирования PSpice или Xspice, имеют вид суффиксов, добавляемых к числовому значению параметра компонента. Суффиксы масштаба обязательно записывать без пробела справа от цифр. Приведем список этих масштабных суффиксов (табл. 9.2).

Таблица 9.2. Размерные суффиксы в языке SPICE3f5/XSPICE

Кратность	Обозначение	SPICE-суффикс
10^{-15}	фемто	f
10^{-12}	пико	p
10^{-9}	нано	n
10^{-6}	микро	u
10^{-3}	милли	m
10^3	кило	k
10^6	мега	meg
10^9	гига	g
10^{12}	тера	t

Отдельного комментария заслуживает приставка "микро". Вместо греческой литеры μ , принятой в мировой инженерной практике для обозначения микро-величин, выбрана ближайшая по начертанию (хотя и без "хвостика") латинская литера u.

Числовую величину можно также указывать в формате с фиксированной или плавающей точкой: f=1e-15, p=1e-12, n=1e-9, u=1e-6, m=1e-3, k=1e3, meg=1e6, g=1e9, t=1e12. Для наглядности допускается прибавлять к этим префиксам обозначения основной единицы в любом написании: kOm, megOhm, gHz и т. п. Для моделирования эти добавленные символы значения не имеют: программа опознает компоненты по SPICE-префиксам (R, C и т. д.), определяет физическую природу и единицы измерения моделируемой величины и использует ее в соответствующих расчетных модулях.

За описаниями компонентов и связей схемы в тексте задания следуют директивы моделирования, определяющие виды анализа цепи и различные способы обработки и представления результатов. Они включаются в текст задачи

моделирования при выборе вида моделирования и настройке режимов выбранного вида анализа. Будем рассматривать состав и действие этих директив на примерах назначения и настройки основных видов моделирования.

9.1.5. Анализ цепи по постоянному току

Построим задачу расчета коллекторных вольт-амперных характеристик отечественного транзистора КТ315А. Электрическая схема для определения вольт-амперных характеристик транзистора приведена на рис. 9.5.

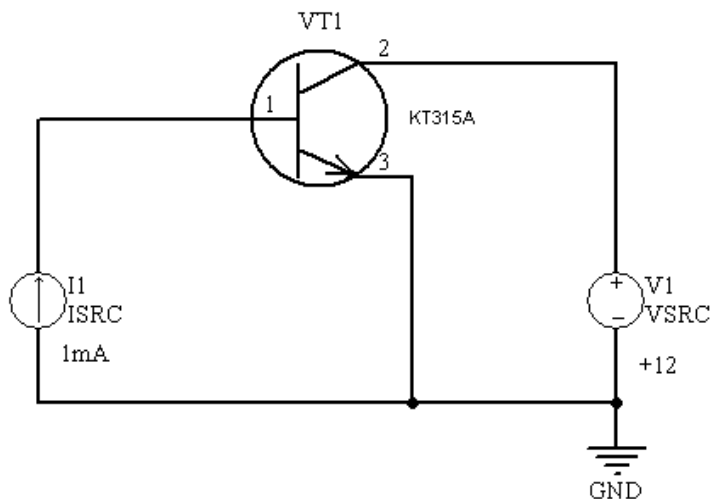


Рис. 9.5. Схема для моделирования вольт-амперных характеристик транзистора

Кроме исследуемого транзистора схема содержит два независимых источника постоянного тока — источник коллекторного напряжения V1 типа VSRC и источник базового тока I1 типа ISRC.

В процессе выполнения задачи попытаемся построить семейство вольт-амперных характеристик при вариации коллекторного напряжения и ступенчатом изменении базового тока.

Составим задание на моделирование. Для этого необходимо:

1. Щелчком на средней кнопке меню **Mixed Sim** активизировать диалог настройки режимов моделирования. Откроется окно настроек анализа **Analyses Setup** (рис. 9.6).

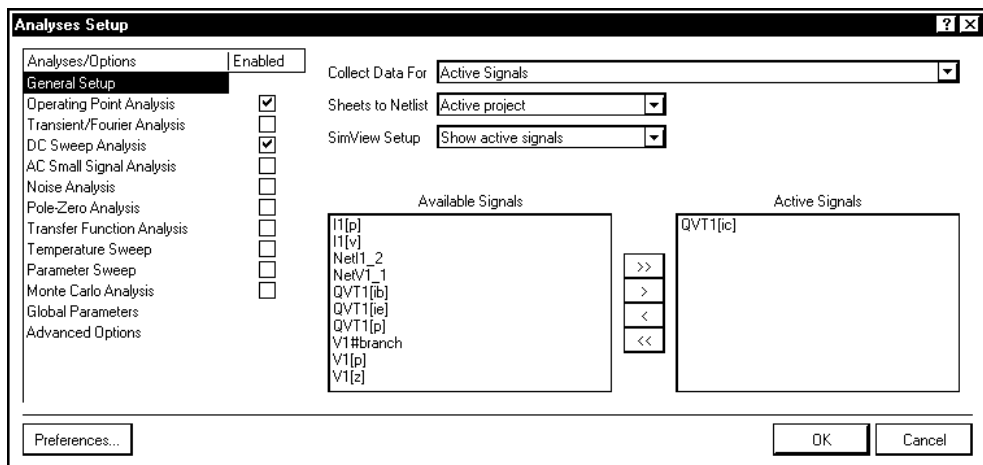


Рис. 9.6. Окно настройки режимов моделирования: общие настройки

2. Указать курсором в списке **Analyses/Options** в левой части окна опцию **General Setup** и назначить настройки, общие для всех видов анализа, которые должны быть выполнены в рамках текущей задачи:
 - в поле **Sheets to Netlist** кнопкой выбора развернуть список из двух возможных вариантов выбора документа, включаемого в задачу моделирования — активный проект (**Active project**) или активный схемный документ (**Active sheet**). Если активный проект содержит несколько схемных документов, программа соберет в один файл задания данные из всех схем проекта. В нашем простом примере достаточен выбор схемного документа;
 - в поле **Collect Data For** определить набор данных, представляющих интерес при моделировании. Возможен выбор одного из пяти вариантов:
 - ◇ **Node Voltage, Supply Current, Device Current and Power** — напряжения в узлах, токи питания, токи в компонентах и мощности;
 - ◇ **Node Voltage, Supply Current and Subcircuit VARs** — напряжения в узлах, токи питания и переменные макромоделей;
 - ◇ **Node Voltage, Supply and Device Current** — напряжения в узлах, токи питания и токи в компонентах;
 - ◇ **Node Voltage and Supply Current** — напряжения в узлах и токи источников питания;
 - ◇ **Active Signals** — активные сигналы.

При моделировании сложных схем далеко не всегда представляет интерес полный состав узлов, ветвей и компонентов схемы, в которых рассчитываются напряжения, токи, мощности и импедансы. В нашей задаче достаточно рассчитать значения тока коллектора транзистора, поэтому назовем опцию **Active Signals** и, выбрав в списке доступных сигналов **Available Signals** обозначение QVT1[ic], щелчком на кнопке "стрелка вправо" перенесем это обозначение в список активных сигналов **Active Signals** (см. рис. 9.6).

- В поле **SimView Setup** выбрать один из двух возможных вариантов представления результатов моделирования:
 - ◇ **Show active signals** — показывать активные сигналы;
 - ◇ **Keep last setup** — сохранить предшествующие настройки.

Первый вариант предпочтителен при однократном выполнении задачи моделирования. Второй вариант целесообразно выбирать при многократном выполнении одной и той же задачи. Он аналогичен используемой в широко известной системе моделирования OrCAD PSpice настройке **Display Control | Last Session | Restore**.

3. Указать в списке **Analyses/Options** окна (см. рис. 9.6) анализ по постоянному току — **DC Sweep Analysis** и активизировать данную опцию "птичкой" в поле **Enabled**. В правой половине окна откроется набор опций настройки выбранного режима анализа (рис. 9.7).

Программа позволяет выполнить анализ при изменении двух параметров элементов схемы, представленной на рис. 9.5.

В главном цикле (**Primary Sweep**) производится вариация напряжения источника коллекторного питания V1. Установить следующие значения параметров главного цикла:

- **Primary Source** — выбрать источник, параметр которого варьируется в главном цикле: в нашем случае это источник V1;
- **Primary Start** — начальное значение напряжения 0.000;
- **Primary Stop** — конечное значение напряжения 10.00 (10 В);
- **Primary Step** — значение шага вариации 250.0m (250 мВ).

Установкой "птички" в строке **Enable Secondary** активизировать вложенный цикл вариации тока базы. В нижележащих полях диалогового окна установить значения параметров вложенного цикла:

- **Secondary Name** — имя переменной, варьируемой во вложенном цикле: в нашей задаче это источник тока базы I1;
- **Secondary Start** — начальное значение тока базы 0.000;
- **Secondary Stop** — конечное значение тока базы 1.000m (1 мА);
- **Secondary Step** — значение шага вариации 50.00u (50 мкА).

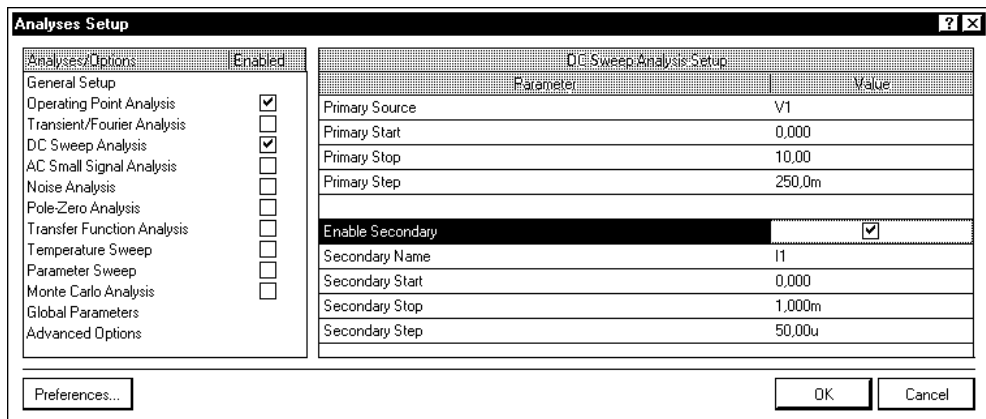


Рис. 9.7. Окно настройки режимов моделирования: анализ по постоянному току — DC Sweep Analysis

- Щелчком на кнопке **ОК** завершить настройки. Программа формирует текстовый файл задания на моделирование Transistor_VA_Char.nsx (листинг 9.2).

Листинг 9.2

Transistor_VA_Char

**SPICE Netlist generated by Advanced Sim server on 17.01.2009 0:44:12*

**Schematic Netlist:*

I1 0 BASE 1mA

V1 COLLECTOR 0 +12

QVT1 COLLECTOR BASE 0 KT315A

.SAVE @QVT1[ic]

```

*PLOT DC -1 1 A=@QVT1[ic]
*PLOT OP -1 1 A=@QVT1[ic]

.OPTION KeepLastSetup=False
*Selected Circuit Analyses:
.DC V5 0 10 0.1 I1 0 0.0005 5E-5
.OP

*Models and Subcircuit:
.model KT315a NPN(Is=21.11f Xti=3 Eg=1.11 Vaf=115 Bf=79.74
+ Ise=233.2f Ne=1.417 Ikf=.2922 Xtb=1.5 Br=1.3 Isc=107.3f
+ Nc=1.298 Ikr=2.561 Rb=12 Rc=1.032 Cjc=8.988p Mjc=.33 Vjc=.75
+ Fc=.5 Cje=18.5p Mje=.33 Vje=.75 +Tr=244.3n Tf=321.4pItf=1
+ Xtf=2 Vtf=60)

.END

```

Директивы, включенные в текст задачи, определяют виды и параметры режимов анализа и ряд дополнительных функций, выполняемых в ходе моделирования. Запись директивы начинается с новой строки и с символа "точка":

- ☐ .OP — расчет схемы по постоянному току в точке покоя, по умолчанию включаемая программой во все задания;
- ☐ .DC — анализ схемы в условиях вариации напряжения или тока источников постоянного тока;
- ☐ .SAVE — директива, определяющая, какие данные должны быть сохранены и использованы при отображении результатов моделирования;
- ☐ .MODEL — директива обращения к SPICE-модели компонента.

Всего в задачи моделирования, выполняемые на основе SPICE-технологии, система моделирования может включать порядка 30 различных директив. Состав директив каждый раз определяется спецификой задачи и настройками режимов анализа.

Выполнив настройку, нужно активизировать в меню **Mixed Sim** команду-кнопку **Run Mixed-Signal Simulation**. Программа выполняет назначенные виды анализа, выводит результаты в файл Transistor_VA_Char.sdf и отображает их в виде графиков в главном окне Altium Designer (рис. 9.8).

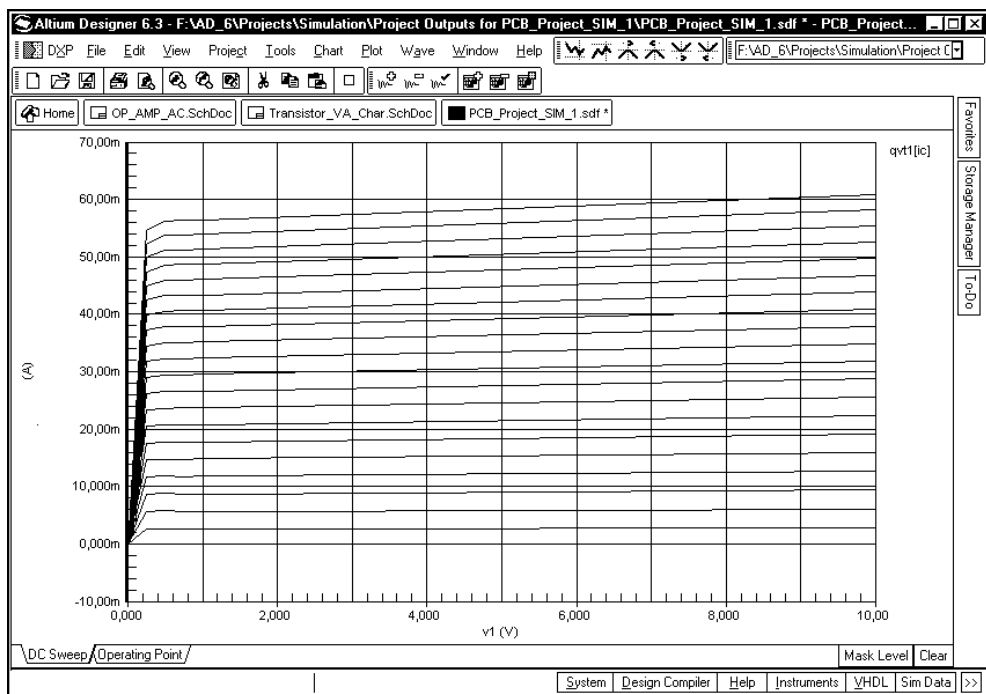


Рис. 9.8. Результат моделирования
вольт-амперных характеристик транзистора

В нашем примере результаты отображаются в главном окне на двух вкладках — **DC Sweep** и **Operating Point**. На вкладке **DC Sweep** строится семейство искомых коллекторных вольт-амперных характеристик транзистора, а на вкладке **Operating Point** (Рабочая точка) приводится величина тока покоя схемы при нулевых значениях коллекторного напряжения и базового тока.

9.1.6. Анализ частотной характеристики цепи

Для определения частотной характеристики электрической цепи на входе ее задается воздействие в виде гармонического сигнала постоянной амплитуды, частота которого изменяется в заданном диапазоне. Этот вид анализа в системах SPICE-моделирования обычно называется **AC Sweep**.

В Altium Designer подобный анализ назван еще и малосигнальным (**AC Small Signal Analysis**), хотя здесь, как и в других системах SPICE-моделирования [6], [7], программа при выполнении задания линеаризует исследуемую цепь,

не налагает ограничений на амплитуду входного сигнала и игнорирует тот факт, что в таких активных цепях, как транзисторный усилительный каскад, операционный усилитель и т. п. выходной сигнал не может превышать напряжение питания схемы.

Рассмотрим этот вид анализа на примере исследования частотной характеристики масштабирующего каскада, выполненного на основе операционного усилителя, охваченного отрицательной обратной связью (рис. 9.9).

На данном примере можно проследить, как глубина обратной связи влияет на значение коэффициента усиления и ширину полосы пропускания такой схемы.

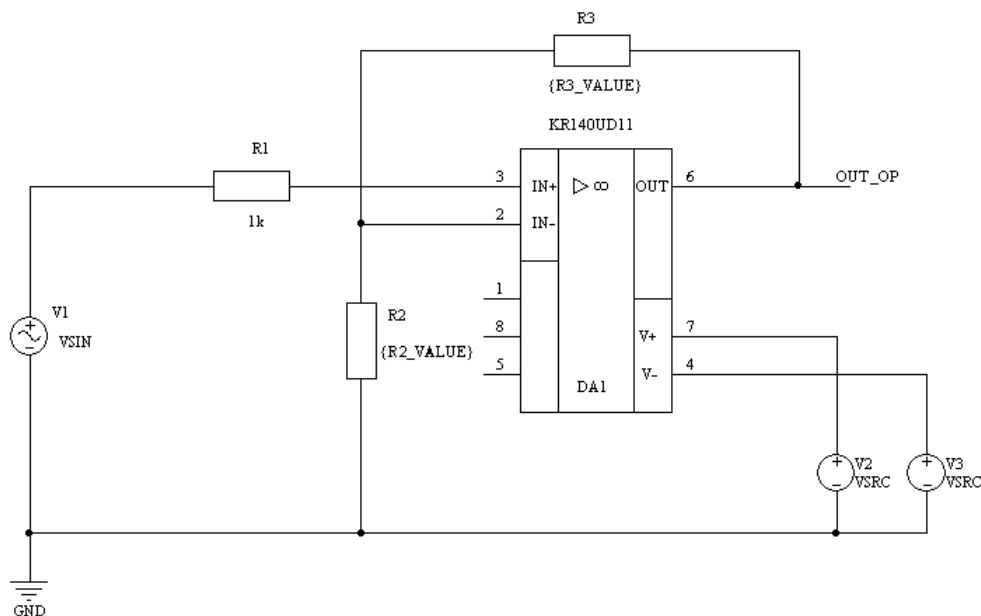


Рис. 9.9. Схема масштабирующего усилителя

Известно, что максимум коэффициента усиления собственно ОУ находится в полосе частот от нуля до некоторой верхней (относительно невысокой) граничной частоты. Причина — высокое входное сопротивление ОУ и наличие внутренних паразитных емкостей, образующих RC-цепи, определяющие уменьшение усиления ОУ с увеличением частоты. Наклон частотной характеристики определяется главным образом первым полюсом его передаточной функции, лежащим обычно на частоте 100...1000 Гц, и составляет приблизительно 20 дБ на декаду. В паспортных данных на операционные усилители

обычно указывается граничная частота, на которой коэффициент усиления ОУ падает до значения 1 (так называемая частота единичного усиления). У используемого в нашем примере усилителя K140УД11 коэффициент усиления по постоянному току составляет 25000, а частота единичного усиления равна 15 МГц. SPICE-модели операционных усилителей отображают это свойство реальных ОУ.

Известно также, что введение отрицательной обратной связи приводит к уменьшению усиления и расширению полосы пропускания усилителя. В нашем примере попытаемся показать, как связаны между собой коэффициент усиления и полоса пропускания конфигурации с отрицательной обратной связью, приведенная на рис. 9.9. Для этого проведем анализ частотной характеристики в условиях вариации параметра схемы, определяющего глубину обратной связи, а именно — сопротивления резистора обратной связи R_3 .

В предположении, что операционный усилитель идеален, т. е. имеет коэффициент усиления $K \rightarrow \infty$, входное сопротивление $R_{вх} \rightarrow \infty$ и выходное сопротивление $R_{вых} \rightarrow 0$, коэффициент усиления схемы рис. 9.9 составляет:

$$K_{оос} = \frac{K}{1 + \beta K} = \frac{K}{1 + KR_2/(R_2 + R_3)} \approx \frac{R_2 + R_3}{R_2} = 1 + \frac{R_3}{R_2}. \quad (9.1)$$

Такая конфигурация называется масштабирующей, поскольку с увеличением сопротивления резистора обратной связи R_3 коэффициент усиления ее приближается к значению R_3/R_2 , которое называют коэффициентом масштаба. Естественно, при равенстве $R_2 = R_3$ коэффициент усиления $K_{оос} = 2$.

Рассмотрим последовательность настройки моделирования.

1. В активном схемном документе Op_AMP_AC.SchDoc (см. рис. 9.9) указать курсором на источник входного сигнала V1, после чего двойным щелчком левой кнопки мыши вызвать диалог настройки параметров компонента и установить значение амплитуды входного сигнала, используемого в режиме анализа частотной характеристики **AC Magnitude 10m** (10 мВ).
2. Щелчком на кнопке **Setup Mixed-Signal Simulation** в меню **Mixed Sim** открыть диалоговое окно настроек режимов анализа и на шаге общих настроек **General Setup** (см. рис. 9.5) назначить:
 - в поле **Sheets to Netlist** указать активный схемный документ (**Active Sheet**);
 - в поле **Collect Data For** указать **Active Signals** (Активные сигналы);
 - в поле **SimView Setup** выбрать вариант **Show active signals** (Показывать активные сигналы);

- в списке узлов и ветвей цепи **Available Signals** указать и перевести в поле активных сигналов **Active Signals** имя узла **OUT_OP**.
3. Выбрать в диалоговом окне (см. рис. 9.6) вид анализа **AC Small Signal Analysis** (Анализ частотной характеристики). Откроется панель настройки режима **AC Small Signal Analysis Setup** (рис. 9.10).
 4. Настроить параметры режима анализа:
 - нижнюю частоту диапазона **Start Frequency** 1.000 (1 Гц);
 - конечную, верхнюю частоту диапазона, до которой выполняется расчет **Stop Frequency** 100.0meg (100 МГц);
 - назначить способ определения точек на оси частот, в которых выполняется расчет — разбивку частотной оси на равные интервалы (**Linear**), разбивку на октавы (**Octave**) или (**Decade**);
 - указать число точек расчета на октаву, на декаду или общее число точек на весь диапазон (при выборе варианта разбивки **Linear**): в нашей задаче укажем 40 точек на декаду.

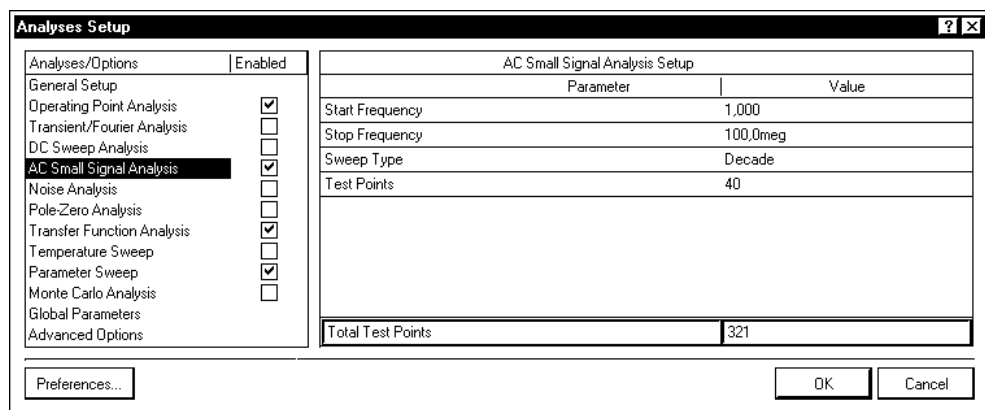


Рис. 9.10. Окно настройки режимов моделирования: анализ частотной характеристики — **AC Small Signal Analysis**

5. На этом настройку можно завершить и дать старт выполнению задачи моделирования. Однако мы собираемся выполнить вариацию сопротивления резистора обратной связи R3. Поэтому перейдем к настройкам, позволяющим провести такой анализ, и только после этого приведем конечный результат. Это будет настройка параметрического анализа.

9.1.7. Параметрический анализ

Система моделирования MSCS позволяет совместить анализ частотной характеристики, характеристики по постоянному току и переходных процессов с вариацией параметров компонентов схемы. Варьировать можно следующие параметры математических моделей компонентов:

- ❑ значения электрического сопротивления, емкости, индуктивности, коэффициента связи в модели связанных катушек индуктивности;
- ❑ значение напряжения и тока независимых источников;
- ❑ коэффициент прямой передачи базового тока транзисторов BF;
- ❑ время задержки распространения сигнала в логических схемах TP.

Варьируемые параметры могут входить в качестве аргументов в ряд математических функций:

- ❑ $ABS(x)$ — абсолютная величина x ;
- ❑ $SGN(x)$ — знак числа x ;
- ❑ $SQRT(x)$ — корень квадратный из x ;
- ❑ $EXP(x)$ — экспонента от x ;
- ❑ $LOG(x)$ — натуральный логарифм x ;
- ❑ $PWR(x, y)$ — степенная функция x^y ;
- ❑ $SIN(x)$, $COS(x)$, $TAN(x)$, $ATAN(x)$ — тригонометрические функции.

Эти функции, в свою очередь, могут входить в другие математические выражения, которые, в соответствии с действующими правилами, включаются в директиву `.PARAM` и должны заключаться в фигурные скобки. В результате в текстовый файл задания `*.nsx` включается строка вида:

```
.PARAM A = {B*SIN(C)}
```

Зададимся целью построить для схемы, приведенной на рис. 9.9, семейство частотных характеристик в диапазоне изменения коэффициента усиления от 2 (при равенстве сопротивлений $R_2 = R_3 = 1$ кОм) до 1001 (при $R_3 = 1$ МОм). Частотные характеристики цепей, заданные таким образом, обычно строятся с градуировкой оси ординат в децибелах и в логарифмическом масштабе по оси частот. Для того чтобы при градуировке оси ординат в децибелах получить равномерное распределение графиков по вертикали, следует задать изменение сопротивления резистора R_3 в геометрической прогрессии. Составим выражение, из которого будет вычисляться значение сопротивления

резистора R3 при последовательных прогонах задания. Чтобы программа строила частотные характеристики, распределяя их на графике по вертикали с шагом 10 дБ, по крайней мере, при коэффициентах усиления схемы более 20 дБ, сопротивление резистора R3 должно от прогона к прогону меняться по закону:

$$R3 = R2 \times 3,17^k, \quad (9.2)$$

где $k = 2, 3, 4, 5, 6$.

Для этого нужно будет воспользоваться операциями с глобальными параметрами.

Рассмотрим процедуру настроек параметрического анализа применительно к исследованию схемы, приведенной на рис. 9.8.

1. Не выходя из диалога настройки режимов анализа (см. рис. 9.10), выбрать в панели диалогового окна опцию **Global Parameters**. В диалоговом окне откроется панель настройки **Global Parameters Setup** (рис. 9.11).
2. Щелчками на кнопке **Add** активизировать назначение четырех новых глобальных параметров, назначить параметрам имена и значения (**Value**):

- $BASE = 3.17$ — основание степени в степенном множителе выражения (9.2): при возведении его в целочисленные степени и умножении на этот множитель сопротивления резистора обратной связи R3 получается ряд следующих значений сопротивления:

$$R2 \times 3,17^2 = 10 \text{ кОм, что дает усиление 20 дБ,}$$

$$R2 \times 3,17^3 = 31,8 \text{ кОм — усиление 30 дБ,}$$

$$R2 \times 3,17^4 = 100 \text{ кОм — усиление 40 дБ и т. д.}$$

- **Coeff** — показатель степени в выражении (9.2), который будет варьироваться при выполнении параметрического анализа;
- **R2_VALUE** — параметр, определяющий значение сопротивления резистора $R2 = 1 \text{ к}$ (1 кОм);
- **R3_VALUE** — параметр, определяющий значение сопротивления R3, изменяющееся в геометрической прогрессии при каждом повторном выполнении анализа. Значение сопротивления R3 будет вычисляться из выражения, которое запишем в графе **Value**:

$$R3_VALUE = \{R2_VALUE * PWR(BASE, coeff)\}.$$

Здесь $PWR(x, y)$ — степенная функция: $PWR(x, y) = x^y$.

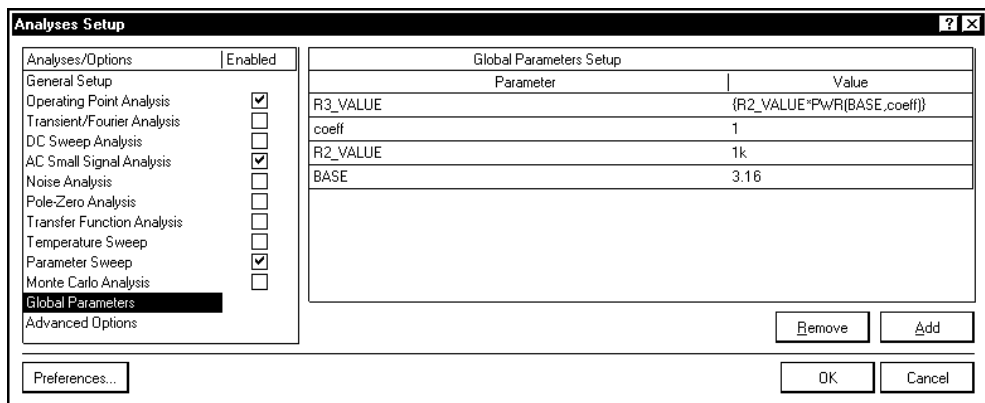


Рис. 9.11. Окно настройки режимов моделирования:
назначение глобальных параметров

3. Активизировать и установить признак **Enabled** для функции параметрического анализа. Откроется панель настроек режима анализа **Parameter Sweep Setup** (рис. 9.12). Программа допускает проведение двухмерного параметрического анализа: для каждого шага вариации первичного параметра (**Primary**) может выполняться полный цикл вариации вторичного параметра (**Secondary**), но в нашей задаче нам это не требуется, поэтому необходимо назначить настройки только для первичного параметра:

- **Primary Sweep Variable** — переменная, варьируемая в первичном цикле: в нашем случае разворачиваем список возможных параметров и выбираем показатель степени `coeff`;
- **Primary Start Value** — начальное значение параметра 0.000;
- **Primary Stop Value** — конечное значение параметра 6.000;
- **Primary Step Value** — значение шага вариации параметра 1.000;
- **Primary Sweep Type** — тип вариации — назначать абсолютные значения параметра.

На этом закончим настройки режимов анализа и щелчком на кнопке **OK** перейдем в главное графическое окно программы. Программа при этом формирует файл списка цепей и компонентов схемы *.nsx, присоединяет к нему директивы моделирования и включает его в плавающую панель **Projects** в ветвь **Generated | Advanced Sim Netlists** дерева структуры активного открытого проекта. Двойным щелчком левой кнопки мыши на

имени файла он делается доступным для просмотра в главном окне программы.

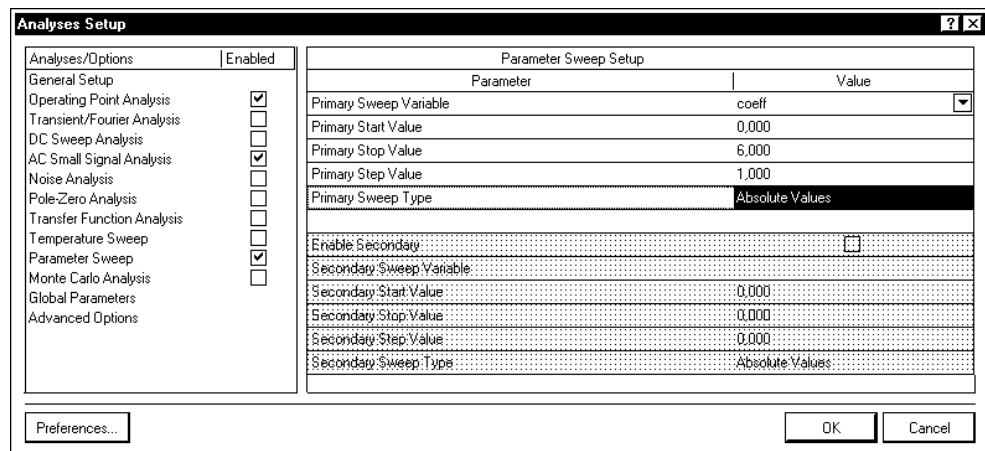


Рис. 9.12. Окно настройки режимов моделирования: вариация параметра — Parameter Sweep

- Щелчком на кнопке **Run Mixed-Signal Simulation** в меню **Mixed Sim** запустить выполнение задачи. Система моделирования выполняет расчеты для всех заданных видов анализа, формирует файл результатов **OP_AMP_AC.sdf**, включает его в ветвь **Generated | Simulation Documents** дерева структуры проекта и выводит данные в главное окно программы (рис. 9.13). Данные в выходном файле представляются в графической и текстовой формах на нескольких вкладках, в соответствии с назначенными видами анализа.

Полученные графики подтверждают высказанные ранее соображения о характере частотной характеристики операционного усилителя и взаимообусловленности значений коэффициента усиления и полосы пропускания рассматриваемой схемы с отрицательной обратной связью.

Когда графики построены, справа от них располагается список имен узлов цепи, напряжения или токи в которых отображаются графиками. В случае, приведенном на рис. 9.13, показаны результаты параметрического анализа, снятые с одного и того же узла цепи — с ее выхода. Здесь в списке указано одно и то же имя, а сопровождают его обозначение номера прохода (p1, p2 и т. д.).

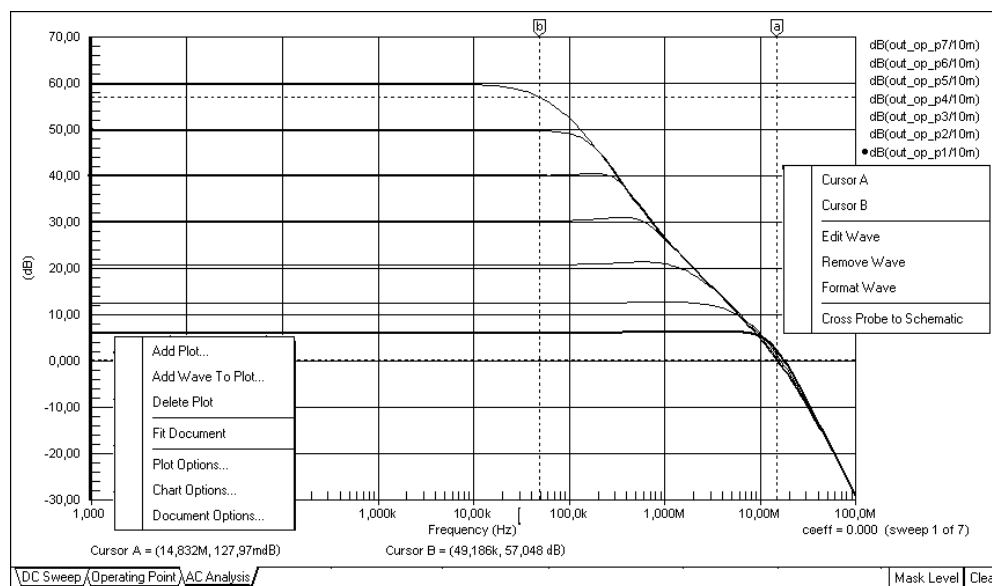


Рис. 9.13. Результат моделирования частотной характеристики ОУ при изменении глубины ООС

9.1.8. Представление результатов моделирования

Программа располагает развитыми средствами управления графическим представлением результатов. Начнем с рассмотрения средств, доступных непосредственно в графическом окне программы.

- Щелчок левой кнопки мыши на имени, соответствующем одной из кривых, выделяет ее утолщенной линией, остальные кривые "бледнеют". Это называется фильтрацией волны. В правом нижнем углу обрамления главного окна программы командой **Mask Level** можно активизировать управление глубиной маскирования остальных, не выделенных кривых (рис. 9.14). Вызывается пиктограмма с движком, положение которого и определяет уровень затенения всех кривых, кроме выделенной.

Повторный щелчок на имени выделенной кривой снимает фильтрацию. Тот же результат дает команда главного меню **Wave | Clear Filter**.

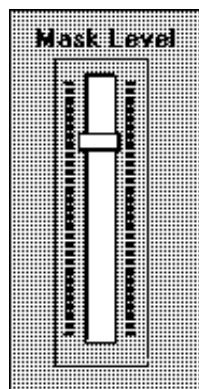


Рис. 9.14. Шкала глубины маскирования графиков

- Щелчок правой кнопкой мыши на имени кривой на рис. 9.13 точно так же выделяет ее, но вдобавок к этому активизируется контекстное меню управления видом графиков (см. рис. 9.13), содержащее команды:
 - **Cursor A** и **Cursor B** — команды вызова двух электронных курсоров, позволяющих выполнить замеры на одной или двух кривых, если выбрать кривые по очереди и на каждую наложить свой курсор. Результаты замеров отображаются строкой в нижней части графического окна. Эти же данные передаются в плавающую панель **Sim Data** (рис. 9.15) и отображаются в поле **Measurement Cursors**. Так, курсоры на рис. 9.13 указывают частоту среза частотной характеристики при усилении 60 дБ (курсор B) и частоту единичного усиления, составляющую, по данным замера, 14,832 МГц — практически паспортное значение для усилителя K140УД11 (показания курсора A). Строкой ниже отображается разность показаний курсоров B – A по осям X и Y;
 - **Edit Wave** — редактировать кривую. Опция идентична команде главного меню **Wave | Edit Wave**. В англоязычной лексике Altium Designer кривые на графиках называются словом **Wave** (волна). Эта команда открывает диалог редактирования **Edit Waveform** (рис. 9.16), содержащий поля:
 - ◇ **Waveform** — отображается список активных сигналов схемы, назначенных на стадии настройки задания. При указании курсором имени одного из сигналов оно копируется в поле **Expression**;

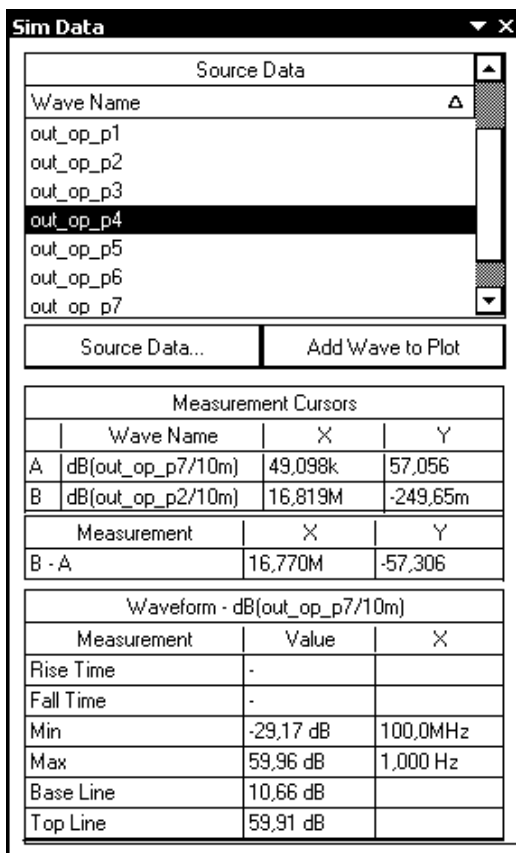


Рис. 9.15. Сводка результатов моделирования в панели **Sim Data**

- ◇ **Expression** — можно составить математическое выражение для преобразования выбранного сигнала;
- ◇ **Functions** — собраны математические знаки, включающиеся в выражения в поле **Expression**;
- ◇ **Complex Functions** — устанавливается форма представления сигнала или его преобразования на графике:
 - ◆ **Magnitude** — значение (в единицах измерения, соответствующих природе сигнала, — вольтах, амперах и т. п.);
 - ◆ **Magnitude (dB)** — значение в децибелах;
 - ◆ **Real** — вещественная составляющая комплексной величины;

- ♦ **Imaginary** — мнимая составляющая комплексной величины;
- ♦ **Phase (Deg)** — фаза напряжения или тока в градусах;
- ♦ **Phase (Rad)** — фаза напряжения или тока в радианах;
- ♦ **Group Delay** — время группового запаздывания.

Щелчком на кнопке **Create** (Создать) диалог закрывают, и отредактированная кривая отображается в графическом окне.

- **Remove Wave** — по этой команде выбранная кривая удаляется с графика.
- **Format Wave** — по этой команде открывается одноименное диалоговое окно (рис. 9.17), в полях которого можно назначить новое имя выбранной кривой, указать единицы измерения и задать цвет кривой на графике.

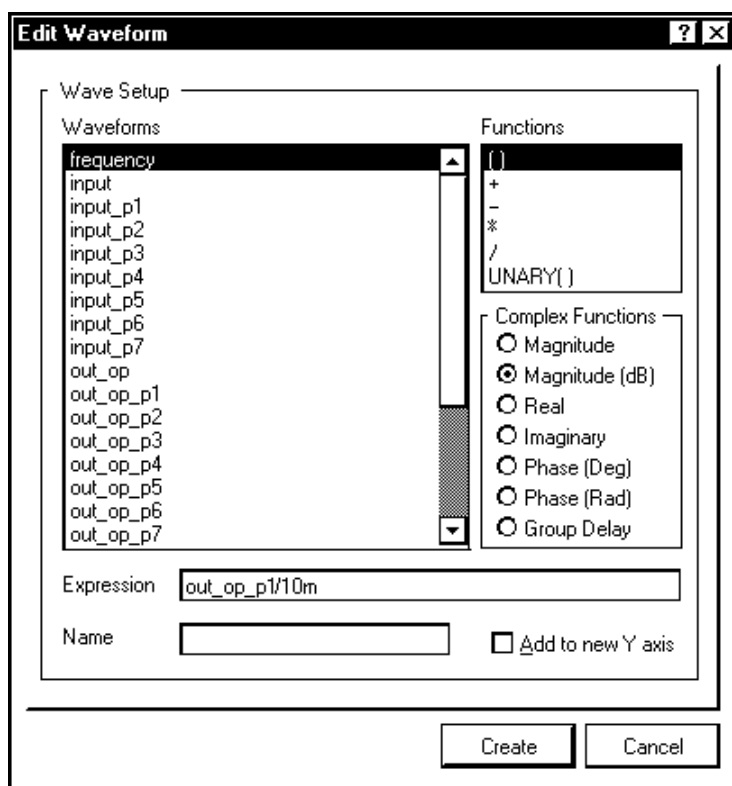


Рис. 9.16. Окно редактирования форм волны

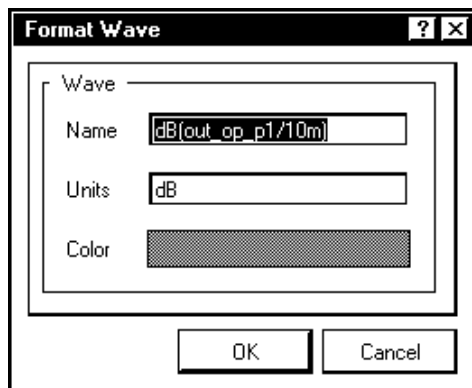


Рис. 9.17. Окно настройки формата волны

- **Cross Probe to Schematic** — отметить в схеме в случае, если в качестве выделенного сигнала в графике выбран ток, отдаваемый в ветвь цепи источником сигнала (с именем сигнала, например, `v1#branch`), по этой команде в схеме выделяется УГО данного источника, а вся остальная схема маскируется.
- Щелчок правой кнопки мыши на оси ординат вызывает контекстное меню с двумя командами — **Delete Axis** и **Format Axis**. Если к графику не добавлялись новые *Y*-оси, в меню доступна только команда **Format Axis**, активизирующая диалоговое окно настройки (рис. 9.18):
 - в поле **Y Axis** назначают метку (**Label**), единицы измерения и масштаба по вертикальной оси;
 - в поле **Scale** устанавливают масштаб по вертикали — размах от минимума до максимума и шаг деления шкалы (**Division Scale**).
- Щелчок правой кнопки мыши в поле графиков открывает другое контекстное меню (рис. 9.19), включающее команды:
 - **Add Plot** — добавить в графическое окно новую рамку для формирования в ней графиков: по этой команде открывается диалоговое окно мастера построения графиков **Plot Wizard**. Построение новой рамки и графиков в ней выполняется в три шага:
 - 1) **Plot Title** — назначить заголовок графиков;
 - 2) **Plot Appearance** — назначить изображение координатных осей и сетки;

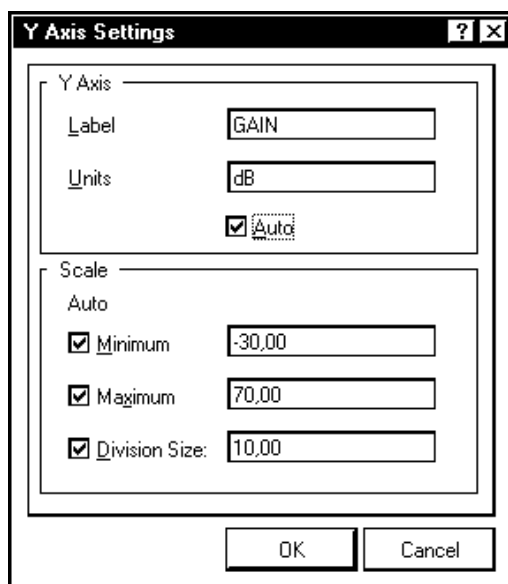


Рис. 9.18. Настройка оси ординат графика

- 3) **Add Waves to Plot** — назначить сигналы для представления на графике.

Последняя команда открывает окно **Add Wave to Plot**, повторяющее в точности окно редактирования волны (см. рис. 9.16). В результате в графическом окне строится новый график с выведенными в него кривыми. В левом обрамлении графического окна против каждого графика появляется кнопка, в которую курсором ставится указатель активности.

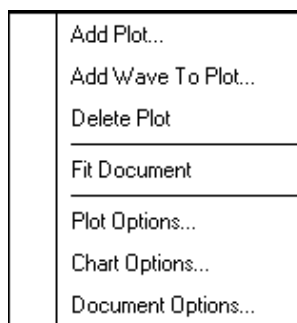


Рис. 9.19. Контекстное меню команд управления графиками

- **Delete Plot** — удалить график, отмеченный указателем активности;
- **Fit Document** — отмасштабировать изображение в активном графике в полные размеры, назначенные для осей абсцисс и ординат;
- **Plot Options** — команда настройки координатных осей и сетки графика;
- **Chart Options** — команда настройки элементов отображения — указание отображаемого вида анализа, имени и единиц измерения независимой переменной, пределов и масштаба отображения по оси X, величин, отображаемых при измерениях электронными курсорами;
- **Document Options** — по этой команде открывается диалоговое окно общих настроек графического документа (рис. 9.20):
 - ◇ в поле **Colours** настраивают цвет фона (**Background**), сетки (**Grid**) и линий на переднем плане (**Foreground**);
 - ◇ кнопкой **Swap Foreground/Background** цвета фона и линий графиков меняются местами — получается негативное изображение;
 - ◇ в поле **View** (Вид) "птичками" активизируют отображение заголовков, меток осей, нанесение точек на графики, изображение кривых утолщенной линией (**Bold Waveforms**);

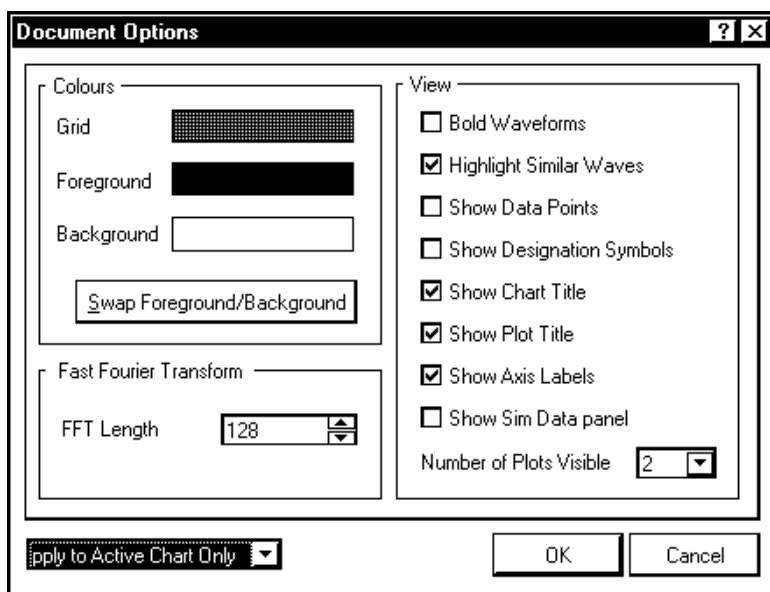


Рис. 9.20. Окно настроек отображения графического документа

- ◇ в поле **Number of Plots Visible** устанавливают число графиков, отображаемых в графическом окне;
- ◇ в поле **Fast Fourier Transform** указывают число точек быстрого преобразования Фурье (при анализе переходных процессов).

Все рассмотренные, а также еще некоторые настройки, на которых мы не будем здесь останавливаться, доступны также из главного меню программы, в которое при выводе на экран графического файла *.sdf добавляются три команды — **Plot**, **Chart** и **Wave**.

9.1.9. Анализ переходных процессов

Рассмотрим анализ переходных процессов на примере активного фильтра нижних частот (ФНЧ) Чебышева на том же операционном усилителе K140УД11. Схема ФНЧ построена на повторителе напряжения: выход ОУ замкнут на инвертирующий вход (рис. 9.21). В иностранных источниках такая схема называется Sallen-Key configuration.

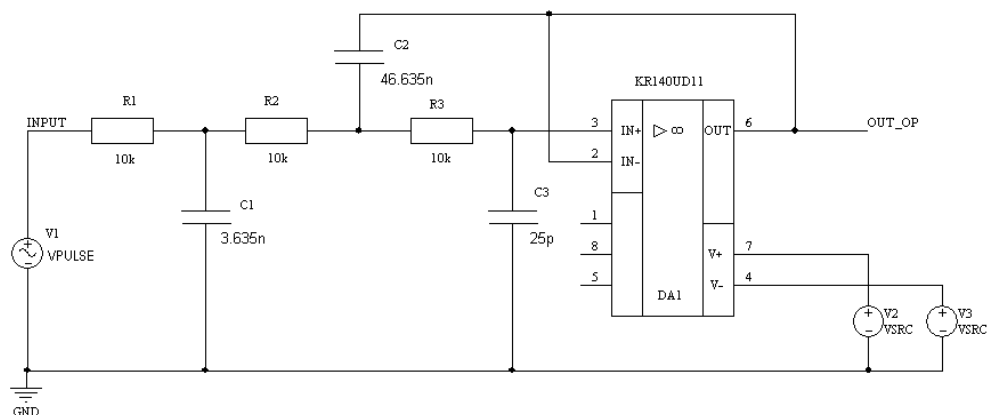


Рис. 9.21. Схема активного ФНЧ Чебышева третьего порядка

Входной сигнал вырабатывает независимый источник напряжения V1 типа VPULSE, параметры которого настраивают так, чтобы можно было выполнить следующие виды анализа:

- ☐ по постоянному току (**DC Sweep**);
- ☐ частотной характеристики (**AC Sweep**);

- ☐ переходных процессов;
- ☐ преобразование Фурье от результата анализа (**Transient/Fourier Analysis**).

Не будем останавливаться на настройках анализа по постоянному току и частотной характеристике, рассмотренных ранее, а обратимся к настройке анализа переходных процессов.

Вначале следует щелчком на кнопке **Setup Mixed-Signal Simulation** в меню **Mixed Sim** открыть диалог настройки, активизировать настройки анализа переходных процессов **Transient/Fourier Analysis** (рис. 9.22) и настроить режимы анализа. При установленной опции **Use Transient Defaults** (Использовать настройки по умолчанию) программа рассчитывает и строит графики на интервале, равном пяти периодам входного сигнала, по 50 значений на период. Эти настройки отображаются в строках **Default Cycles Displayed** (Число периодов по умолчанию) и **Default Points Per Cycle** (Число точек на период по умолчанию) в поле параметров окна настройки. Верхняя половина панели настроек при этом затеняется.

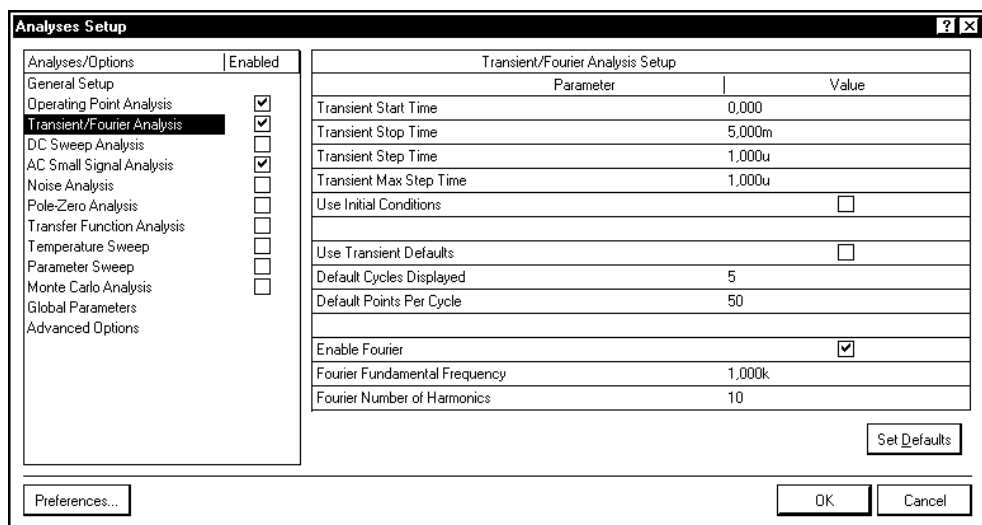


Рис. 9.22. Окно настройки режимов моделирования: анализ переходных процессов и Фурье-анализ

При анализе рассматриваемого примера откажемся от настроек по умолчанию и назначим в поле **Parameters** значения **Value** следующим параметрам:

- ☐ **Transient Start Time** — момент начала отсчета времени анализа 0.000;

- ❑ **Transient Stop Time** — момент окончания расчета 5.000m (5 мс);
- ❑ **Transient Step Time** — шаг интегрирования 1.000u (1 мкс);
- ❑ **Transient Max Step Time** — значение верхнего предела шага интегрирования 1.000u (1 мкс);
- ❑ **Enable Fourier** — активизировать преобразование Фурье: по этой опции вычисляется амплитуда первой и заданного числа высших гармоник сигнала в указанных узлах схемы;
- ❑ **Fourier Fundamental Frequency** — частота основной (первой) гармоники сигнала 1.000k (1 кГц);
- ❑ **Fourier Number of Harmonics** — число высших гармоник 10.

Результаты выполнения преобразования Фурье отображаются в файле результатов *.sdf в форме таблицы. Одновременно программа вычисляет значение коэффициента нелинейных искажений исследуемого сигнала.

Затем нужно указать курсором строку настройки анализа частотной характеристики (см. рис. 9.10) и задать верхнюю частоту диапазона 1 МГц, деление диапазона на декады, расчет и вывод значений выходного сигнала в 40 точках на декаду. Щелчком на кнопке **OK** закрыть диалог настройки и запустить выполнение задачи кнопкой **Run Mixed-Signal Simulation** (см. рис. 9.4).

Приведем сначала результат анализа частотной характеристики.

На вкладку **AC Sweep** выводится зависимость амплитуды и фазы выходного сигнала от частоты — АЧХ и ФЧХ фильтра (рис. 9.23). Поскольку на входе схемы действует гармонический сигнал амплитудой 1 В, значения ординат верхнего графика численно равны значениям коэффициента передачи цепи. Графики построены с использованием приемов, рассмотренных в *разд. 9.1.8*. Частотная характеристика имеет характерную для чебышевских фильтров неравномерность коэффициента передачи в полосе пропускания (рис. 9.23, а).

На графике АЧХ (рис. 9.23, б) значение выражено в децибелах. Неравномерность, измеренная электронным курсором, составляет 3 дБ. Именно эта величина задавалась при проектировании фильтра. Частота среза 15,304 кГц также близка к расчетному значению 15,92 кГц.

На рис. 9.23, в приведена фазо-частотная характеристика (ФЧХ) фильтра.

Анализ переходных процессов может преследовать различные цели. При моделировании импульсных и релаксационных цепей, генераторов пилообразного напряжения это может быть исследование формы сигнала в различных узлах цепи, временных задержек и т. п.

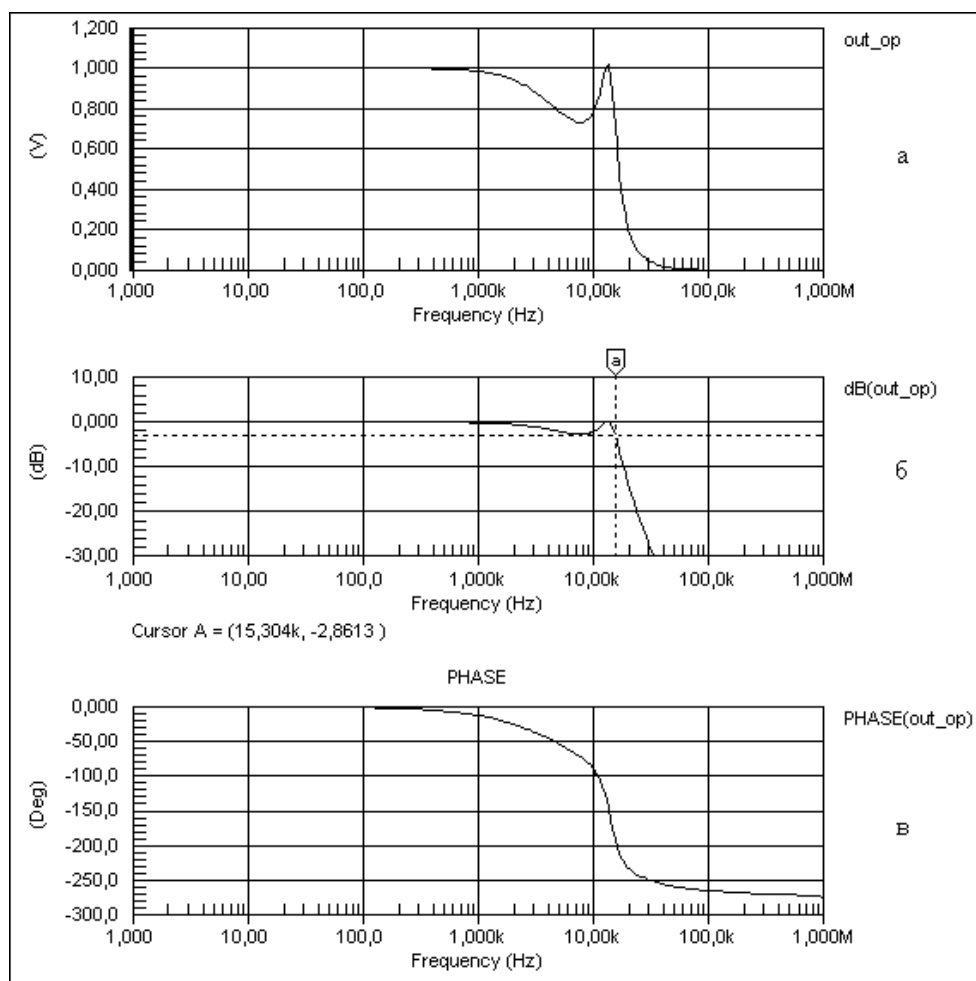


Рис. 9.23. Частотная и фазовая характеристики ФНЧ по рис. 9.21

При моделировании линейных цепей, таких как резонансные контуры, электрические фильтры, обычно исследуют соответствие характеристик цепи определенным теоретическим предпосылкам. В этом случае удобна переходная или импульсная характеристика цепи. При известной импульсной характеристике и произвольном входном сигнале переходный процесс на выходе получается как результат вычисления интеграла наложения:

$$u_{\text{ВЫХ}}(t) = \int_0^t u_{\text{ВХ}}(\tau) h(\tau - t) d\tau. \quad (9.3)$$

При выполнении данного вида анализа программа находит импульсную характеристику цепи и рассчитывает в дискретной форме значения интеграла наложения (интеграл Дюамеля) на заданном временном интервале. К сожалению, сама импульсная характеристика цепи остается недоступной для непосредственного наблюдения. Для ее построения воспользуемся косвенным методом. Найдем реакцию цепи при воздействии на вход единичного скачка напряжения (функции Хевисайда). Эта реакция называется переходной характеристикой цепи. Далее получим график производной от переходной характеристики. Из теории линейных электрических цепей известно, что эта производная и есть импульсная характеристика цепи (функция Грина).

Вычислительное ядро SPICE-системы моделирования допускает назначение нулевого времени нарастания фронтов импульсного сигнала. Поэтому при настройке параметров источника входного сигнала V1 укажем значение параметра **Rise Time** равным нулю.

Для построения импульсной характеристики необходимо выполнить следующую процедуру:

1. В среде активного документа *.sdf щелчком правой кнопки мыши вызвать контекстное меню и в нем указать команду **Add Plot** или командой главного меню **Plot | New Plot**. Запускается процедура формирования графика **Plot Wizard** (см. рис. 9.19). На первом шаге указать имя нового графика.
2. Настроить элементы отображения координатной сетки нового графика.
3. Щелчком на кнопке **Add** запустить построение нового графика. Открывается диалоговое окно **Add Wave to Plot** (рис. 9.24).
4. В поле **Functions** выбрать функцию вычисления производной $\text{DER}()$ ¹. Имя функции $\text{DER}()$ копируется в поле **Expression**.
5. В поле **Waveforms** выбрать имя выходной цепи схемы out_op. В поле **Expression** формируется выражение $\text{DER}(\text{out_op})$.
6. Кнопкой **Create** образовать график производной выходного сигнала, которая и будет искомой импульсной характеристикой цепи.

На рис. 9.25 приведены результаты моделирования переходной (верхний график) и импульсной (нижний график) характеристик ФНЧ, рассматриваемого в нашем примере.

¹ DER — от английского Dervative — производная.

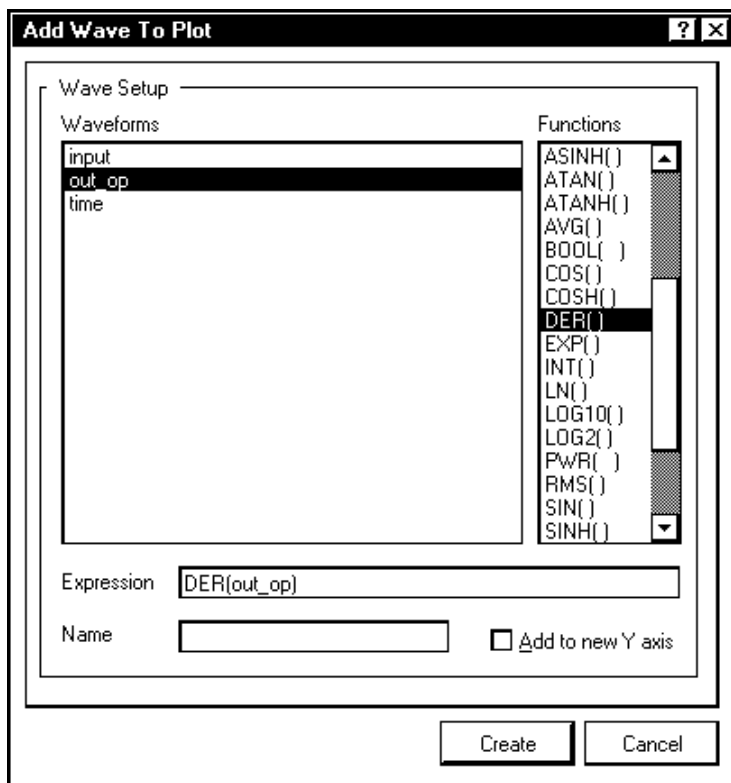


Рис. 9.24. Диалоговое окно образования нового графика

На рис. 9.26 приведены результаты моделирования переходного процесса на выходе того же ФНЧ (см. рис. 9.21) под воздействием линейно нарастающего входного сигнала с длительностью переднего фронта (параметр **Rise Time** источника входного сигнала V1) 50 мкс.

На верхнем графике рис. 9.26 показана временная диаграмма входного сигнала, а на нижнем — переходный процесс в выходной цепи фильтра.

Результаты анализа переходного процесса при воздействии на вход схемы периодической последовательности прямоугольных импульсов с периодом 2 мс и длительностью переднего и заднего фронтов импульса по 1 мкс, отображаемые на вкладке **Transient Analysis** главного графического окна программы, показаны на рис. 9.27.

Остановимся еще на одной функции обработки результатов анализа. В дополнение к Фурье-анализу в режиме моделирования переходного процесса,

дающего в конечном результате таблицу значений амплитуд гармоник выходного сигнала цепи, в Altium Designer введена функция быстрого преобразования Фурье (Fast Fourier Transform) непосредственно над значениями выходного сигнала.

Чтобы осуществить БПФ, нужно в среде активного документа *.sdf активизировать команду главного меню **Chart | Create FFT Chart** (График | Создать график БПФ). Программа выполняет преобразование и выводит результат в главное графическое окно (рис. 9.28).

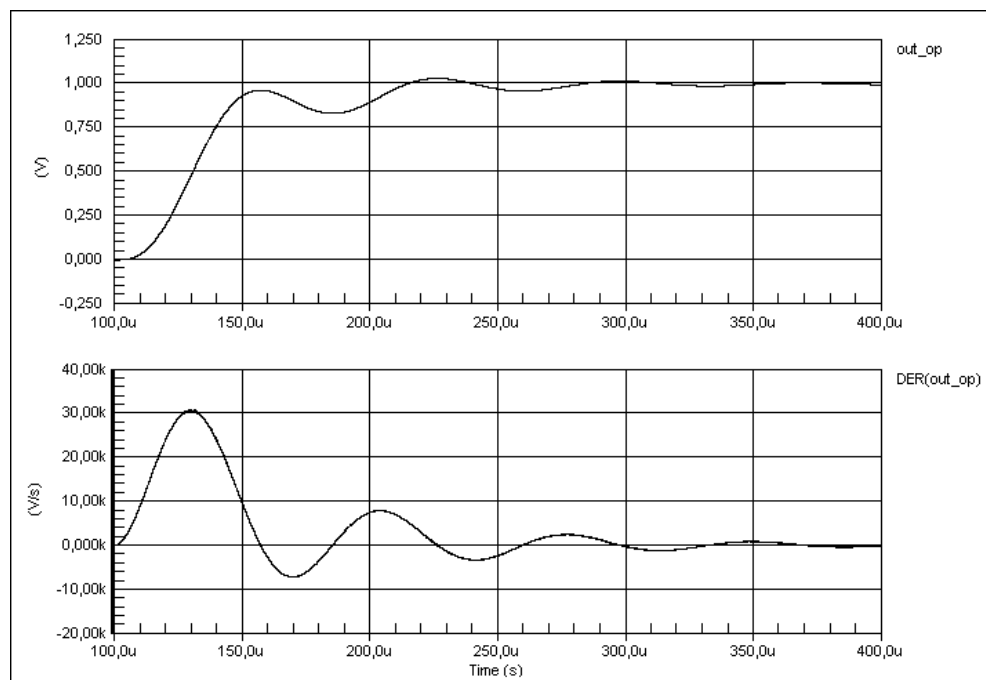


Рис. 9.25. Переходная и импульсная характеристики ФНЧ по рис. 9.21

Известно, что частотный спектр периодической последовательности импульсов прямоугольной формы содержит только нечетные гармоники. Этот факт подтверждается спектром, приведенным на рис. 9.28: огибающая, проведенная по вершинам линий полученного спектра, будет иметь пики на частотах 500, 1500, 2500 Гц и т. д., т. е. на частоте первой, третьей, пятой и следующих нечетных гармоник частоты следования импульсов 500 Гц (см. рис. 9.27).

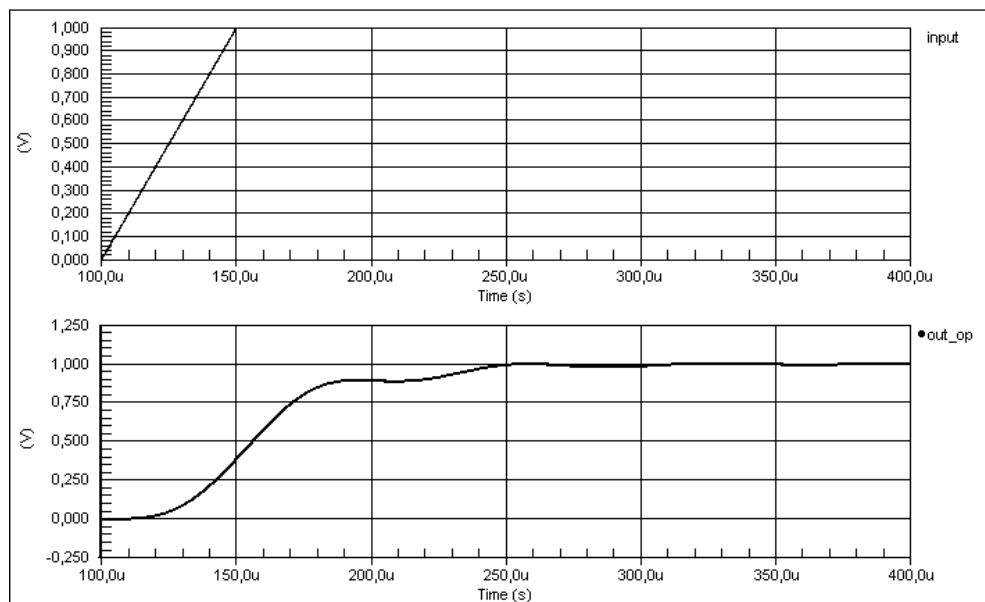


Рис. 9.26. Переходный процесс на выходе ФНЧ под воздействием трапециевидного входного сигнала

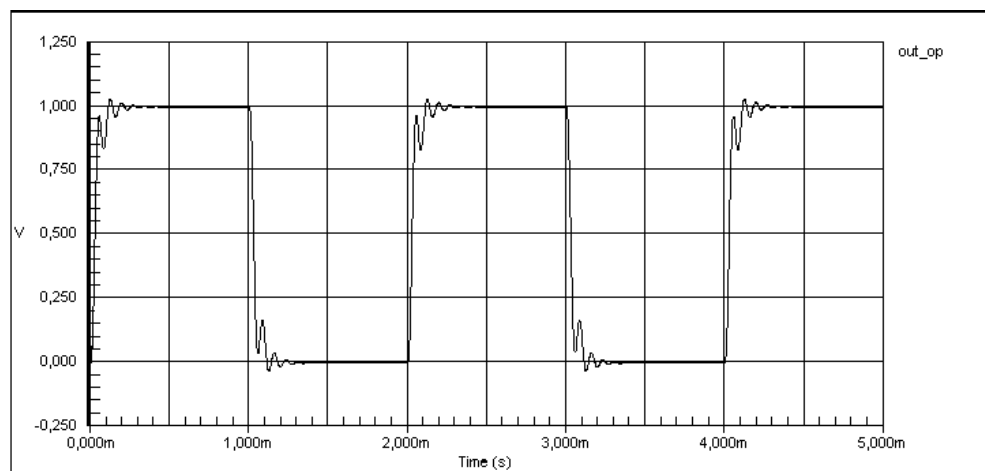


Рис. 9.27. Переходный процесс на выходе ФНЧ под воздействием периодического сигнала прямоугольной формы

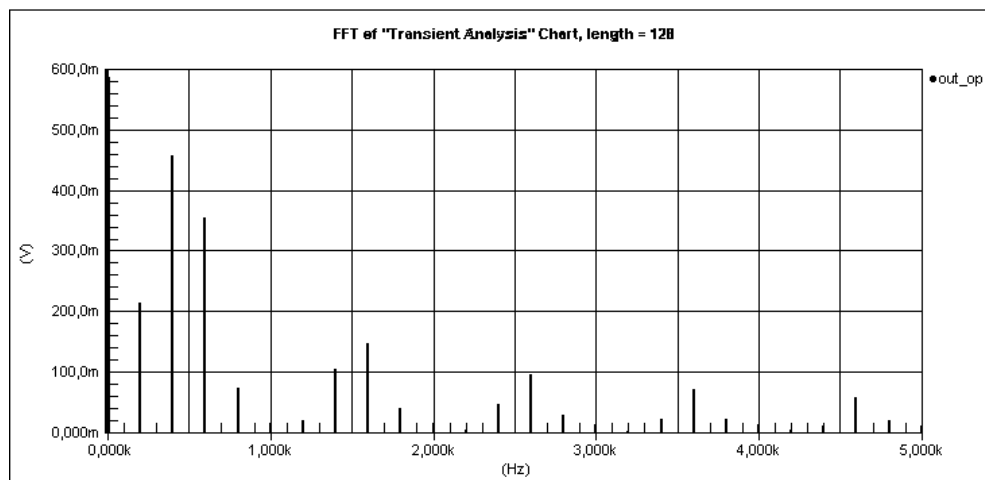


Рис. 9.28. Результат быстрого преобразования Фурье сигнала по рис. 9.27

9.1.10. Анализ функции передачи по постоянному току

Анализ функции передачи по постоянному току **Transfer Function Analysis** необходимо совместить с анализом цепи по постоянному току (**DC Sweep**). При выполнении задачи программа рассчитывает значения входного и выходного сопротивлений схемы и коэффициент передачи (усиления) цепи по постоянному току.

В окне настройки анализа передаточной функции в качестве аргумента указывается имя единственного элемента схемы, по отношению к которому вычисляются названные значения — источника входного сигнала.

При формировании задания в формате внутреннего языка системы моделирования — в файле *.nsx составляются директивы моделирования .TF для всех сигнальных цепей схемы и источников питания и сигнала.

Результаты анализа отображаются в файле *.sdf в текстовой форме на вкладке **Transfer Function**.

9.1.11. Анализ нулей и полюсов передаточной функции

Система моделирования Altium Designer позволяет найти полюсы и нули передаточной функции исследуемой цепи. Рассмотрим этот вид анализа на примере пассивного ФНЧ Баттерворта 5-го порядка с круговой частотой среза $\omega_c = 1000$ рад/с (рис. 9.29).

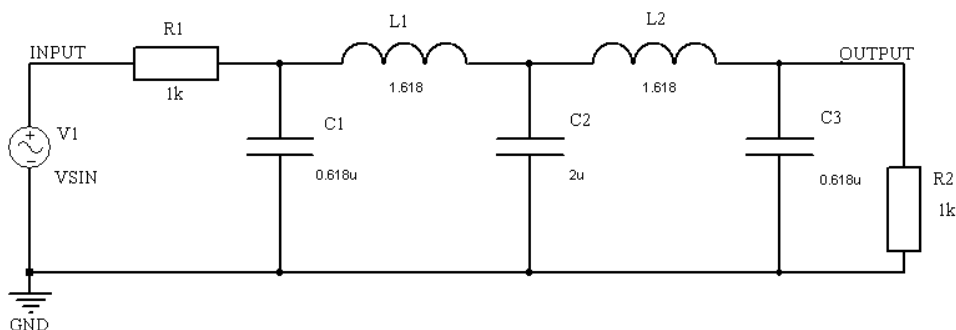


Рис. 9.29. Схема пассивного ФНЧ Баттерворта пятого порядка

Сначала проанализируем частотную характеристику фильтра. Вся процедура настройки и выполнения анализа в частотной области (**AC Sweep**) рассмотрена ранее, поэтому приведем сразу график АЧХ цепи (рис. 9.30).

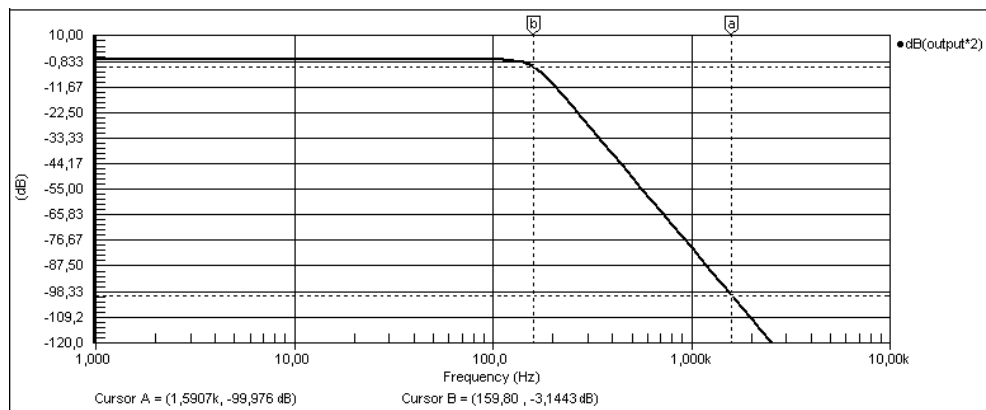


Рис. 9.30. Частотная характеристика ФНЧ по рис. 9.29

Полученная характеристика подтверждает ожидаемые свойства цепи:

- ❑ частота среза F_c (частота, на которой коэффициент передачи цепи падает до 0,707 значения при нулевой частоте) составляет 159,8 Гц, что соответствует круговой частоте $\omega_c = 2\pi F_c = 1000$ рад/с;
- ❑ значение коэффициента передачи на частоте $10F_c = 1,59$ кГц составляет $-99,97$ дБ, что с высокой точностью соответствует значению крутизны спада АЧХ за пределами полосы пропускания ФНЧ Баттерворта 5-го порядка, равному $20n = 20 \cdot 5 = 100$ дБ на декаду, где $n = 5$ — порядок фильтра.

Рассмотрим теперь настройку анализа нулей и полюсов передаточной функции цепи.

1. Щелчком на кнопке **Setup Mixed-Signal Simulation** в меню **Mixed Sim** открыть диалог настройки и активизировать настройки анализа нулей и полюсов **Pole-Zero Analysis** (рис. 9.31).

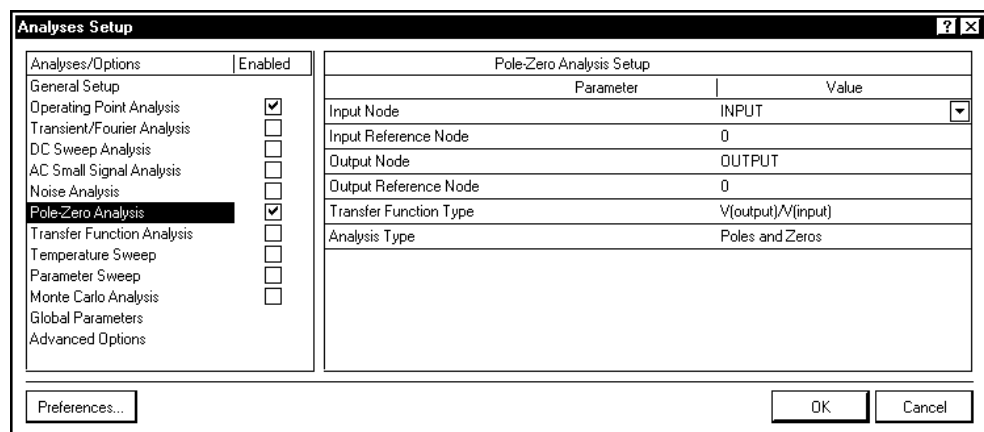


Рис. 9.31. Окно настройки режимов моделирования: анализ нулей и полюсов передаточной функции

2. В поле **Analyses/Options** задать настройку в колонке **Enabled** и указать курсором режим **Pole-Zero Analysis**.
3. В строках поля настроек **Pole-Zero Analysis Setup** указать имена входного (**Input Node**) и выходного (**Output Node**) узлов цепи: в нашем примере это имена INPUT и OUTPUT.

Остальные настройки анализа, установленные по умолчанию, должны нас устраивать:

- узлы, относительно которых программа исчисляет значения входного и выходного сигнала (**Input Reference Node** и **Output reference Node**) — "земля" (**Value 0**);
- вид передаточной функции (**Transfer Function Type**) определяется выражением $V(\text{output})/V(\text{input})$;
- вид анализа (**Analysis Type**) — **Poles and Zeros** — полюсы и нули.

- Щелчком на кнопке **ОК** закрыть диалог настроек и запустить выполнение задачи.

Transfer Function of $(V(\text{output}) - V(0)) / (V(\text{input}) - V(0))$ in Volts / Volt

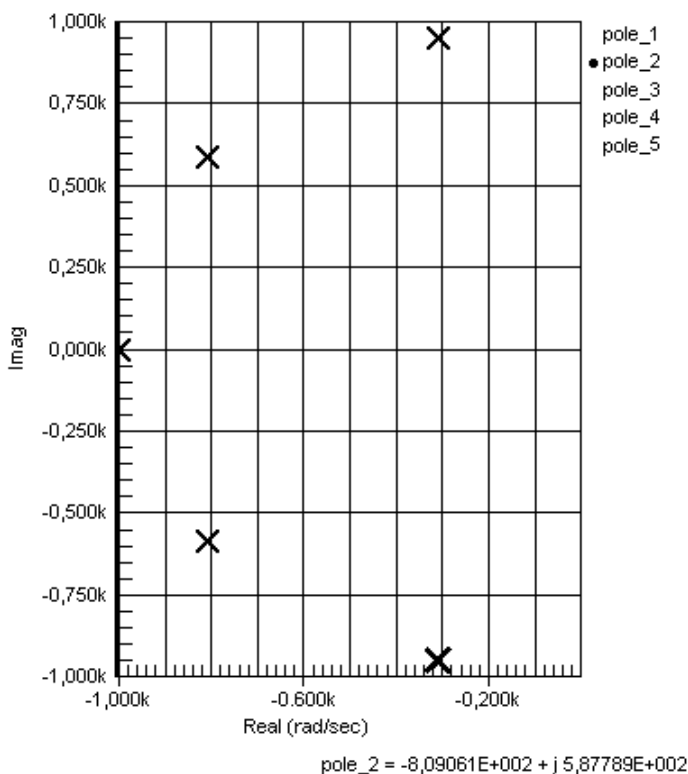


Рис. 9.32. Нуль-полюсная диаграмма ФНЧ по рис. 9.29

Программа выполняет назначенные виды анализа и выводит нуль-полусную диаграмму в графическое окно на вкладке **Pole-Zero Analysis**. Диаграмма строится в осях Real-Imaginary. Полюсы передаточной функции обозначаются крестиками, нули — кружками (рис. 9.32). В нашем примере нули отсутствуют, поскольку цепь минимально-фазовая.

Известно, что полюсы передаточной функции по мощности фильтра Баттерворта порядка n располагаются на окружности единичного радиуса через равные углы, кратные $2\pi/n$.

Комплексная частотная характеристика нормированного фильтра Баттерворта 5-го порядка имеет пять полюсов, лежащих в левой комплексной полуплоскости: вещественный полюс $p_1 = -1$ и две пары комплексно-сопряженных полюсов $p_2, p_3 = -0,8090 \pm j0,5877$; $p_4, p_5 = -0,3090 \pm j0,9510$.

Приведенные на рис. 9.32 значения вещественной и мнимой компонент одного из полюсов составляют $-0,8090 \cdot 10^3$ и $+0,5877 \cdot 10^3$, что соответствует компонентам полюса p_2 , умноженным на 1000, т. е. на значение круговой частоты среза. При указании курсором на имена остальных четырех полюсов значения, отображаемые на экране, также подтверждают, что исследуемая цепь есть ФНЧ Баттерворта 5-го порядка. Таким образом, результат анализа соответствует высказанным теоретическим предположкам.

9.1.12. Моделирование случайных разбросов параметров компонентов

Цель данного анализа — исследовать влияние случайного разброса значений параметров компонентов схемы на ее выходную функцию. Он выполняется совместно с одним или несколькими стандартными видами анализа (DC Sweep, AC Sweep, Transient/Fourier) и проводится методом статистических проб (методом Монте-Карло) путем нескольких прогонов основного анализа с различными случайными отклонениями параметров компонентов схемы от номинала.

Параметрам компонентов назначаются случайные отклонения в пределах назначенного предельного отклонения (допуска). Значения параметров для каждого прогона задачи вычисляются по формуле:

$$x = x_{\text{ном}}(1 + \Delta\xi), \quad (9.4)$$

где $x_{\text{ном}}$ — номинальное значение параметра; Δ — относительное предельное отклонение (допуск); ξ — вырабатываемая генератором псевдослучайных чисел центрированная случайная величина, определенная на интервале $(-1, +1)$ с заданным законом распределения.

Анализ может выполняться для двух видов случайного разброса параметров:

- от компонента к компоненту (признак Device);
- от партии к партии (признак Lot).

При анализе обязательно нужно назначить хотя бы один из видов разброса. Программа присваивает параметрам компонентов случайные отклонения индивидуально для каждого компонента и по каждому виду разброса (если не используется отслеживание). Оба отклонения вычисляются по формуле (9.4) и суммируются.

Совместное назначение обоих видов разброса имеет смысл, когда отклонения параметров индивидуальных компонентов нельзя считать независимыми, но и корреляция их слаба. Такая картина наблюдается, например, у резисторов, если допуск, указанный производителем для партий, имеет одно значение, а в пределах партии разброс значений сопротивления от резистора к резистору заметно меньше указанного допуска на партию. Такая же ситуация имеет место для интегральных микросхем, когда в пределах одного кристалла наблюдается высокая степень повторяемости параметров компонентов, а от микросхемы к микросхеме может наблюдаться значительно больший их разброс.

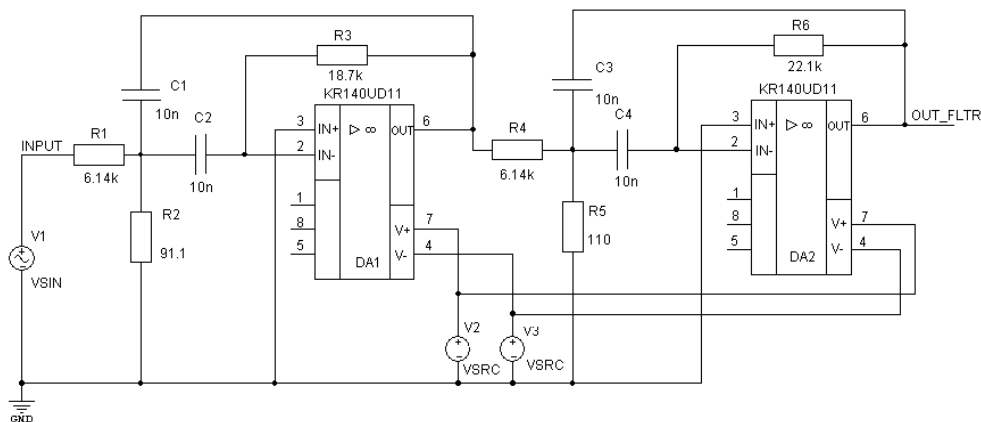


Рис. 9.33. Схема полосно-пропускающего фильтра Чебышева 4-го порядка

При указании сразу двух видов разброса программа назначает в пределах одного прогона задачи отклонения, обусловленные разбросом типа Device, а от прогона к прогону — отклонения, определяемые разбросом партий Lot.

Рассмотрим данный вид анализа на примере частотной характеристики активного полосно-пропускающего фильтра Чебышева 4-го порядка с центральной частотой 11 кГц и шириной полосы пропускания 3 кГц (рис. 9.33).

Последовательность действий такова.

1. В активном схемном документе указать команду меню **Setup Mixed-Signal Simulation** и активизировать в диалоговом окне настройки **Analyses Setup** вид анализа **Monte Carlo Analysis** (рис. 9.34).

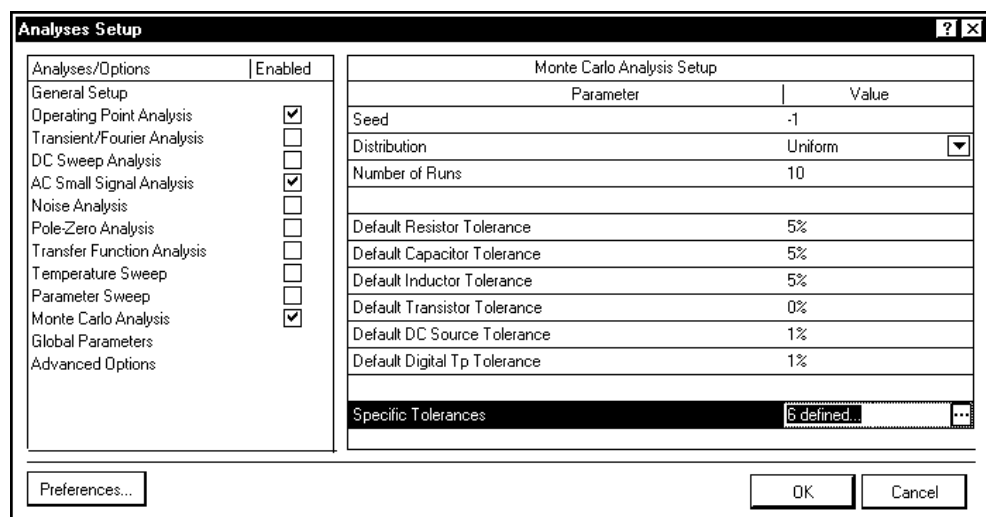


Рис. 9.34. Окно настройки режимов моделирования: анализ влияния разброса параметров методом Монте-Карло

2. В поле настроек **Monte Carlo Analysis Setup** выполнить следующие настройки режима анализа:

- установить значение параметра **Seed** (от 1 до 32 767), задающее последовательность, вырабатываемую генератором псевдослучайных чисел, определяющих величину отклонения параметров компонентов в процессе анализа. Значению -1 соответствует последовательность, вырабатываемая по умолчанию. Если возникают сомнения по поводу

результатов моделирования, можно выбрать другие последовательности случайных отклонений;

- в поле **Distribution** кнопкой в колонке **Value** раскрыть список и назначить вид закона распределения отклонений параметров компонентов от номинала:
 - ◇ **Uniform** — равномерное распределение в пределах указанного допуска;
 - ◇ **Gaussian** — гауссово (нормальное) распределение на том же интервале, при этом считается, что предельное отклонение параметра компонента равняется тройному стандарту распределения: $\Delta = 3\sigma$;
 - ◇ **Worst Case** — "худший случай", при назначении которого параметрам компонентов присваиваются только предельные значения, соответствующие краям поля допуска;
- в поле **Number of Runs** указать число прогонов основного анализа;
- в нижележащих полях диалоговой панели назначить допуски на параметры компонентов схемы (**Default Resistor Tolerance** и др.). Допуски могут быть назначены и использоваться по умолчанию для шести групп базовых компонентов:
 - ◇ резисторов;
 - ◇ конденсаторов;
 - ◇ катушек индуктивности;
 - ◇ транзисторов (коэффициент передачи базового тока BF);
 - ◇ источников постоянного напряжения и тока (DC Source);
 - ◇ логических схем (время распространения — Digital Tr Tolerance).

Допуск можно назначить как в натуральных единицах измерения параметра, так и в процентах от номинала. Применительно к исследуемой схеме (рис. 9.32) имеет смысл задать допуски только для резисторов, конденсаторов и источников питания. Назначим допуск $\pm 5\%$ для резисторов и конденсаторов и $\pm 1\%$ для разброса напряжения источников питания (**Default DC Source Tolerance**). Установленные допуски используются по умолчанию и распространяются на все компоненты исследуемой схемы.

ЗАМЕЧАНИЕ

Отметим, что назначенные допуски не распространяются на параметры элементов внутренней схемы макромоделей.

3. Активизировать страницу настройки анализа частотной характеристики цепи **AC Small Signal Analysis** и назначить:
 - **Start Frequency** 5.000k — нижнюю границу частотного диапазона 5 кГц;
 - **Stop Frequency** 25.00k — верхнюю границу частотного диапазона 25 кГц;
 - **Sweep Type Linear** — разбивку диапазона на равные интервалы;
 - **Test points** 501 — число точек, в которых вычисляются значения коэффициента передачи.
4. Щелчком на кнопке **OK** закрыть диалог настройки.
5. Кнопкой **Run Mixed-Signal Simulation** запустить выполнение задания. Программа выполняет заданное число прогонов и выводит результат в главное графическое окно (рис. 9.35).

Видим, что при 5-процентном допуске на пассивные компоненты форма частотной характеристики цепи может значительно отличаться от расчетной, построенной без учета разбросов. Заметен также большой разброс значений коэффициента передачи цепи в центре полосы пропускания.

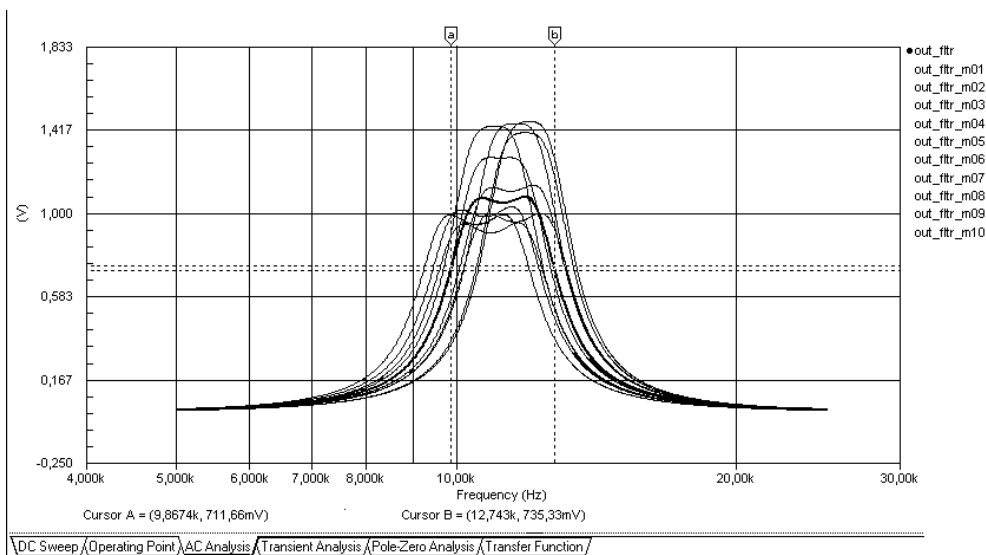


Рис. 9.35. Результаты анализа Монте-Карло:
частотная характеристика полосового фильтра по рис. 9.33

Для исследования влияния на выходную функцию разброса параметров одного или нескольких критических компонентов при настройке режимов анализа этим компонентам можно назначить индивидуальные настройки, которые обладают приоритетом над настройками по умолчанию.

Рассмотрим последовательность настройки и результаты моделирования для этого случая.

1. В диалоговом окне настройки (рис. 9.34) щелчком левой кнопки мыши в правом углу поля **Specific Tolerances** активизировать кнопку выбора и вызвать окно индивидуальных настроек (рис. 9.36).

Monte Carlo - Specific Tolerances							
		Device			Lot		
Designator	Parameter	Tolerance	Tracking No.	Distribution	Tolerance	Tracking No.	Distribution
R1		2%	6	Gaussian	5%	1	Uniform
R2		2%	1	Gaussian	5%	1	Uniform
R3		2%	2	Gaussian	5%	1	Uniform
R4		2%	4	Gaussian	5%	1	Uniform
R5		2%	5	Gaussian	5%	1	Uniform
R6		2%	3	Gaussian	5%	1	Uniform

Рис. 9.36. Настройка индивидуальных разбросов параметров компонентов

2. Щелчком на кнопке **Add** выделить первую строку в поле индивидуальных настроек.
3. Щелчком мышью в правом конце поля **Designator** активизировать кнопку выбора и выбрать из раскрывающегося списка позиционное обозначение компонента.
4. Поле **Parameter** оставить пустым! (В прежних версиях — Protel 99, Protel DXP, P-CAD 2001-2006 требовалось назначать в этом поле номинальное значение параметра.)
5. В колонках **Tolerance** полей **Device** и **Lot** задать предельные отклонения параметра для индивидуальных компонентов ($\pm 2\%$) и для партий компонентов ($\pm 5\%$).
6. В колонках **Tracking #** ("отслеживание №") полей **Device** и **Lot** ввести число, которое необходимо при моделировании коррелированных разбросов

параметров нескольких компонентов: если для нескольких компонентов указана одно и то же значение, программа использует одну и ту же последовательность случайных чисел при назначении отклонений параметров всех этих компонентов.

7. В колонках **Distribution** полей **Device** и **Lot** указать вид закона распределения разбросов для компонента и для партий. Для нашего примера установим нормальное распределение (**Gaussian**) для разброса типа **Device** и равномерное распределение (**Uniform**) для разброса **Lot**.
8. Повторить действия пп. 2...7 для индивидуальной настройки шести резисторов нашей схемы.
9. Щелчком на кнопке **ОК** закрыть диалог индивидуальных настроек. Программа возвращается в диалоговое окно (см. рис. 9.34). В поле **Value** строки **Specific Tolerances** появляется сообщение **6 defined** (определено 6 настроек).
10. В полях настройки окна (см. рис. 9.33) назначить нулевое значение допуска на разброс по умолчанию для всех компонентов схемы кроме избранных.
11. Щелчком на кнопке **ОК** завершить настройки.

Результат выполнения задачи с индивидуальными настройками разброса сопротивлений резисторов схемы приведен на рис. 9.37.

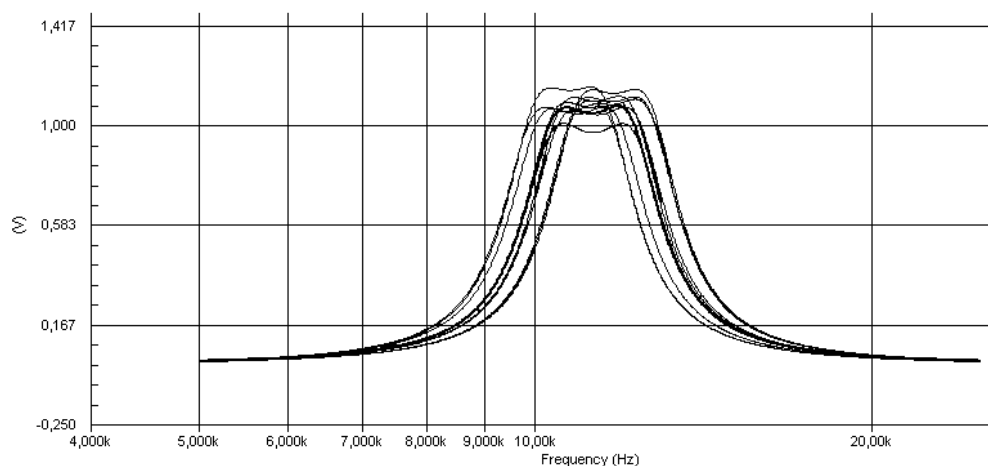


Рис. 9.37. Результаты анализа Монте-Карло: частотная характеристика полосового фильтра по рис. 9.33 при 2-процентном разбросе резисторов

Приведенные графики показывают значительно меньший разброс формы частотной характеристики по сравнению с рис. 9.34. Причины — ограниченное число компонентов, подверженных разбросу, и меньший, по сравнению с предыдущим случаем, допуск ($\pm 2\%$ против $\pm 5\%$).

Рассмотрим также пример моделирования для "худшего случая" погрешности компонентов. Для этого вернемся к настройке с 5-процентным допуском для всех компонентов по умолчанию и установим в поле **Distribution** признак **Worst Case**.

Результаты моделирования "худшего случая" для десяти прогонов задачи приведены на рис. 9.38.

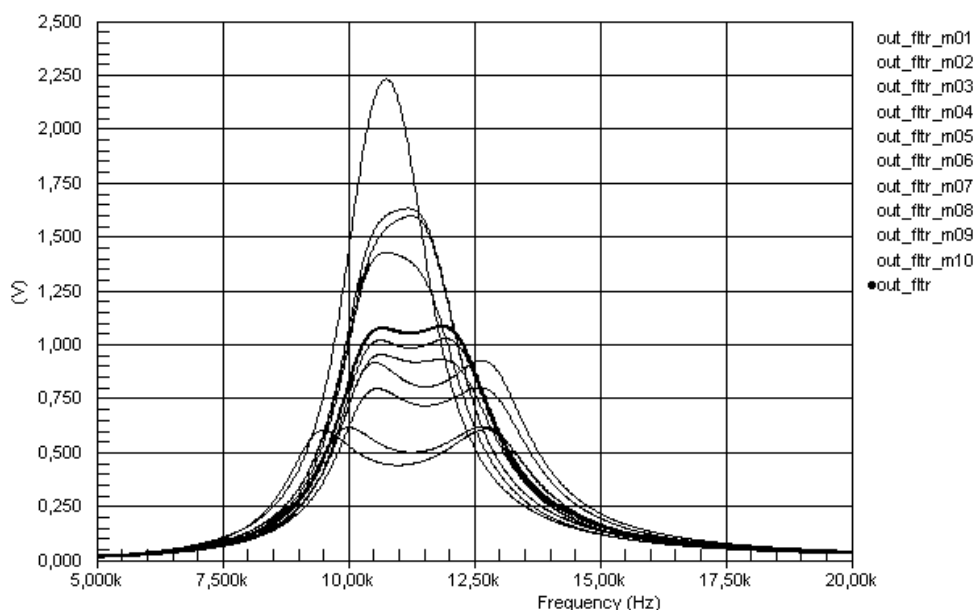


Рис. 9.38. Результаты анализа "худшего случая": частотная характеристика полосового фильтра по рис. 9.33

Видим значительно больший разброс формы частотной характеристики фильтра, чем при нормальном и равномерном распределении разброса параметров компонентов.

При моделировании методом статистических проб большой интерес представляет постпроцессорная обработка, которая позволила бы составить таблицу или построить гистограмму распределения таких показателей выходной

функции цепи, как ширина полосы пропускания, крутизна склона, значение коэффициента передачи и т. п. К сожалению, Altium Designer не располагает такими средствами, в отличие от широко известных систем моделирования OrCAD PSpice и MicroCAP.

9.1.13. Моделирование шумовых характеристик

Данный вид анализа позволяет построить функцию спектральной плотности шума на выходе цепи или в каком-то ее узле и определить вклад отдельных компонентов в суммарный выходной шум. Источниками внутреннего шума считаются резисторы и полупроводниковые приборы. Внешний, входной шум порождают источники сигнала и питания. Конденсаторы, катушки индуктивности и зависимые источники напряжения и тока считаются идеальными, свободными от собственного шума.

Рассмотрим процедуру настройки и выполнения анализа собственных шумов на примере масштабирующего усилителя (рис. 9.8). Откажемся от параметрического анализа и укажем фиксированные значения сопротивлений резисторов в цепи обратной связи: присвоим значения $R2 = 1 \text{ кОм}$ (1k) и $R3 = 100 \text{ кОм}$ (100k). Амплитуде напряжения источника входного сигнала V1 присвоим значение 1 В.

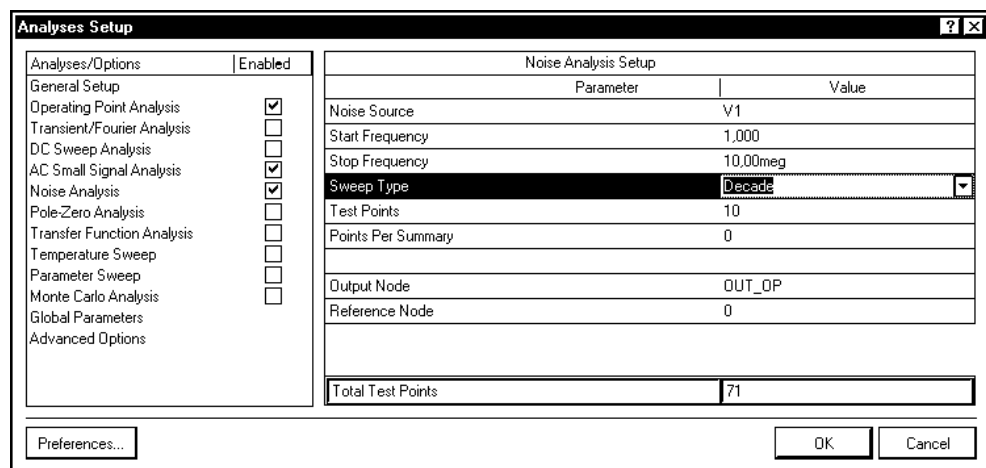


Рис. 9.39. Окно настройки режимов моделирования: анализ шумовых характеристик усилителя по рис. 9.8

Анализ шумов настраивают на панели **Noise Analysis Setup** диалогового окна настроек (рис. 9.39).

Границы частотного диапазона и его разбиения на шаги настраивают так же как при анализе частотной характеристики. В дополнение к ним нужно выполнить следующие установки:

- ☐ **Noise Source** — выбрать источник входного шума из раскрывающегося списка источников шума;
- ☐ **Output Node** — задать выходной узел, для которого выполняется расчет;
- ☐ **Reference Node** — назначить узел, относительно которого измеряется напряжение в выходном узле (по умолчанию это "земля" — узел 0);
- ☐ **Points per Summary** — при указании значения "0" (ноль) расчет выполняется только для выходного узла; при указании "1" вычисляются вклады отдельных компонентов в выходной шум.

Программа вычисляет значения спектральной плотности шума в заданном числе точек и строит графики спектральной плотности входного и выходного шумов. Спектральная плотность имеет размерность $\text{В}^2/\text{Гц}$, т. е. это мощность на единицу полосы частот, выделяющаяся на сопротивлении 1 Ом.

Полученные результаты приведены на рис. 9.40.

Программа выводит два графика — спектральной плотности шума в назначенном выходном узле **NO(out_or)** и спектральную плотность шума, приведенного ко входу усилителя **NI(out_or)**. В пределах полосы пропускания цепи спектральная плотность выходного шума, приведенная ко входу, определяется из соотношения: $\text{NI} = \text{NO}/K^2$, где K — коэффициент передачи цепи по напряжению или по току, в зависимости от того, исследуются ли передаточные свойства цепи по напряжению или току. Показанные на графиках замеры, выполненные электронными курсорами, подтверждают это соотношение: при коэффициенте усиления $K = 100$ отсчет $3,3901 \cdot 10^{-6} \text{ В}^2/\text{Гц}$ на графике спектральной плотности выходного шума и $334,27 \cdot 10^{-12} \text{ В}^2/\text{Гц}$ на графике приведенного шума отличаются приблизительно в $100^2 = 10\,000$ раз.

Следует отметить, что расчет спектральной плотности шума, приведенного ко входу, программа выполняет правильно, только если при настройке параметров амплитуде входного сигнала **AC Magnitude** присвоить значение 1 В. При других значениях коэффициент пересчета выходного шума ко входу меняется обратно пропорционально квадрату отношения амплитуды входного сигнала к значению 1 В.

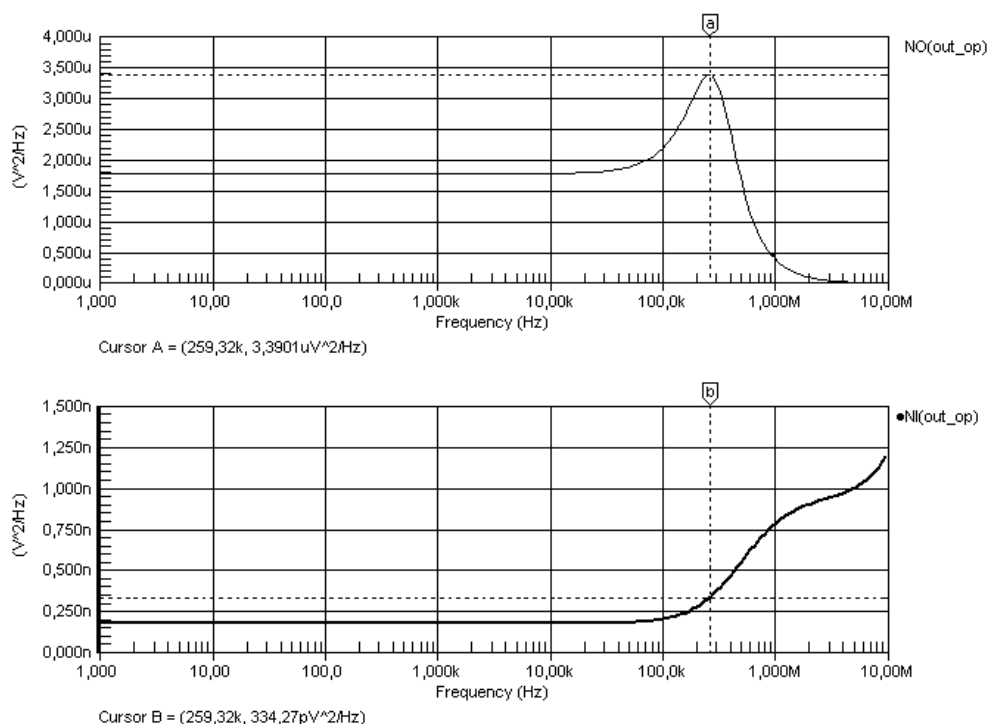


Рис. 9.40. Результаты анализа шумовых характеристик: спектральная плотность выходного шума и шума, приведенного ко входу усилителя

9.1.14. Моделирование вариаций температуры

Характеристики цепи можно проанализировать в условиях изменения температуры компонентов. По существу отличие его от рассмотренного нами параметрического анализа состоит в том, что вариации подвергается не значение основного электрического параметра компонентов, а параметров его SPICE-модели, характеризующих зависимость основных параметров компонента от температуры. Рассмотрим данный вид анализа на примере входной вольт-амперной характеристики биполярного транзистора.

Известно, что при фиксированном значении тока базы напряжение на переходе база – коллектор кремниевого биполярного транзистора с изменением температуры изменяется приблизительно на $-2 \text{ мВ}/^\circ\text{C}$. Попытаемся подтвердить этот факт моделированием. Для этого вернемся к исследованной ранее схеме для построения вольт-амперных характеристик транзистора КТ315А

(см. рис. 9.5) и изменим настройки режима анализа по постоянному току **DC Sweep** (см. рис. 9.7).

Для этого нужно.

1. Отключить использование вложенного цикла и подвергнуть схему вариации единственного воздействия — тока базы.
2. Указать в поле **Primary Source** источник базового тока I1.
3. Указать пределы и шаг вариации базового тока:
 - **Primary Start** — начальное значение тока базы 0.000;
 - **Primary Stop** — конечное значение тока базы 500u (500 мкА);
 - **Primary Step** — шаг вариации тока базы 5.000u (5 мкА)
4. Открыть панель общих настроек **General Setup** и назначить активный сигнал, подлежащий выводу в графики — **BASE** — при этом программа будет строить графики напряжения на базе транзистора в системе координат ток-напряжение.
5. В поле назначения режимов анализа **Analyses/Options** активизировать режим вариации температуры **Temperature Sweep** и установить параметры температурной вариации:
 - **Start Temperature** — начальное значение температуры -25°C ;
 - **Stop Temperature** — конечное значение температуры $+125^{\circ}\text{C}$;
 - **Step Temperature** — значение шага вариации 25°C .
6. Щелчком на кнопке **ОК** завершить настройки и, как это уже неоднократно описывалось, запустить выполнение задачи.

Полученные результаты моделирования приведены на рис. 9.41.

Замеры напряжения на базе транзистора, выполненные электронными курсорами на кривой **base_t3**, соответствующей температуре p - n -переходов транзистора 0°C и на кривой **base_t7** для температуры 100°C , составляющие, соответственно, 730,69 и 575,61 мВ, позволяют оценить значение температурного дрейфа входной вольт-амперной характеристики транзистора: $(V(\text{base_t7}) - V(\text{base_t3}))/100 = -1,55 \text{ мВ}/^{\circ}\text{C}$.

Полученное значение близко к известной из литературы величине температурного дрейфа $-2 \text{ мВ}/^{\circ}\text{C}$. Отличие можно объяснить тем, что в SPICE-модели транзистора значения параметров модели и их температурных ко-

эффицентов (при моделировании температурных свойств транзистора используются 10 температурных коэффициентов и 18 уравнений [6]) установлены с недостаточной точностью либо применяемые уравнения в недостаточной степени описывают температурный дрейф параметров модели.

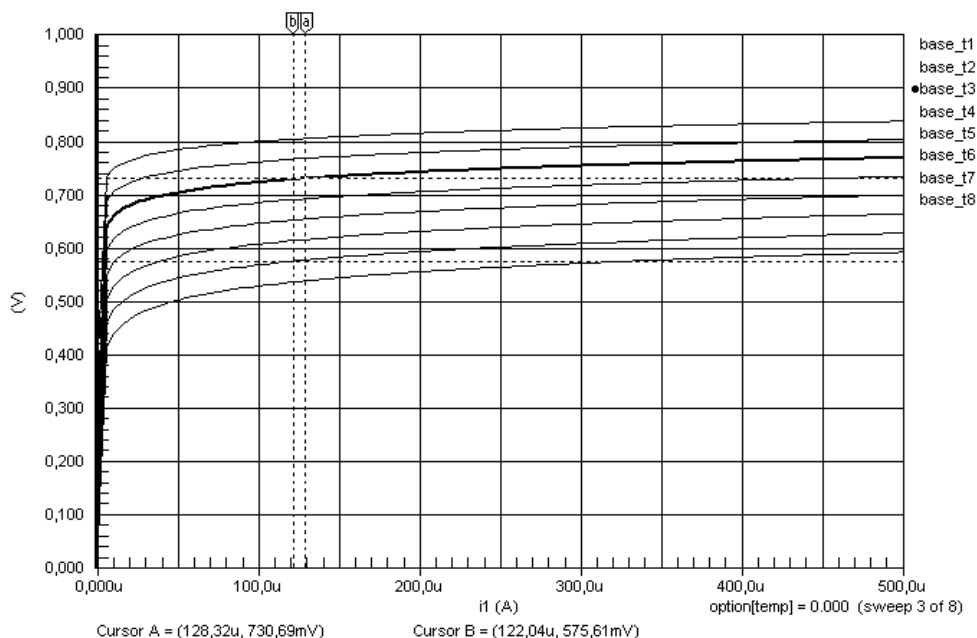


Рис. 9.41. Результат анализа влияния температуры на входную вольт-амперную характеристику транзистора по схеме рис. 9.5

9.2. Моделирование цифровых функциональных узлов

При анализе цифровых радиоэлектронных функциональных узлов перед программой моделирования стоят две основные задачи:

- определять время задержки распространения цифрового сигнала по цепочке логических вентилей или спусковых схем и разность времени распространения по нескольким разветвляющимся и сходящимся путям с целью устранения "эффекта гонок" в цифровых узлах;

- контролировать превышение нагрузочной способности — коэффициентов объединения входов и разветвления по выходу интегральных микросхем.

Ранее уже отмечалось, что, в принципе, возможно моделирование цифровых схем точно так же, как и аналоговых, используя SPICE-модели всех без исключения электрорадиоэлементов, входящих в узлы цифрового функционального узла или составляющих его интегральных микросхем. Это, однако, приводит к неоправданному увеличению объема SPICE-описаний и загрузке компьютера лишними вычислениями переходных процессов во всех без исключения узлах цифровых схем.

Кроме того, модели элементов, находящихся внутри микросхем, производители не поставляют, паспортные данные на них отсутствуют, поэтому самому составить такие модели невозможно.

Более рационально моделировать цифровые компоненты и схемы на их основе на "поведенческом" уровне. Это требует создания моделей компонентов, описывающих логику их функционирования, нагрузочную способность и задержки передачи логических сигналов.

9.2.1. Язык описания моделей цифровых компонентов

В подсистеме схемотехнического моделирования MSCS применяется специальный язык описания моделей цифровых компонентов — Digital SimCode, подобный языку программирования С. В Digital SimCode входят функции, определяющие такие параметры компонента, как задержка распространения логического сигнала, нагрузочные характеристики и др. Поведение компонента определяется на основе таблиц истинности, математических функций, операторов условного перехода, выбора (IF...Then, CASE).

Описания на языке Digital SimCode представляют собой текстовые файлы в кодах ASCII, составляются текстовым редактором и сохраняются в дисковой памяти с расширением txt.

Рассмотрим структуру описания модели на примере цифровой интегральной микросхемы ТТЛШ SN74LS74 фирмы Texas Instruments (сдвоенного D-триггера с предварительной установкой и сбросом). Микросхемы серии SN74XX имеют отечественные аналоги. В частности, рассматриваемый пример может служить для моделирования отечественной микросхемы К555ТМ2.

Описание состоит из нескольких секций¹.

- ❑ В секции 1 с выражения `# ls74 source` (идентификатора `SimCode`-функции) начинается описание модели. Идентификатор обозначает начало "исходного" (термин, принятый в программировании для обозначения текста программы на языке высокого уровня) текста описания модели компонента `SN74LS74` и используется программой для вызова модели.
- ❑ Секция 2 — объявление данных.

Ключевое слово `INPUTS` объявляет имена входных контактов одной логической ячейки компонента. Одновременно с логическими входами здесь же объявляются контакты питания и "земли" — `VCC` и `GND`.

В данном примере строка `INPUTS` имеет вид:

```
INPUTS VCC, GND, PRE, DATA, CLK, CLR;
```

Знак `;` разделяет выражения языка `SimCode`.

Ключевое слово `OUTPUTS` объявляет имена выходных контактов одной логической ячейки компонента. Обратим внимание на то, что вместе с выходными контактами в последующем списке перечисляются и входные контакты, но с суффиксом `_LD`. Это делается для моделирования влияния входных цепей микросхемы на выходы источников сигналов. В списке выходных есть также контакт питания, но отсутствует "земля".

В данном примере строка `OUTPUTS` имеет вид:

```
OUTPUTS VCC_LD, PRE_LD, DATA_LD, CLK_LD, CLR_LD, QN, Q;
```

Обозначения `QN` и `Q` принадлежат, соответственно, инверсному и прямому выходам триггера.

Ключевые слова `INTEGERS` и `REALS` объявляют, соответственно, целочисленные и вещественные переменные, используемые программой при моделировании.

Строка `PWR_GND_PINS(VCC,GND)` объявляет, какие контакты компонента используются для подачи питания и подключения к "земле" и, соответственно, какие напряжения подводятся к этим контактам.

Строка `SUPPLY_MIN_MAX(4.75,5.25)` устанавливает нижний и верхний пределы напряжения источника питания, при которых работает микросхема (из паспортных данных).

¹ Документ Digital SimCode Technical Reference TR0117 (v. 1.2) June 09, 2006.

Строка `VOL_VOH_MIN(0.2,-0.4,0.1)` объявляет минимальное значение напряжения, на которое уровень логического нуля на выходе превышает потенциал "земли" (0,2 В), а уровень логической единицы ниже напряжения питания (−0,4 В).

Строка `VIL_VIH_VALUE(1.25,1.35)` объявляет нижний и верхний пределы напряжения входного логического сигнала, при котором происходит переключение.

Строка `IO_PAIRS(PRE:PRE_LD, DATA:DATA_LD, CLK:CLK_LD, CLR:CLR_LD)` объявляет парность входов/выходов.

- Секция 3 — инициализация SimCode-функции — представляет собой логический оператор `IF (init_sim) THEN`, который выполняется однократно в начале моделирования. В этой секции устанавливаются значения таких параметров модели, как паспортные данные на микросхему, которые не изменяются под влиянием внешних воздействий. Выходам также следует присвоить необходимые логические уровни (`ONE` или `ZERO`). Текст данной секции должен заканчиваться командой `EXIT` — выход.
- Секция 4 — операторы `LOAD` и `DRIVE`, объявляющие входные и выходные параметры микросхемы — уровни логических сигналов на входах и выходах, времена задержки срабатывания при переключении из одного состояния в другое.
- Секция 5 — описание логики функционирования микросхемы. В данном примере для этого достаточно таблицы истинности `EXT_TABLE`. В моделях других компонентов логику описывают условные операторы `IF...THEN`, операторы поразрядной установки выходов логики (`STATE_BIT`), преобразования числа из двоичного кода в десятичный (`NUMBER`) и др. — всего около 40 различных преобразований.
- Секция 6 — контроль нарушений начальной установки компонента. В данном примере происходит сбой логики, если на входах начальной установки `PRE` (в отечественных микросхемах обозначается как вход `S`) и сброса `CLR` (вход `R`) оказываются одновременно два активных логических уровня (в данном случае — лог. "0"). В реальном устройстве такие ситуации приводят к нарушению работы. При выполнении моделирования остановки задачи не происходит, программа выдает предупреждающее сообщение.
- Секция 7 — это конструкции, приведенные в листинге 9.3 или 9.4.

Листинг 9.3

```
DELAY <output> [<output> ...] = <delay>;
```

Листинг 9.4

```
DELAY <output> [<output> ...] =  
CASE (<conditional expression>) : <delay>  
    CASE (<conditional expression>) : <delay>  
    [CASE (<conditional expression>) : <delay> ...]  
END;
```

Она передает в программу значения времени задержки распространения логического сигнала на обозначенные в строке DELAY выходы (<output>). Оператор DELAY выполняется один раз, и значение времени распространения передается на те из выходных контактов компонента, логическое состояние которых меняется в ходе моделирования. Значение времени распространения вычисляется на основании паспортных данных микросхемы, параметров входного сигнала и функционального описания. Контакты, логическое состояние которых не меняется, помечаются флажком NO_CHANGE, и на них время задержки не передается. Значение времени задержки не передается также на контакты, перечисленные в списке оператора LOAD (см. Секцию 4). Значение <delay> может быть вещественной константой, вещественной переменной или выражением вещественного типа.

Оператор выбора CASE <условное выражение> : <задержка> позволяет назначить одно из нескольких возможных значений времени задержки в зависимости от того, какое из условных выражений при вычислении дает результат TRUE (истинно). Если таковых в цепочке не окажется, назначается время задержки из последней строки CASE <условие> : <время>.

Кроме стандартных условных выражений в теле оператора CASE могут использоваться следующие термы, обозначающие переход микросхемы из состояния в состояние:

- TRAN_LH — из логического нуля (лог. "0") в логическую единицу (лог. "1");
- TRAN_LX — из лог. "0" в произвольное логическое состояние;
- TRAN_HL — из лог. "1" в лог "0";
- TRAN_HX — из лог. "1" в произвольное логическое состояние;

- TRAN_HZ — из лог. "1" в высокоимпедансное состояние;
- TRAN_XL — из произвольного логического в лог. "0";
- TRAN_XH — из произвольного логического в лог. "1";
- TRAN_LZ — из лог. "0" в высокоимпедансное состояние;
- TRAN_ZL — из высокоимпедансного состояния в лог. "0";
- TRAN_ZH — из высокоимпедансного состояния в лог. "1";
- TRAN_ZX — из высокоимпедансного состояния в произвольное;
- TRAN_XZ — из произвольного состояния в высокоимпедансное;
- TRAN_XX — из произвольного в другое произвольное.

Выходные контакты можно обозначать своими именами или переменными целочисленного типа, содержащими индекс выходного контакта. В одном операторе DELAY имена контактов и переменные не должны применяться одновременно.

Листинг 9.5 содержит полный SimCode-текст модели компонента SN74LS74.

Листинг 9.5

```
//=====
// Section 1 - SimCode Function Identification
# ls74 source
//1/2- 74LS74 D flip-flop Digital Simcode Model
//typical prop delay values from TI 1981 2nd edition data book
//=====
// Section 2 - Data declarations
INPUTS VCC, GND, PRE, DATA, CLK, CLR;
OUTPUTS VCC_LD, PRE_LD, DATA_LD, CLK_LD, CLR_LD, QN, Q;
INTEGERS tblIndex;
REALS tphl_val, tphl_val, ts_val, th_val, trec_val, tt_val, temp_tp,
      clk_twl, clk_twh, pre_clr_twl, ril_val, rih_val, ricc_val;
PWR_GND_PINS(VCC,GND); //set pwr_param and gnd_param values
SUPPLY_MIN_MAX(4.75,5.25); //test for min supply=4.75 and max supply=5.25
VOL_VOH_MIN(0.2,-0.4,0.1); //vol_param=gnd_param+0.2,voh_param=pwr_param-
0.4
VIL_VIH_VALUE(1.25,1.35); //set input threshold values: vil and vih
```

```

IO_PAIRS(PRE:PRE_LD, DATA:DATA_LD, CLK:CLK_LD, CLR:CLR_LD);

// Section 3 - SimCode Function Initialization

IF (init_sim) THEN
    BEGIN        //select prop delay, setup, hold, and width times
//NOTE: both ttlh and tthl are the same value
        tt_val= (MIN_TYP_MAX(tt_param: NULL, 5n, NULL));
        temp_tp= (PWL_TABLE(sim_temp: -75, -5n, 125, 5n)); //tp temperature af-
fect
        tplh_val= (MIN_TYP_MAX(tp_param: NULL, 14n, 25n)) + temp_tp;
        tphl_val= (MIN_TYP_MAX(tp_param: NULL, 20n, 40n)) + temp_tp;
        ts_val= (20n);
        th_val= (5n);
        trec_val= (5n);
        clk_twl= (25n);    //not specified - derived from fmax
        clk_twh= (25n);
        pre_clr_twl= (20n);
        //LS stdout drive IOL max=8mA @ VOL typ=0.35V:rol_param=0.35V/8mA=43.75
        //LS stdout drive IOL max=8mA @ VOL max=0.5V: rol_param=0.5V/8mA=62.5
        rol_param= (MIN_TYP_MAX(drv_param: 62.5, 43.75, NULL));
        //LS stdout drive IOS min=20mA @ VCC max=5.25V:
        roh_param=5.25V/20mA=262.5
        //LS stdout drive IOS max=100mA @ VCC
        max=5.25V:roh_param=5.25V/100mA=52.5
        roh_param= (MIN_TYP_MAX(drv_param: 262.5, NULL, 52.5));
        //LS input load IIH max=20uA @ Vin=2.7V: ril= (2.7-
        vol_param)/20uA=125k
        ril_val= (MIN_TYP_MAX(ld_param: NULL, NULL, 125k));
        //LS input load IIL max=-0.4mA @ Vin=0.4V:rih= (voh_param-
        0.4)/0.4mA=10.5k
        rih_val= (MIN_TYP_MAX(ld_param: NULL, NULL, 10.5k));
        //Icc @ 5V: 2500= 4mA/2 typical, 1250= 8mA/2 max
        ricc_val= (MIN_TYP_MAX(i_param: NULL, 2500, 1250));
        STATE Q = ONE;        // initialize output states
        STATE QN = ZERO;

    EXIT;

END;

```

```
// Section 4 - LOAD and DRIVE Statements
```

```
DRIVE Q QN = (v0=vol_param,v1=voh_param,ttlh=tt_val,tthl=tt_val);
```

```
LOAD PRE_LD DATA_LD CLK_LD CLR_LD =
```

```
(v0=vol_param,r0=ril_val,v1=voh_param,r1=rih_val,io=1e9,t=1p);
```

```
// Section 5 - Device Functionality
```

```
EXT_TABLE tblIndex
```

PRE	CLR	CLK	DATA	Q	QN
0	1	X	X	H	L
1	0	X	X	L	H
0	0	X	X	H	H
1	1	^	X	DATA	~DATA
1	1	X	X	Q	~Q;

```
LOAD VCC_LD = (v0=gnd_param,r0=ricc_val,t=1p);
```

```
// Section 6 - Tests for Device Setup Violations
```

```
IF (warn_param) THEN
```

```
  BEGIN
```

```
    IF (PRE && CLR) THEN
```

```
      BEGIN
```

```
        SETUP_HOLD(CLK=LH DATA Ts=ts_val Th=th_val "CLK->DATA");
```

```
        RECOVER(CLK=LH PRE CLR Trec=trec_val "CLK->PRE or CLR");
```

```
      WIDTH(CLK Twl=clk_twl Twh=clk_twh "CLK");
```

```
      WIDTH(PRE CLR Twl= pre_clr_twl "PRE or CLR");
```

```
    END;
```

```
  END;
```

```
// Section 7 - Output Delays/Post Events
```

```
DELAY Q QN =
```

```
  CASE (TRAN_LH) : tplh_val
```

```
  CASE (TRAN_HL) : tphl_val
```

```
END;
```

```
EXIT;
```

Двойной наклонной чертой начинаются строки комментария. Заканчивается текст описания модели командой EXIT.

Описания моделей компонентов компилируются, объединяются по сериям компонентов в файлы с расширением `scb` и помещаются в библиотечный каталог пакета. При выполнении задачи программа по имени функции извлекает из `SCB`-файла данные для нужного компонента, изменяет их, если это указано пользователем, и производит моделирование.

9.2.2. Подключение моделей цифровых компонентов

Структура описания цифрового и аналогового библиотечного компонентов идентична. Это описание состоит из условно-графического обозначения схемного символа (УГО), описания электрических контактов и связанных со схемным компонентом моделей — топологического посадочного места (Footprint), схемотехнической модели (в случае цифровых компонентов это SimCode-модель), модели Signal Integrity, трехмерной геометрической модели.

Процедура подключения схемотехнической SimCode-модели к цифровому компоненту по существу не отличается от рассмотренной ранее в *разд. 2.4*. Тем не менее специфика языка и формата описания моделей цифровых компонентов налагает определенные особенности на эту процедуру¹.

1. Командой **System** в нижнем обрамлении главного окна **Altium Designer** вызвать контекстное меню и щелчком на команде **Libraries** открыть одноименную плавающую панель (рис. 9.42).
2. Раскрыть список подключенных библиотек и выбрать в нем схемную библиотеку, к компонентам которой будут присоединяться модели. Имя библиотеки для рассматриваемых далее примеров `S_555_SIM.SchLib`. В центральном поле панели отобразится дерево структуры выбранной библиотеки.
3. Указать курсором на компонент `K555TM2`, щелчком правой кнопки мыши вызвать контекстное меню и выбрать в нем команду **Edit Component**.

Откроется главное окно редактора схемной библиотеки, в верхнем поле которого отображается схемный символ одной из логических секций редактируемого компонента, а в нижнем левом поле (рис. 9.43) — список моделей, ранее присоединенных компонентов и три командные кнопки — **Add** (Добавить), **Remove** (Удалить) и **Edit** (Редактировать).

¹ Документ *Creating and Linking a Digital SimCode Model*. Application Note AP0139 (v. 1.0) February 14, 2006.

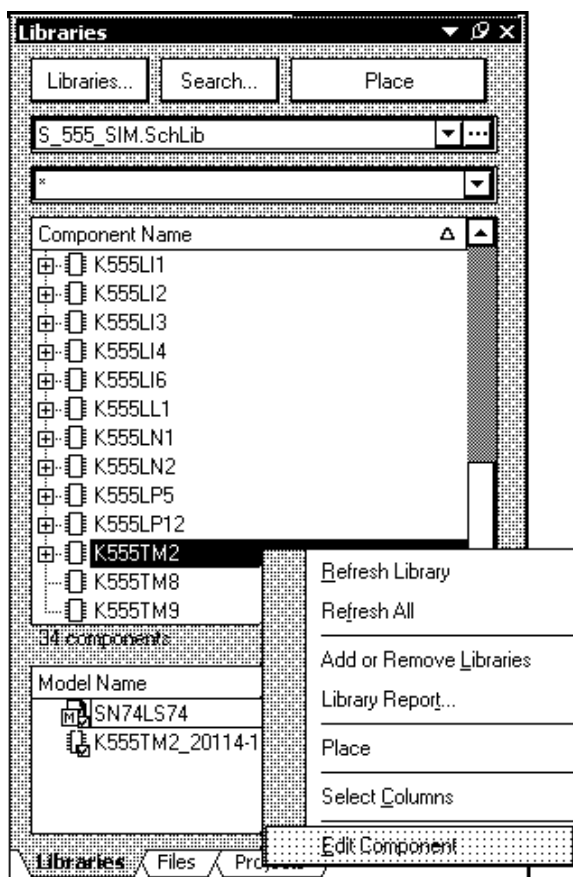


Рис. 9.42. Плавающая панель **Libraries** и контекстное меню вызова команды редактирования компонента

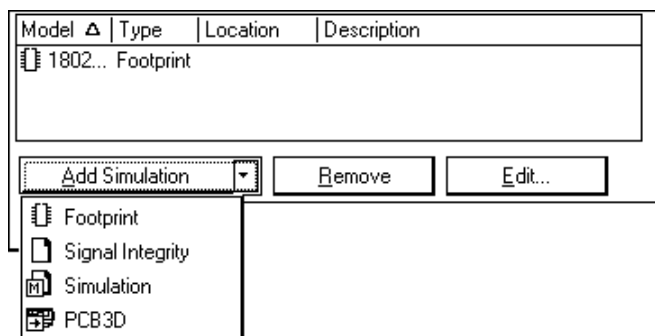


Рис. 9.43. Команды выбора моделей для редактирования

Кнопка **Add** многофункциональная: по щелчку на кнопке в правом ее конце открывается список вариантов выбора категории модели.

4. Щелчком на кнопке выбора открыть список и указать в нем категорию **Simulation**. Откроется диалоговое окно присоединения схемотехнических моделей (рис. 9.44) с пустыми полями на нижних вкладках.
5. На верхней вкладке **Model Kind**:
 - выбрать тип **General** и подтип (Model Sub-Kind) **Generic Editor**;
 - в поле **Spice Prefix** выбрать в списке префикс "A" для обозначения категории Digital SimCode-модели;
 - в поле **Model Name** ввести имя модели, присоединяемой к компоненту схемной библиотеки: для триггера K555TM2 это имя — SN74LS74;
 - в поле **Description** ввести произвольное текстовое описание модели (не обязательно);

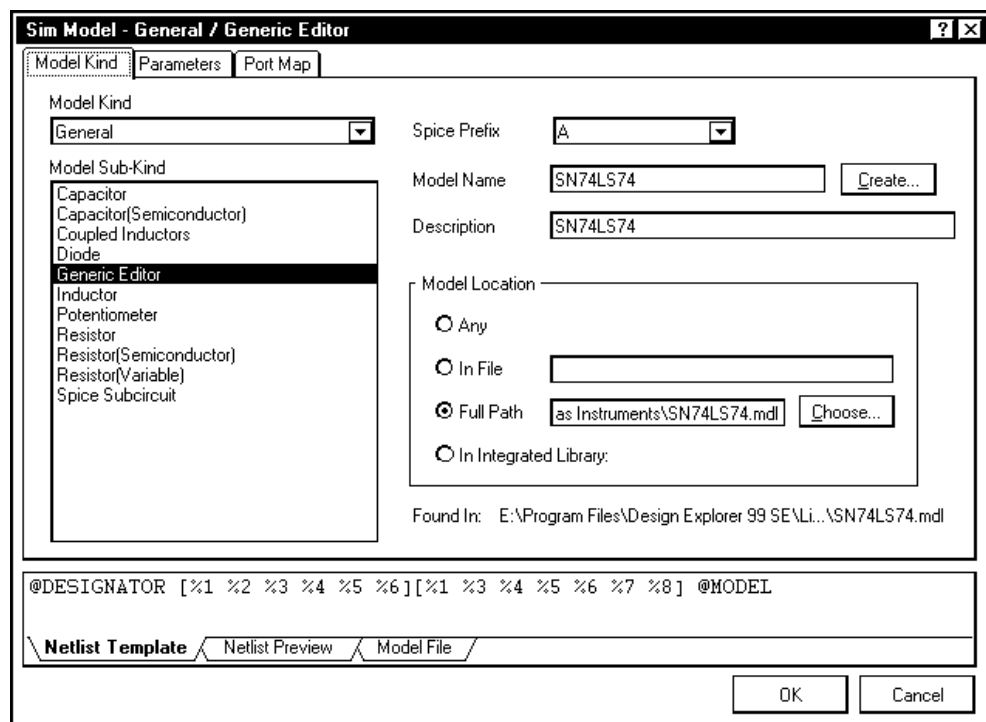


Рис. 9.44. Окно с активной панелью выбора модели цифровой микросхемы

- поле **Model Location**, из которого осуществляется поиск связи с файлом модели, остается пока замаскированным серой маской, для его активизации нужно выполнить следующий шаг.
6. Открыть нижнюю вкладку **Netlist Template**. В поле вкладки составить текстовую строку-шаблон, который будет использоваться программой при формировании NSX-файла задачи моделирования:

```
@DESIGNATOR [%1 %2 %3 %4 %5 %6][%1 %3 %4 %5 %6 %7 %8] @MODEL
```

Поля этой строки-шаблона имеют следующий смысл:

- @DESIGNATOR — формальный параметр, замещаемый при формировании задачи позиционным обозначением компонента в схеме;
- @MODEL — формальный параметр, замещаемый в тексте задачи ссылкой на модель компонента;
- [%1 %2 %3 %4 %5 %6] и [%1 %3 %4 %5 %6 %7 %8] — списки формальных параметров, замещаемых обозначениями (номера) входных и выходных выводов компонента, составленные в порядке следования обозначений выводов в Секции 2 тела SimCode-модели компонента. В нашем случае это выходы одной логической секции (одного триггера) микросхемы SN74LS74:

```
INPUTS VCC, GND, PRE, DATA, CLK, CLR;
```

```
OUTPUTS VCC_LD, PRE_LD, DATA_LD, CLK_LD, CLR_LD, QN, Q;
```

У отечественного аналога — микросхемы K555TM2 — эти обозначения имеют свои эквиваленты (рис. 9.44):

- VCC и GND — скрытые выходы питания и "земли", имеющие в цоколевке, соответственно, номера 14 и 7;
- PRE — вход \bar{S} установки в состояние лог. "1";
- DATA — вход D;
- CLK — вход тактового импульса C;
- CLR — вход \bar{R} установки в состояние лог. "0";
- QN — инверсный выход триггера \bar{Q} ;
- Q — прямой выход триггера.

7. После введения строки-шаблона становится активным поле поиска модели **Model Location**. Область поиска модели определяется активизацией одного из четырех возможных направлений:

- **Any** — во всех доступных областях памяти, где встречается указанное имя модели;
 - **In File** — поиск только в файле с указанным именем *.mdl во всех доступных областях хранения;
 - **Full Path** — в файле *.mdl, к которому указан полный путь;
 - **In Integrated Library** — в интегрированной библиотеке, из которой компонент извлекается для использования в схеме.
8. Когда файл модели найден, путь к нему и имя файла отображается на панели диалогового окна (рис. 9.43), ниже области **Model Location**. Одновременно с этим на вкладке **Model File** отображается текст файла модели (рис. 9.45).

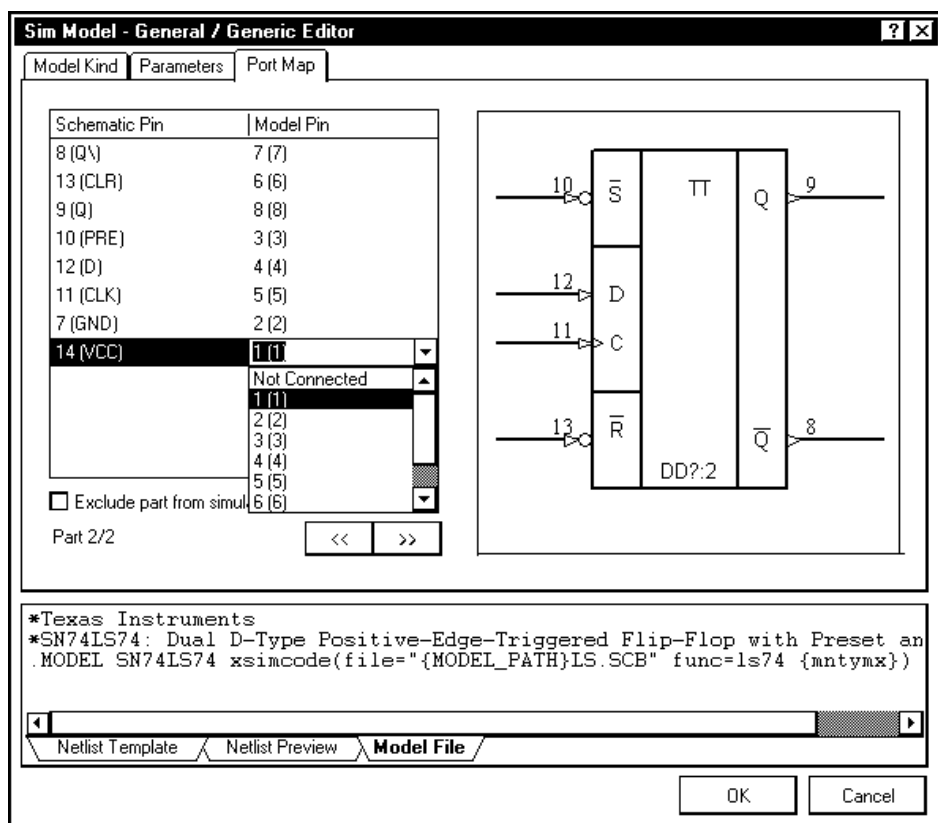


Рис. 9.45. Активная панель привязки выводов схемного компонента к контактам XSPICE-модели

Обратим внимание, что в поле диалогового окна выводится не сам текст описания модели на языке Digital SimCode, а ссылка-указатель пути к библиотеке компилированных SimCode-моделей:

```
*Texas Instruments
```

```
*SN74LS74: Dual D-Type Positive-Edge-Triggered Flip-Flop with Preset and Clear
```

```
.MODEL SN74LS74 xsimcode(file="{MODEL_PATH}LS.SCB" func=ls74 {mntymx})
```

Первые две строки — комментарий. Третья строка текста представляет собой известную по SPICE-моделированию директиву `.MODEL`, тело которой содержит следующие поля:

- SN74LS74 — имя Digital SimCode-модели компонента;
- xsimcode — обозначение типа модели, указывающее, что это модель цифрового компонента в конструкциях языка Digital SimCode;

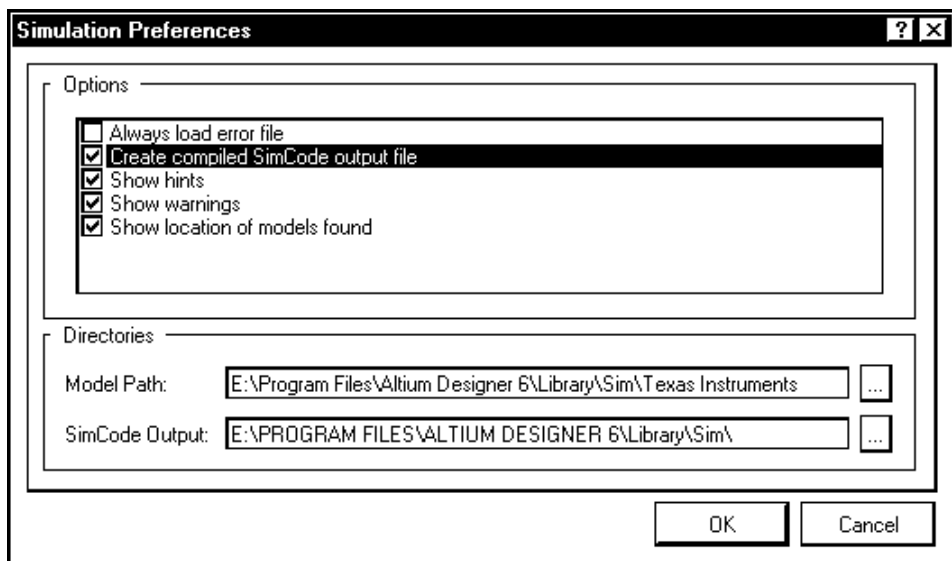


Рис. 9.46. Окно выбора пути к каталогу XSPICE-моделей

- {MODEL_PATH}LS.SCB — конструкция, являющаяся указателем пути к текстовому файлу модели *.txt или, как в данном примере, к файлу компилированной библиотеки SimCode-моделей LS.SCB. Указатель пути {MODEL_PATH} назначается в диалоге настройки режимов анализа: щелчком на кнопке **Preferences** в диалоговом окне **Analyses Setup** открывается окно **Simulation Preferences** (рис. 9.46). В поле **Model**

Path активизируется поиск пути в дереве каталогов файловой системы компьютера. Найденный путь отображается в этом поле диалогового окна. Вместо выражения `MODEL_PATH` в фигурных скобках можно указать полный путь к файлу модели;

- `func=ls74` — обозначение функции цифровой микросхемы, заимствованное из строки `# ls74 source` текста Digital SimCode-модели;
- `{mntymx}` — выражение, являющееся для программы командой передачи параметров, установленных пользователем в SimCode-модель компонента и замещающих параметры, назначенные по умолчанию.

Следующий шаг — согласование обозначений выводов схемного символа с обозначениями их в строке-шаблоне **Netlist Template**. Настройка выполняется в поле-вкладке **Port Map** диалогового окна редактирования модели (см. рис. 9.45):

- в колонке **Schematic Pin** отображаются не подлежащие редактированию цоколевочные обозначения и имена контактов схемного компонента;
- в колонке **Model Pin** для каждой пары контактов — схемного символа и модели — развернуть список обозначений контактов модели и назначить каждому контакту схемного символа парный, в том порядке, в котором записаны обозначения %1, %2, %3 и т. д. в строке-шаблоне **Netlist Template**. При этом контакты питания (VCC) и "земли" (GND) в колонке выводов модели всегда имеют обозначения 1 и 2.

- После того как будут правильно составлены все пары обозначений, в нижней вкладке **Netlist Preview** появляется строка

```
ADD? [<14> <7> <10> <12> <11> <13>]
[<14> <10> <12> <11> <13> <8> <9>] SN74LS74,
```

отображающая будущее обозначение одной логической секции компонента в NSX-файле задачи моделирования.

Система моделирования позволяет пользователю управлять значениями таких параметров SimCode-модели, как время передачи логического сигнала со входа компонента на выход, значения уровней логического сигнала, нагрузочные характеристики и ряд других. Настройка параметров выполняется в поле-вкладке окна редактирования модели (рис. 9.47). По умолчанию программа предоставляет возможность настройки значений 11 параметров модели:

- ☐ **Propagation** — время распространения логического сигнала со входа на выход. В поле **Value** устанавливается значение MIN или MAX для того,

чтобы при моделировании задавалось одно из предельных паспортных значений времени задержки, указанное в тексте файла SimCode-модели;

- ❑ **Loading** — нагрузочные характеристики по входу микросхемы. В поле **Value** устанавливаются значения MIN или MAX, чтобы при моделировании использовалось одно из предельных паспортных значений нагрузочной способности;
- ❑ **Drive** — нагрузочные характеристики по выходу микросхемы. В поле **Value** устанавливаются значения MIN или MAX, чтобы при моделировании применялось одно из предельных паспортных значений нагрузочной способности;
- ❑ **Current** — ток потребления микросхемы. В поле **Value** устанавливаются значения MIN или MAX, чтобы при моделировании устанавливалось одно из предельных паспортных значений потребляемого тока;

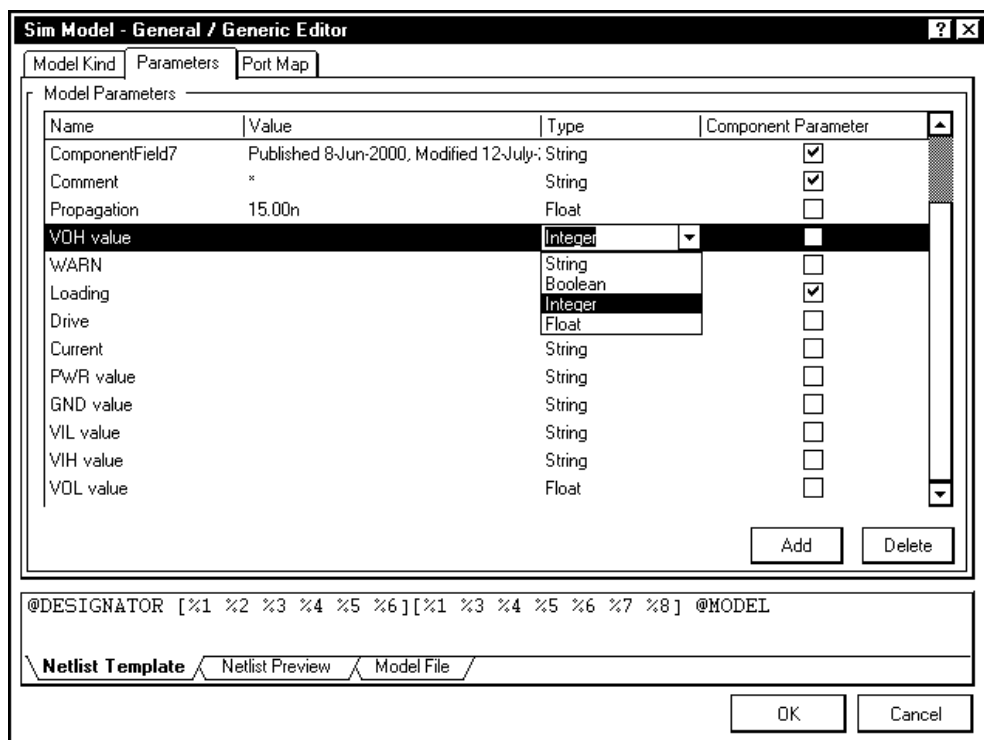


Рис. 9.47. Активная панель настройки параметров XSPICE-модели

- ❑ **PWR Value** — напряжение питания микросхемы. Значение, вводимое пользователем в поле **Value**, обладает приоритетом перед назначенными по умолчанию в описании модели. Если значение введено, необходимо также указать величину напряжения на контакте "земли";
- ❑ **GND Value** — напряжение на контакте "земли". Значение, вводимое пользователем в поле **Value**, обладает приоритетом перед назначенными по умолчанию в описании модели (ноль вольт). Если значение введено, необходимо также указать напряжение на контакте питания;
- ❑ **VIL Value** — напряжение нижнего уровня логического сигнала на входе. Обладает приоритетом над значением, установленным по умолчанию в описании модели компонента;
- ❑ **VIH Value** — напряжение верхнего уровня логического сигнала на входе. Обладает приоритетом над значением, установленным по умолчанию в описании модели компонента;
- ❑ **VOL Value** — напряжение нижнего уровня логического сигнала на выходе. Значение, установленное в поле **Value**, обладает приоритетом над значением, установленным по умолчанию в описании модели компонента;
- ❑ **VOH Value** — напряжение верхнего уровня логического сигнала на выходе. Обладает приоритетом над значением, установленным по умолчанию в описании модели компонента;
- ❑ **WARN** — предупреждение: в поле **Value** вносится значение **ON** в случае, если пользователем вводятся значения параметров модели, в то время как эти параметры определены в описании модели. При вводе недопустимых значений программа выдает предупреждения. Значение параметра по умолчанию — **OFF** (отключено).

Состав параметров определяется при редактировании модели. Кнопками **Add** (Добавить) и **Delete** (Удалить) можно добавить новые или удалить лишние параметры.

В поле **Type** панели-вкладки **Parameters** (рис. 9.47) должен быть определен тип данных для каждого параметра. Варианты выбора разворачиваются кнопкой в правом конце поля:

- ❑ **String** — текстовая строка;
- ❑ **Boolean** — логический тип: допустимые значения ON и OFF;
- ❑ **Integer** — целочисленный тип;
- ❑ **Float** — данные с плавающей точкой.

Ввод числовых значений перечисленных параметров в поле **Value** не обязателен (Optional). Если значение не введено, по умолчанию параметр принимает значение, установленное в SimCode-описании модели. Если значение введено, оно обладает приоритетом над тем, что установлено в описании модели. В большинстве случаев нет необходимости вводить "пользовательские" значения параметров.

К параметрам относится также ряд текстовых строк, несущих общие сведения о компоненте — фирма-производитель, вид логики, технология, дата выпуска модели и др.

Все рассмотренные функции редактирования моделей можно также активизировать из среды открытой схемной библиотеки по команде главного меню **Tools | Model Manager**.

9.2.3. Подготовка библиотеки компонентов

Из проведенного обсуждения становится ясно, что для создания "с чистого листа" библиотеки компонентов для схемотехнического моделирования цифровых функциональных узлов кроме приемов создания графики, назначения и описания электрических контактов, параметров компонента разработчик должен профессионально освоить язык описания моделей (фактически язык программирования) Digital SimCode. О существовании систем автоматизации программирования моделей на языке Digital SimCode, которые позволяли бы пользователю ввести паспортные данные компонента и получить его SimCode-описание, сведений нет.

Тем не менее можно создать библиотеки цифровых компонентов отечественного производства с графикой по ЕСКД и необходимыми описаниями моделей. Известно, что подавляющее большинство цифровых компонентов отечественного производства имеет зарубежные прототипы. В частности, ТТЛ/ТТЛШ-микросхемы серии SN74XX фирмы Texas Instruments — это прототипы нескольких отечественных серий (K133, K155, K555, K531, K1531, K1533 и ряда др.). Микросхемы КМОП-серии 40XX фирмы ST Electronics, серий 74С, 74АС той же Texas Instruments — прототипы нескольких серий отечественных КМОП-микросхем (K176, K561, K1561, K1554, K1564). К сожалению, далеко не все библиотеки компонентов зарубежных фирм, вошедшие в поставку Altium Designer, обеспечены связанными библиотеками SimCode-моделей. Так, применительно к микросхемам упомянутых производителей — фирм ST Electronics и Texas Instruments, включающих около 10 700 компо-

нентов, не более 1000 компонентов имеют связанные SimCode-модели¹. И еще меньше среди отечественных микросхем таких совпадений, когда они являются полными аналогами упомянутых зарубежных, да еще и обеспеченных SimCode-моделями. Так, для примеров моделирования, которые будут рассмотрены далее, пригодна библиотека S_555_Sim.SchLib отечественной серии ТТЛШ-микросхем К555. В состав библиотеки включены 33 микросхемы из полного состава серии в 107 микросхем. Больше совпадений найти не удалось.

9.2.4. Подготовка схемы и моделирование

Схему составляют по правилам, рассмотренным ранее, в *главе 4*. Особенно формируются схемы, связанные с моделированием, сводятся к необходимости включения в схему источников сигнала и питания из библиотеки Simulation Sources. IntLib. Источниками сигнала могут быть независимые источники импульсных сигналов VPULSE и VPWL, управляемые источники напряжения BVSRС, ESRC, HSRC, ключи, управляемые напряжением (VSW). Ряд более сложных источников сигнала, таких как используемые в системах моделирования DesignLab 8, OrCAD PSpice, MicroCAP зависимые источники-перемножители, формирователи цифровой фазоманипулированной последовательности в библиотеках системы моделирования Altium Designer (MSCS), отсутствуют. Такие источники придется формировать как часть общей схемы функционального узла.

При подготовке моделирования цифровых функциональных узлов необходимо активизировать в диалоговом окне настроек функцию **Advanced Options** (рис. 9.48) и установить имя цепи, относительно которой измеряются напряжения в узлах схемы **Spice Reference Net Name** (по умолчанию это цепь GND) и значения напряжений источников питания **Digital Supply VCC** микросхем ТТЛ-технологии (по умолчанию +5 В) и **Digital Supply VDD** микросхем КМОП-технологии (по умолчанию +15 В).

По правилам формирования схемных компонентов для цифровых микросхем контакты питания и "земли" могут быть скрытыми: они не показываются на схеме, а в свойствах компонента для них объявляются имена цепей, к которым они должны подключаться в схеме.

¹ Сведения из поставляемого с библиотеками САПР P-CAD 2002-2006 (той же фирмы-производителя Altium Ltd) файла-каталога библиотек Library Index.xls.

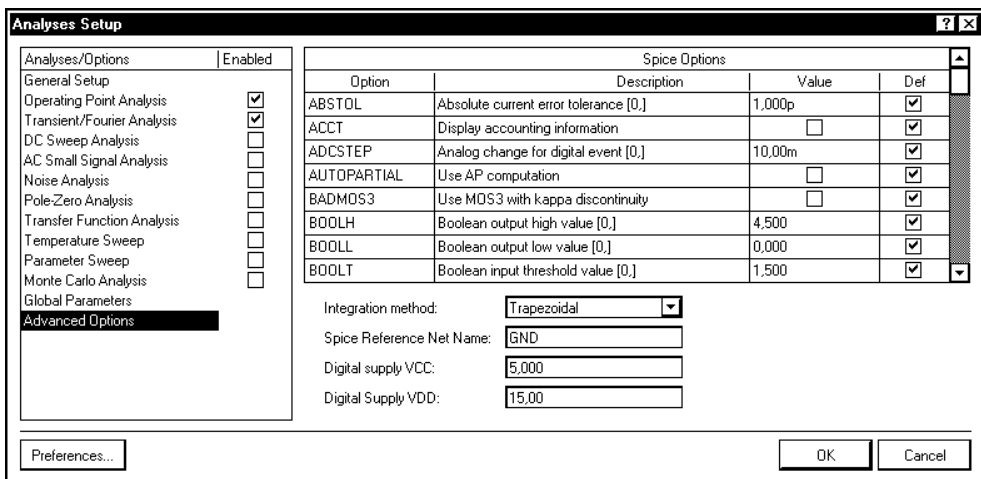


Рис. 9.48. Окно настройки режимов моделирования: назначение узлов питания и "земли" для цифровых компонентов

Если схема, подлежащая моделированию, содержит только цифровые компоненты и их сигнальные связи, включать в схему источник питания постоянного тока нет необходимости: программа моделирования подключает контакты VCC или VDD к внутренним виртуальным источникам питания.

Если же схема содержит аналоговые полупроводниковые компоненты, пассивные элементы, которые должны подключаться к источникам питания, эти источники, а также цепь GND ("земля") следует явно включить в состав схемы.

Приведем примеры описания двух цифровых функциональных узлов с использованием рассмотренных функций настройки и управления задачей моделирования.

Синхронный формирователь одиночного импульса

Формирователь построен на двух отечественных D-триггерах К555ТМ2 (рис. 9.49).

Формирователь работает следующим образом.

На вход D первого триггера подается постоянное напряжение высокого логического уровня (лог. "1").

На вход С первого триггера (рис. 9.50, график **in**) в произвольный момент времени поступает перепад логического сигнала с источника импульсного сигнала U2, переводящий контакт С из состояния лог. "0" в состояние "лог. "1".

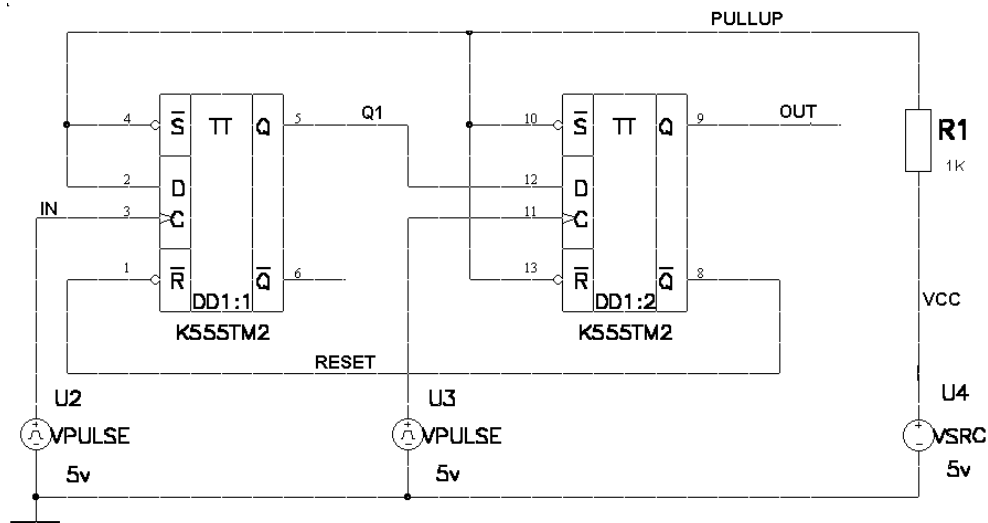


Рис. 9.49. Схема формирователя одиночного синхронного импульса

По фронту входного сигнала на входе С состояние входа D первого триггера передается на выход Q (рис. 9.50, график **q1**): триггер переходит в состояние лог. "1".

Высокий уровень сигнала передается на вход D второго триггера (контакт 12DD1).

На тактовый вход С второго триггера поступают импульсы тактовой серии (рис. 9.50, **clk**) с источника импульсного сигнала U3. По нарастающему фронту первого тактового импульса, совпадающего с высоким логическим уровнем на входе D второго триггера (DD1:2), выход Q этого триггера переходит в состояние лог. "1" (рис. 9.50, **out**).

Сигнал лог. "0" с инверсного выхода второго триггера поступает на вход сброса R (контакт 01DD1) первого триггера и сбрасывает его в исходное состояние (лог. "0" на выходе Q). Таким образом, напряжение сигнала на входе D первого триггера приобретает нижний уровень (лог. "0").

По ближайшему нарастающему фронту импульса тактовой серии состояние лог. "0" со входа D второго триггера передается на его выход Q. Таким обра-

зом, высокий уровень логического сигнала на выходе Q второго триггера удерживается в течение одного такта импульсов тактовой серии. В результате формируется одиночный импульс, синхронизированный с импульсами тактовой серии. Длительность импульса равна одному периоду импульсов тактовой серии.

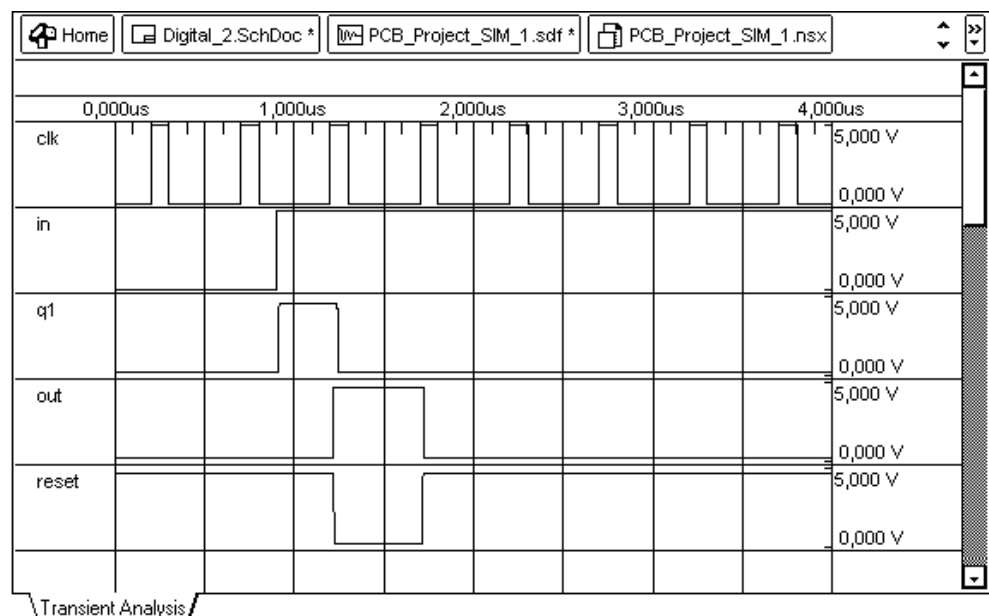


Рис. 9.50. Временные диаграммы сигналов в формирователе по рис. 9.49

В качестве источников импульсов тактовой серии и асинхронного входного логического сигнала используются источники импульсного сигнала типа VPULSE. Можно также выбрать источники импульсного напряжения других типов: независимые источники напряжения в виде кусочно-линейной функции времени VPWL, зависимые (управляемые) источники напряжения типа ESRС (ИНУН), HSRC (ИНУТ).

Источник напряжения U4 типа VSRC вырабатывает постоянное напряжение +5 В для подачи потенциала лог. "1" на вход D первого триггера для обеспечения логики функционирования формирователя, а также на входы начальной установки триггеров схемы "в единицу" (S) и "в ноль" (R). Последнее диктуется руководством по применению интегральных микросхем ТТЛ.

При запуске задачи моделирования программа автоматически генерирует текстовый файл задания на моделирование в формате входного языка XSPICE программы моделирования MSCS.

В соответствии с требованиями алгоритма моделирования XSPICE программа добавляет к именам цепей, назначенным на схеме, суффиксы \$AD — узлам, к которым подключены контакты компонента из списка INPUTS, суффиксы \$DA — парным со входными контактам из списка OUTPUTS и суффиксы \$DV — непарным узлам, к которым подключены выходные контакты компонента (также из списка OUTPUTS).

Кроме того, программа включает в текст файла задания встроенные виртуальные аналого-цифровые и цифроаналоговые буферы, так называемые Node Bridges (Мосты на узлах), модели которых отвечают за преобразование данных между алгоритмами моделирования аналоговых (SPICE 3f5) и цифровых (XSPICE) цепей. Эти буферы описываются как обычные компоненты и обозначаются именами вида ADVB1, ADVB2 и т. д. Они подключаются к тем же узлам схемы, что и контакты цифровых компонентов. Строка описания каждого буфера содержит ссылку на модель, которая подключается к задаче директивой .MODEL. Сколько и каких буферов нужно, программа определяет автоматически, анализируя схему.

Листинг 9.6 содержит текст NSX-файла задания на моделирование для данного примера.

Листинг 9.6

```
*SPICE Netlist generated by P-CAD Schematic on 10/12/2004 10:08:06
*for: K555TM2_form.nsx

*Add Node Bridge Data
ADVB1 [0 ASYNC_03DD1 CLOCK_11DD1 LOG_1 VCC][GND$AD ASYNC_03DD1$AD
+CLOCK_11DD1$AD LOG_1$AD VCC$AD] adc_mod
ADVB2 [ASYNC_03DD1$DA CLOCK_11DD1$DA LOG_1$DA VCC$DA]
+ [ASYNC_03DD1 CLOCK_11DD1 LOG_1 VCC] dac_mod
ADVB3 [D_05DD1$DV OUT_09DD1$DV R_08DD1$DV][D_05DD1 OUT_09DD1
+R_08DD1] dav_mod
.model adc_mod xadc
.model dac_mod xdac
.model dav_mod xdav
```

```

*Schematic Netlist:
ADD1:1 [VCC$AD GND$AD LOG_1$AD LOG_1$AD ASYNC_03DD1$AD
+R_08DD1$DV][VCC$DA LOG_1$DA LOG_1$DA ASYNC_03DD1$DA
+R_08DD1$DV 7 D_05DD1$DV] SN74LS74
ADD1:2 [VCC$AD GND$AD LOG_1$AD D_05DD1$DV CLOCK_11DD1$AD
+LOG_1$AD][VCC$DA LOG_1$DA D_05DD1$DV CLOCK_11DD1$DA LOG_1$DA R_08DD1$DV
+OUT_09DD1$DV] SN74LS74
VU2 ASYNC_03DD1 0 DC 0 PULSE(0.1 4.5 0.9u 1n 1n 3.3u 7u)
VU3 CLOCK_11DD1 0 DC 0 PULSE(0.1 4.5 0.2u 1n 1n 0.1u 0.5u)
VU4 VCC 0 5v

*Models and Subcircuit:
.MODEL SN74LS74
+ XSIMCODE(FILE="E:\Program Files\Design Explorer 99 SE\Library\SIM\
+LS.SCB" FUNC=LS74 )

.END

```

Последовательность действий не отличается от моделирования переходного процесса (**Transient Analysis**) в аналоговых функциональных узлах.

При настройке задачи на панели **General** (см. рис. 9.6) назначаем узлы схемы, сигналы в которых представляют интерес и должны отображаться на графиках. В лексике подсистемы моделирования MSCS эти узлы обозначаются как **Active Signals** (Активные сигналы). В качестве таковых выбираем тактовые входы С первого и второго триггеров (узлы 03DD1 11DD1 на рис. 9.49), выход Q первого, а значит, и вход D второго триггера (узел 05DD1), прямой и инверсный выходы второго триггера (узлы 09DD1 и 08DD1).

На вкладке **Transient/Fourier Analysis** устанавливаем значения полного времени моделирования 5 мкс и шага дискретизации 5 нс. Результаты моделирования приведены на рис. 9.50 и 9.51. Временные диаграммы на рис. 9.51 позволяют определить длительность задержки фронтов синхронного выходного импульса на прямом и инверсном выходах второго триггера схемы (**09dd1**, **08dd1**) относительно переднего фронта тактового импульса на входе С второго триггера (**11dd1**).

Полученные значения времени задержки соответствуют паспортным данным на микросхему K555TM2. Временные диаграммы в виде ломаных линий свидетель-

ствуют о том, что моделирование выполнялось не на "электрическом" уровне, с помощью SPICE-моделей элементов, образующих внутреннюю структуру микросхемы, а на основе "поведенческой" SimCode-модели компонента.

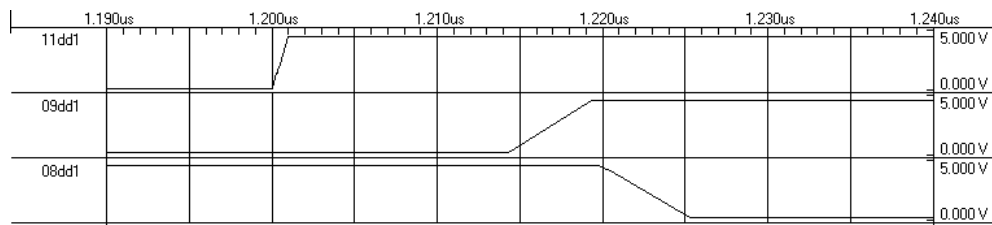


Рис. 9.51. Задержка распространения сигналов в формирователе по рис. 9.49

Запаздывание фронта сигнала на инверсном выходе триггера на 5 нс относительно прямого выхода объясняется неполной симметрией электрической схемы D-триггера и определяется временем срабатывания петли положительной обратной связи триггера.

Рассмотрим еще один пример.

Четырехразрядный реверсивный двоично-десятичный счетчик с дешифратором

Схема построена на отечественных микросхемах счетчика К555ИЕ6 и дешифратора на 10 выходов К555ИД6 (рис. 9.52).

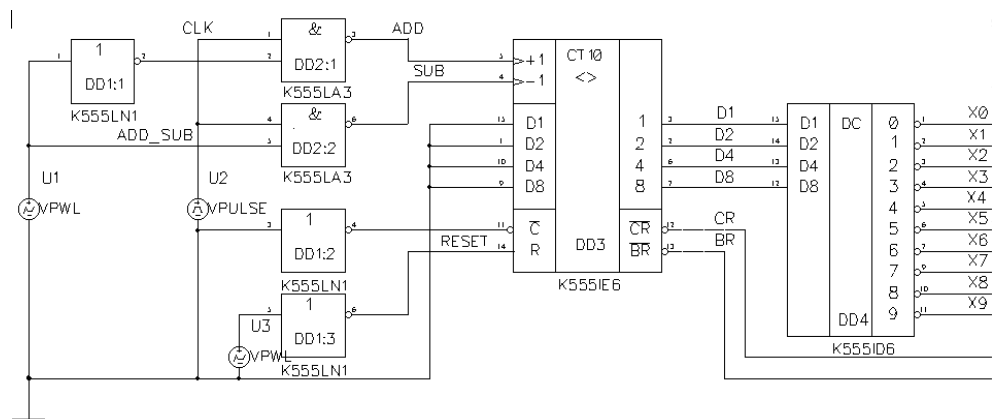


Рис. 9.52. Схема реверсивного счетчика с дешифратором

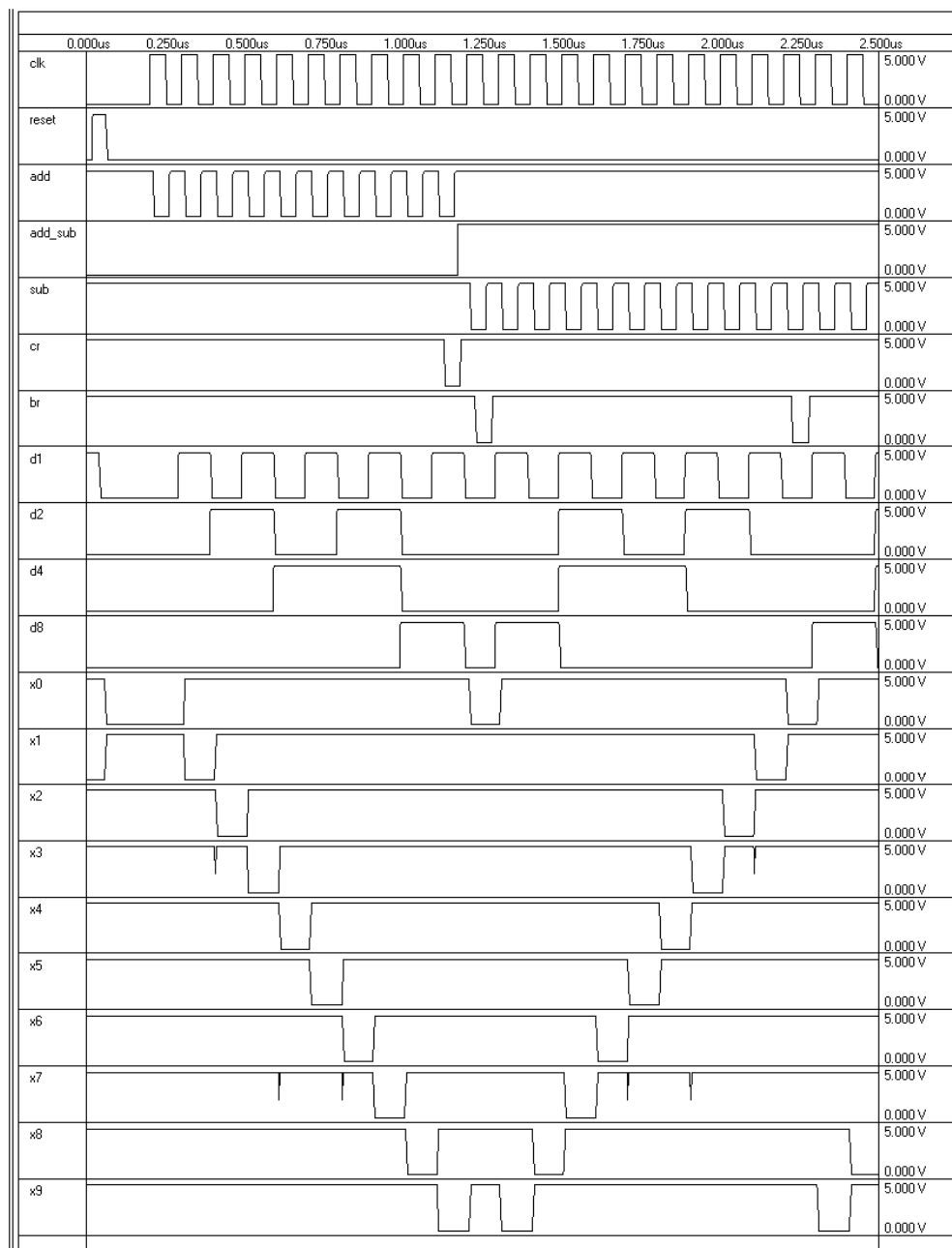


Рис. 9.53. Результаты моделирования: временные диаграммы сигналов в схеме по рис. 9.52

Поясним функционирование узла, ссылаясь на временные диаграммы, полученные в результате моделирования (рис. 9.53).

Входные сигналы вырабатывают три импульсных источника напряжения:

- с источника U3 типа VPWL на инвертор DD1.3 и далее, с его выхода, на R-вход счетчика подается импульс начальной установки (рис. 9.52, RESET);
- с источника U2 типа VPULSE на входы вентилях DD2.1, DD2.2 подаются тактовые импульсы с частотой следования 10^7 имп./с (рис. 9.52, CLK);
- с источника U1 типа VPWL непосредственно на вход DD2.2 и через инвертор DD1.1 на вход DD2.1 подается логический сигнал, переключающий подачу тактовых импульсов на входы суммирования и вычитания счетчика DD3 (рис. 9.52, ADD_SUB).

В течение первых десяти периодов импульсов тактовой серии тактовые импульсы поступают с выхода вентиля DD2.1 на вход суммирования (+1), в результате счетчик накапливает число в пределах 0...9. Соответствующий параллельный двоичный код 0000...1001 (рис. 9.53, D1, D2, D4, D8) поступает на входы дешифратора DD4.

На выходах дешифратора X0...X9 последовательно возникает и держится в течение одного периода импульсов тактовой серии логический сигнал уровня лог. "0" (рис. 9.53, X0, X1, ..., X9).

При поступлении на вход счетчика десятого тактового импульса происходит переполнение, и счетчик сбрасывается в нулевое состояние. При этом на выходе переноса вырабатывается короткий импульс переноса в старшую декаду (рис. 9.53, CR).

После прохождения через вентиль DD2.1 десяти тактовых импульсов он запирается, а отпирается вентиль DD2.2 и через него тактовые импульсы начинают поступать на вход вычитания (–1) счетчика DD3.

Поскольку счетчик в момент переключения в режим вычитания находится в состоянии 0000, сразу вырабатывается импульс займа из старшей декады (рис. 9.53, **br**), счетчик устанавливается в состояние 1001, после чего в цикле вычитания двоичный код в нем уменьшается до 0000. Логический сигнал на выходах X9...X0 дешифратора DD4 "пробегаёт" в обратном порядке.

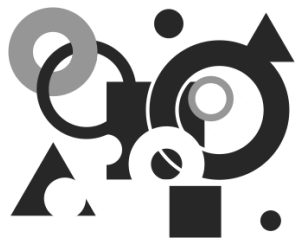
На временных диаграммах выходных напряжений дешифратора наблюдаются "спицы" (Spikes) в моменты переключения триггеров счетчика. Этот эффект характерен для асинхронных двоичных счетчиков и значительно ослаблен в синхронных счетчиках.

* * *

В заключение данной главы сравним возможности рассмотренной подсистемы моделирования Mixed-Signal Circuit Simulator с широко распространенными системами схемотехнического моделирования OrCAD PSpice и MicroCAP 7.0.

- ❑ Разработка языка описания моделей цифровых компонентов Digital SimCode облегчила моделирование смешанных аналого-цифровых устройств. Преобразование данных между алгоритмами моделирования аналоговых и цифровых схем выполняется с помощью встроенных моделей-буферов.
- ❑ В отличие от OrCAD PSpice (DesignLab 8) допустимо выполнение параметрического анализа, вариации температуры и анализа Монте-Карло в одном задании, хотя и с ограничениями.
- ❑ Возможна вариация в одной задаче двух параметров. Это больше, чем позволяет OrCAD PSpice, но не дотягивает до возможностей MicroCAP, позволяющего варьировать до десяти параметров.
- ❑ Отсутствуют средства анализа параметрической чувствительности и параметрической оптимизации схем.
- ❑ Отсутствуют средства постпроцессорной обработки результатов многовариантного анализа (Performance Analysis).

Несмотря на отмеченные недостатки, система Mixed-Signal Circuit Simulator — это достаточно мощное средство схемотехнического моделирования, поддерживающее сквозную технологию ведения проекта.



Моделирование паразитных эффектов в печатном монтаже

С ростом сложности радиоэлектронных функциональных узлов, увеличением частоты обрабатываемых аналоговых сигналов и тактовых частот передачи информации в цифровых узлах, повышением плотности компоновки печатных плат и прокладки трасс печатного монтажа обостряется проблема обеспечения электромагнитной совместимости внутри модулей РЭС на печатных платах.

Известно, что при длине проводников, превышающей четверть длины волны передаваемого колебания, такие проводники должны проектироваться как цепи с распределенными параметрами (длинные линии). Для них необходимо определять волновое сопротивление и согласовывать его с выходным сопротивлением источников и входным сопротивлением приемников сигнала. При распространении сигнала по несогласованной линии передачи происходит интерференция падающей и отраженной волны, приводящая к искажению формы сигнала. Кроме того, при любых частотах сигналов, циркулирующих в печатных проводниках, и при любых тактовых частотах обмена цифровой информацией остается актуальным присутствие паразитных индуктивных и емкостных связей между проводниками, приводящее к паразитным наводкам в электрических цепях функционального узла.

Разработки САПР отвечают на эти вызовы поставкой все более изощренных программных средств. К ним относится интегрированный в Altium Designer модуль анализа "целостности сигналов" Signal Integrity. Программа обеспечивает быстрое интерактивное моделирование следующих эффектов в печатном монтаже:

- ☐ расчет полного сопротивления печатных проводников для цепей печатной платы, указанных при настройке;

- ❑ анализ "звона", обусловленного отражениями сигнала (Reflections) от концов печатного проводника, рассматриваемого как отрезок длинной линии;
- ❑ анализ наводок (Crosstalk), создаваемых "проводником-агрессором" на "проводники-жертвы" (термины программы Signal Integrity) и обусловленных паразитными индуктивными и емкостными связями между печатными проводниками.

Расчет основан на применении макромоделей входных и выходных буферов активных электрорадиокомпонентов (интегральных микросхем). Модели входных и выходных каскадов (буферов) активных компонентов, так называемые IBIS-модели (I/O Buffer Information Specification), поставляются производителями современной компонентной базы и подключаются к компонентам схемной библиотеки по общим правилам, рассмотренным в *разд. 2.4*. Пользователь может редактировать параметры моделей. Для работы программы не требуются SPICE-модели или другие модели аналоговых компонентов.

10.1. Предварительные замечания

Для того чтобы моделирование состоялось, необходимо выполнить предварительную настройку проекта и учесть целый ряд начальных условий¹.

1. Прежде всего, нужно настроить группу правил проектирования, относящихся к анализу целостности сигналов. Для этого в активном документе печатной платы командой главного меню **Design | Rules** следует развернуть в диалоговом окне настройки правил узел дерева правил **Signal Integrity** (рис. 10.1).

Всего настройке подлежат до 12 параметров режима анализа. Рассмотрим эти параметры по однородным группам:

- **Signal Stimulus** — источник виртуального сигнала, присоединяемый по умолчанию ко всем контактам компонентов, которым присвоен статус выходных: назначается характер сигнала (одиночный, периодический прямоугольный импульс или постоянное напряжение), а также начальный момент, длительность и период следования; назначается также высокий (High Level) или низкий (Low Level) начальный уровень импульса;

¹ Документ Performing Signal Integrity Analyses. Tutorial TU0113 (v. 1.2) December 12, 2005.

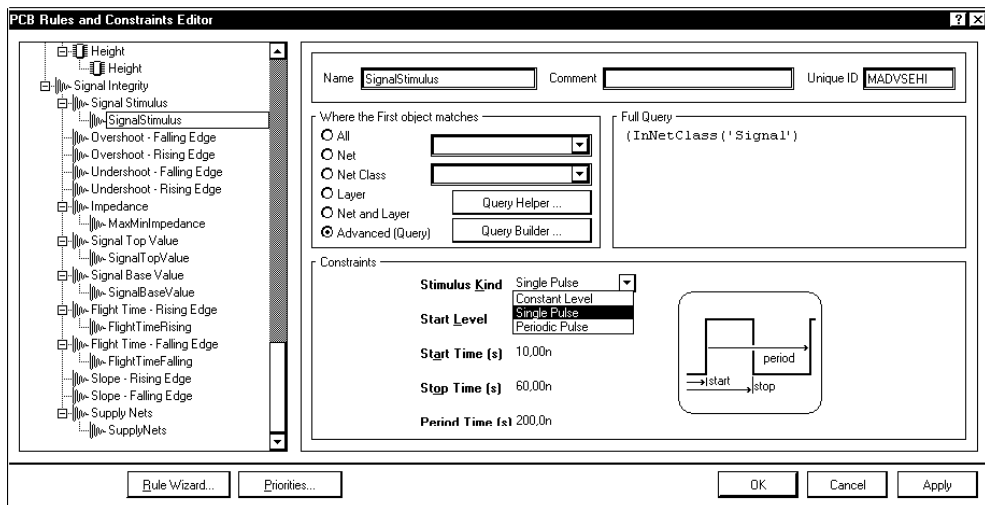


Рис. 10.1. Настройка правил для анализа Signal Integrity

- **Overshoot и Undershoot** — выбросы вверх и вниз от установившегося уровня сигнала на переднем (**Rising Edge**) и заднем (**Falling Edge**) фронтах импульса;
- **Impedance** — волновое сопротивление печатного проводника: назначается минимально и максимально допустимое значение сопротивления;
- **Signal Base Value** и **Signal Top Value** — нижний и верхний уровни сигнала в установившемся значении;
- **Flight Time** — время достижения сигналом порогового уровня срабатывания логики на переднем (**Rising Edge**) и заднем (**Falling Edge**) фронтах импульса;
- **Slope** — время изменения сигнала от порогового до гарантированного значения логической единицы или нуля на переднем (**Rising Edge**) и заднем (**Falling Edge**) фронтах импульса;
- **Supply Nets** — значения напряжений в цепях питания и "земли".

Всем рассмотренным параметрам, кроме напряжения питания, назначены значения по умолчанию: напряжение в цепи "земля" имеет значение 0 В, выбросы имеют амплитуду 1 В, длительность всех фронтов 1 нс. С помощью процедуры построения запросов (**Query**) назначенные правила можно распространить на заданные цепи или классы цепей проекта.

2. Моделирование выполняется только в том случае, если исследуемый документ — электрическая схема, печатная плата (или то и другое) *входят в структуру проекта*.
3. Моделирование может выполняться как на стадии работы с электрической принципиальной схемой функционального узла, так и на стадии проектирования печатной платы. Первый вариант, когда проект содержит только электрическую схему, имеет ограниченное применение. Он позволяет лишь приблизительно проанализировать задержки распространения и искажения формы сигналов на основании указанных при настройке значений средней длины и волнового сопротивления проводников на гипотетической печатной плате. Второй вариант, когда в состав проекта входит документ печатной платы, позволяет выполнить анализ в полном объеме, для реальных значений длины, взаимного расположения и волнового сопротивления печатных проводников. При этом анализ можно запустить как из открытого схемного документа, так и документа печатной платы.
4. Обязательное условие, позволяющее получить осмысленный результат — наличие в документах проекта как минимум одной интегральной схемы, имеющей выходной контакт (или контакты). К этим контактам программа присоединяет виртуальные источники импульсного сигнала, без чего невозможен анализ искажений. Значения параметров импульса, установленные в ветви Signal Stimulus при настройке правил, обладают приоритетом над значениями, присваиваемыми программой по умолчанию.
5. Каждый компонент нужно снабдить корректной IBIS-моделью. IBIS-модели можно присоединить на этапе формирования библиотеки компонентов. Если модель не присоединена к компоненту, вызванному из библиотеки, необходимо назначить ее в диалоге настройки на стадии подготовки схемного документа проекта к анализу паразитных эффектов.
6. Обязательно должны быть установлены правила проектирования для цепей питания и "земли". Действие этих правил (Scope) должно распространяться на соответствующую цепь или на класс цепей.
7. В состав слоев печатной платы непременно должен входить хотя бы один внутренний слой металлизации типа Plane, подключенный к цепи питания или "земли". Соответствующие Plane-слои, цепи, к которым они подключаются, характеристики межслойного диэлектрика назначаются в менеджере структуры слоев (см. разд. 3.2.5). Не допускается рассечение Plane-слоев на части. Программа игнорирует металлизацию, даже если она сплошная, но выполнена во внутренних сигнальных слоях многослойной

платы. При отсутствии Plane-слоев программа назначает соответствующие виртуальные слои, однако результатам моделирования можно доверять только в случае, если в реальной плате присутствуют внутренние сигнальные слои со сплошной металлизацией, на которые выведены цепи питания. Это связано с тем, что программа работает с печатными проводниками как с микрополосковыми линиями. Если же под проводником нет подстилающего заземленного слоя, невозможно правильно определить волновое сопротивление таких цепей, и результаты анализа не подтвердятся на практике.

10.2. Присоединение IBIS-моделей к компонентам проекта

Не все компоненты проекта обязательно имеют заранее присоединенные IBIS-модели, позволяющие выполнить анализ Signal Integrity. Модели можно присоединить непосредственно в процессе анализа либо в схемном документе проекта.

В первом случае следует запустить анализ командой главного меню **Tools | Signal Integrity**. Программа прежде всего проверяет компоненты проекта на наличие присоединенных IBIS-моделей и, обнаружив компоненты, к которым такие модели не присоединены, выводит диалоговое окно с сообщением об отсутствии моделей у части или у всех компонентов: **Not all components have Signal Integrity models set up** (рис. 10.2, окно на переднем плане).

Для присоединения моделей следует щелчком мышью на кнопке **Model Assignments** активизировать диалог назначения моделей (рис. 10.2).

В полях диалогового окна отображается список компонентов проекта, графическая иллюстрация схемного символа и посадочного места компонента, указанного курсором, его краткое описание и ссылка на библиотеку.

В анализе целостности сигнала предусмотрено семь типов компонентов: резисторы, конденсаторы, катушки индуктивности, диоды, биполярные транзисторы, электрические соединители и интегральные микросхемы. Для пассивных компонентов программа использует встроенные модели, назначенные по умолчанию. Для интегральных микросхем по умолчанию задана модель HC — High-Speed CMOS (высокоскоростная КМДП-технология).

Чтобы выбрать другой технологический стандарт, нужно щелчком правой кнопки мыши открыть контекстное меню и указать в нем команду **Change Technology** — изменить технологию. По этой команде разворачивается список из 27 аббревиатур, обозначающих общеизвестные технологические стандарты интегральных микросхем [2].

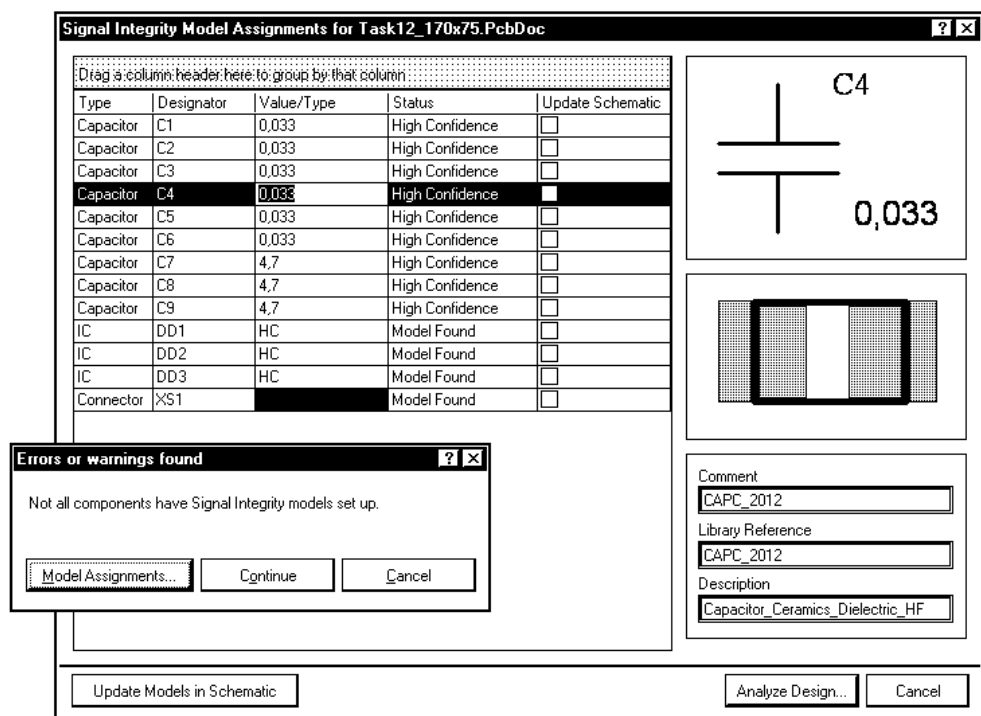


Рис. 10.2. Диалог присоединения моделей Signal Integrity

Тот же список открывается щелчком левой кнопки мыши в колонке **Value/Type** строки с обозначением любой из микросхем проекта. Приведем выдержки из этого списка применительно к некоторым из серий отечественных микросхем (табл. 10.1).

Программа анализирует правильность сделанного выбора, обозначая степень пригодности присоединенной модели в колонке **Status** таблицы в диалоговом окне (рис. 10.2).

Таблица 10.1. Технологические стандарты популярных микросхем

Аббре- виатура	Описание (англ.)	Описание (рус.)	Серия оте- чественных микросхем
TTL	Transistor-Transistor Logic	ТТЛ	133, K133, R155
S, AS, LS, ALS	Shottky, Advanced Shottky, Low Power Shottky	Структура с диодами Шотки (ТТЛШ)	K555, 1531, 1533, KP1533
CMOS	Complementary MOS	КМДП	K176, 564, K561
HC, AHC	High Speed CMOS Advanced High Speed CMOS	Быстродействующая КМДП	1554, K1554
LVC, ALVC	Low Voltage CMOS Advanced Low Voltage CMOS	Низковольтная КМДП	1564, KP1564
F	Fast	Быстродействующие	500, K500

Программа анализирует правильность сделанного выбора, обозначая степень пригодности присоединенной модели в колонке **Status** таблицы в диалоговом окне (рис. 10.2).

Всего принято семь обозначений статуса присоединенной модели:

1. **No match** — в диалоге присоединения не удалось найти признаки, позволяющие определить тип компонента.
2. **Low confidence** — тип компонента определен, но нет серьезных признаков, подтверждающих правильность выбора.
3. **Medium confidence** — средняя степень доверия к правильности выбора.
4. **High confidence** — есть серьезные основания считать выбор типа компонента правильным.
5. **Model found** — найдена модель, подходящая для данного компонента.
6. **User defined** — пользователем выбрана модель, отличающаяся от той, которую программа предлагает по умолчанию.

7. **Model added** — модель отредактирована пользователем, присоединена к компоненту и передана в схемный документ проекта.

Для редактирования выбранной модели нужно щелчком правой кнопки мыши в диалоговом окне **Model Assignment** (рис. 10.2), в строке выбранного компонента вызвать контекстное меню и указать в нем команду **Advanced**. Откроется диалоговое окно **Signal Integrity Model** (рис. 10.3).

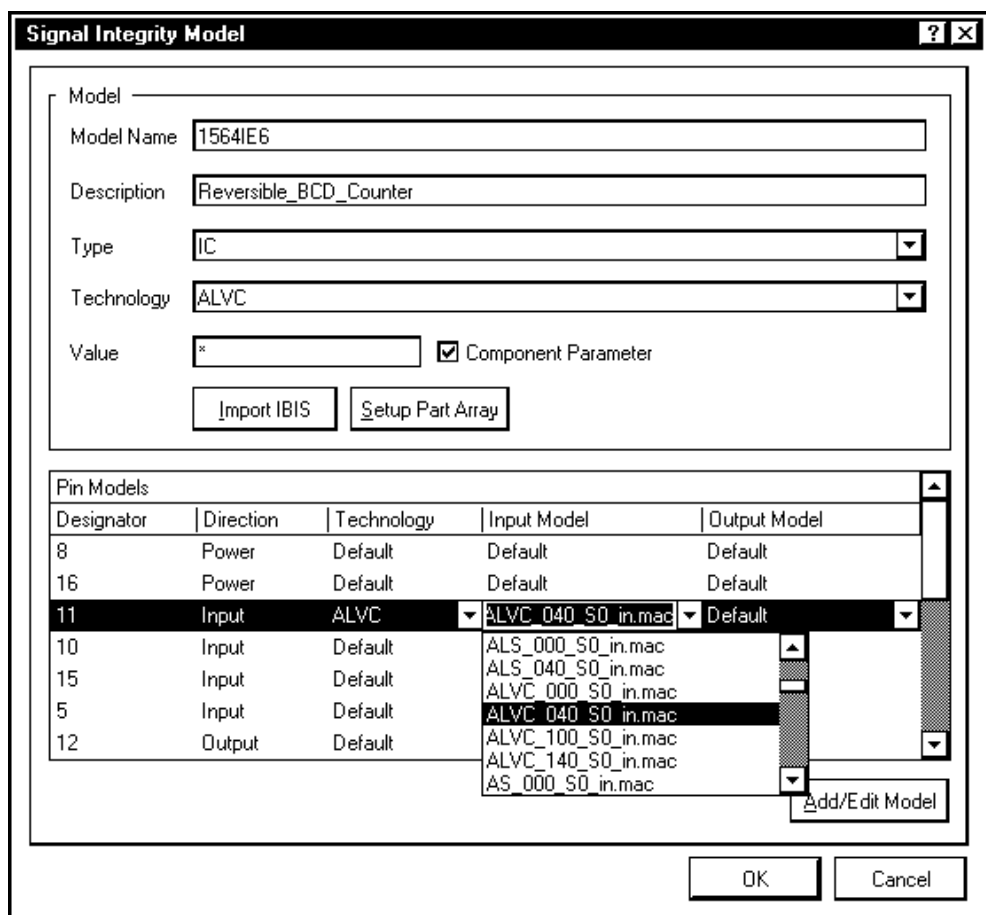


Рис. 10.3. Диалог выбора IBIS-моделей сигнальных буферов цифровых микросхем

В полях зоны **Model** назначают следующие атрибуты компонента:

- ☐ **Model Name** — имя модели;

- ❑ **Description** — описание компонента или его модели;
- ❑ **Type** — тип компонента: кнопкой в правой части поля разворачивается список из семи возможных типов и выбирается нужный;
- ❑ **Value** — значение электрического параметра (указывается для пассивных компонентов);
- ❑ **Technology** — технология: кнопкой в правой части поля разворачивается знакомый нам список технологических стандартов, из которого выбирается нужный.

Кнопка **Import IBIS** запускает стандартную процедуру поиска файлов для импорта внешних IBIS-моделей.

Кнопка **Setup Part Array**, активная только при редактировании моделей пассивных компонентов, открывает диалог составления модели для массива однородных компонентов — резисторной, конденсаторной сборки, сборки катушек индуктивности.

В нижележащем поле **Pin Models** отображается список контактов выбранного компонента, заимствованный у схемного символа. В колонках этого списка содержатся признаки каждого контакта, часть из которых можно редактировать, чтобы сформировать корректную IBIS-модель:

- ❑ **Designator** — обозначение (номер) контакта в цоколевке компонента;
- ❑ **Direction** — направление передачи сигнала через данный контакт:
 - **Power** — выводы питания и "земли";
 - **Input** — входной буфер (контакт) микросхемы;
 - **Output** — выходной буфер (контакт) микросхемы;
 - **IO** — двунаправленный, входной/выходной буфер;
 - **TRI** — выходной буфер с третьим, высокоимпедансным состоянием.

Эти признаки приходят со схемным компонентом и не редактируются в диалоге настройки IBIS-модели. Далее:

- ❑ **Technology** — технологический стандарт: щелчком в правом конце поля разворачивается знакомый нам список 27 технологий, из которого можно выбрать подходящую для каждого контакта (существуют микросхемы, у которых входные или выходные буферы выполняются в технологическом стандарте, отличающемся от остальной части микросхемы, например, АСТ- или НСТ-КМДП с ТТЛ-входами);

- ❑ **Input Model** — модель входного буфера: щелчком мыши разворачивается список из более чем 100 имен моделей — по несколько для каждого технологического стандарта;
- ❑ **Output Model** — модель выходного буфера: щелчком мыши разворачивается список также из более чем 100 имен моделей — по несколько для каждого технологического стандарта.

Щелчок мыши на кнопке **Add/Edit Model** запускает диалог редактирования выбранной модели входного или выходного буфера микросхемы. Открывается диалоговое окно **Pin Model Editor** с тремя панелями-вкладками:

- ❑ **Main** — главные сведения о модели: имя, комментарий, технология, значение напряжения питания;
- ❑ **Clamping** — ограничения: напряжения в цепи питания и "земли", динамическое входное сопротивление по цепям питания и "земли" (рис. 10.4, а);
- ❑ **Resistance/Capacitance** — значения активных сопротивлений и емкостей на контакте входного или выходного буфера (рис. 10.4, б)

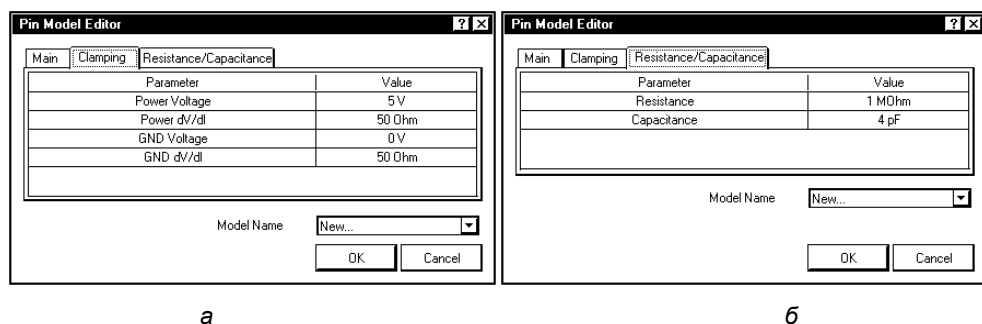


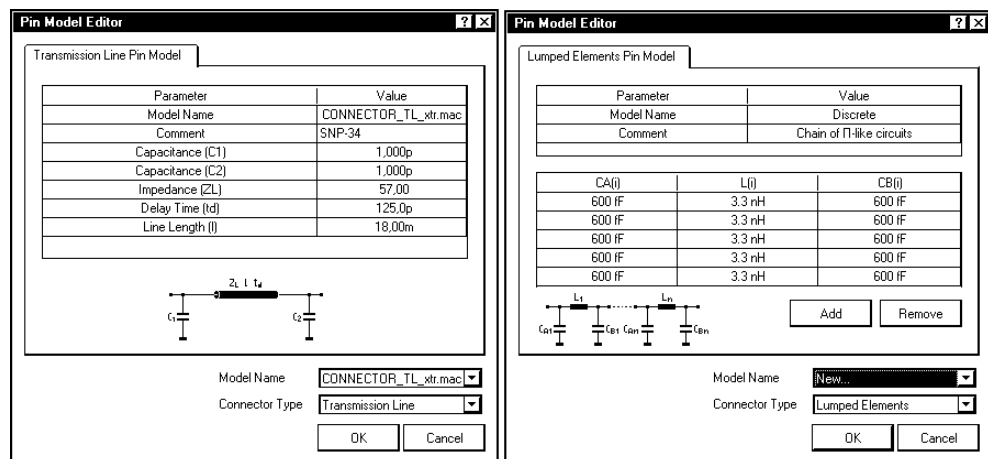
Рис. 10.4. Окно редактирования параметров IBIS-модели:

- а — назначение напряжений и сопротивлений в цепях питания и "земли";
 б — назначение параметров сигнального буфера

Остановимся на модели контактов электрического соединителя. Если цепи функционального узла проходят через контакты электрического соединителя, требуются меры по согласованию характеристических параметров контакта и волнового сопротивления линий передачи (печатного проводника на плате и провода или кабеля внешней связи функционального узла).

Программа предлагает для контактов соединителя две модели:

- ❑ **Transmission Line Pin Model** — линия передачи с заданным временем задержки, волновым сопротивлением, физической длиной и с конденсаторами малой емкости на входе и выходе (рис. 10.5, а);
- ❑ **Lumped Elements Pin Model** — линия задержки из П-образных звеньев из двух конденсаторов и катушки индуктивности в каждом звене (рис. 10.5, б).



а

б

Рис. 10.5. Панель выбора модели для контактов соединителя:

а — модель в виде линии передачи;

б — модель в виде линии задержки из П-образных звеньев

Все параметры той и другой модели настраивают в диалоговых окнах, показанных на рис. 10.5.

Процедура присоединения и настройки IBIS-моделей завершается последовательностью щелчков на кнопке **ОК** в соответствующих диалоговых окнах (см. рис. 10.3–10.5). Программа возвращается в диалоговое окно, показанное на рис. 10.2. После этого щелчком на кнопке **Update Models in Schematic** назначенные модели закрепляют за компонентами в схемном документе проекта.

Для присоединения IBIS-модели из среды открытого схемного документа следует указать курсором компонент и двойным щелчком левой кнопки мыши активизировать диалоговое окно свойств компонента (см. рис. 4.4). В по-

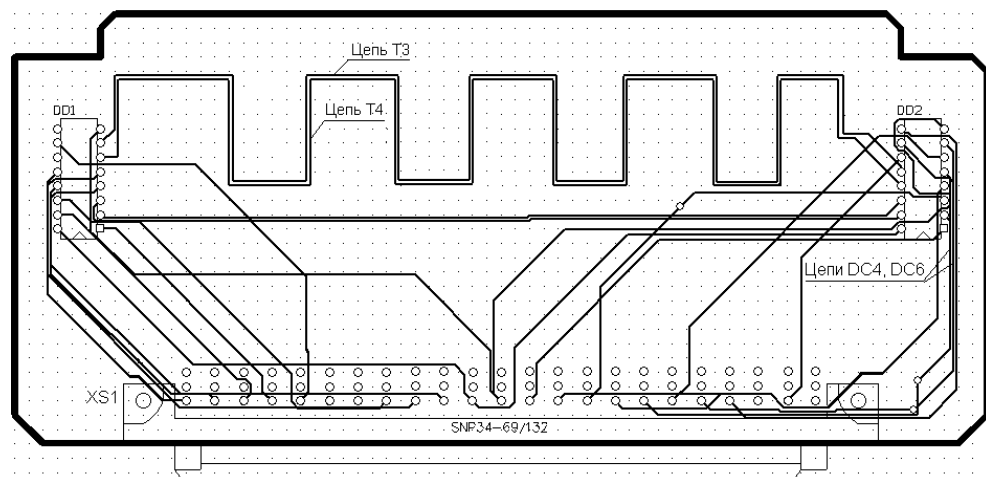


Рис. 10.7. Печатная плата функционального узла с искусственно удлиненной парой связей

Первый раз анализ запускают щелчком на кнопке **Analyse Design** в диалоговом окне присоединения моделей (см. рис. 10.2). Открывается плавающая панель **Signal Integrity**¹ (рис. 10.8).

В левой панели окна отображаются результаты первичного анализа документов проекта на нарушения назначенных правил проектирования, а также указаны выявленные разведенные и неразведенные трассы, приведен расчет длины печатных проводников и их волнового сопротивления. В англоязычной терминологии САПР P-CAD и Altium Designer такой способ отображения результатов анализа называется Screening.

В колонках **Net** и **Status** отображаются имена цепей проекта и результат DRC-контроля правил — текстовое сообщение и цветовая метка против имени каждой цепи:

- ☐ зеленая (**Passed**) — проверка выполнена, и результат в пределах допуска, установленного в правилах;
- ☐ желтая (**Not analyzed**) — проверка не проводилась, в частности не выполняется проверка цепей, выходящих на контакты соединителя;
- ☐ красная (**Failed**) — правила не выдержаны: хотя бы один результат выходит за пределы допуска, указанного в правилах.

¹ Документ Altium Designer Panels Reference. Technical Reference TR0104 (v. 2.0) June 16, 2006.

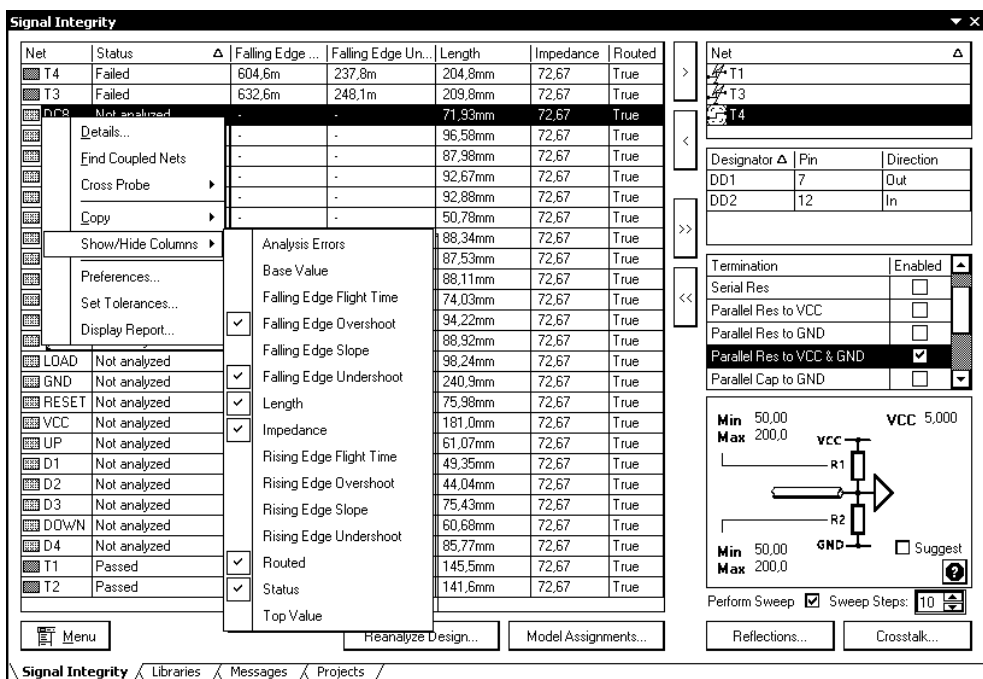


Рис. 10.8. Панель **Signal Integrity** с результатами оценки параметров печатного монтажа и команды управления отображением результатов

Кнопкой **Menu** или щелчком правой кнопки мыши можно открыть контекстное меню, выбрать в нем команду **Show/Hide Columns** (Показать/скрыть колонки) и указать результаты анализа, которые необходимо отобразить в панели (см. рис. 10.8: меню поверх таблицы). Результаты расчета длины, волнового сопротивления печатных проводников и перечень выявленных разведенных и неразведенных трасс отображаются независимо от результатов DRC-контроля.

Остановимся еще на нескольких командах контекстного меню первого уровня, показанного на рис. 10.8:

- **Details** — по этой команде выводится окно с полным списком результатов первичного анализа для выбранной цепи (рис. 10.9);
- **Preferences** — по этой команде открывается диалоговое окно настройки элементов режима анализа (рис. 10.10) с пятью панелями-вкладками:
 - **General** (Общие) — на этой вкладке нас интересует только функция назначения системы единиц измерения: для нашего проекта назначим метрическую систему — в поле **Units** активизируем кнопку **mm**;

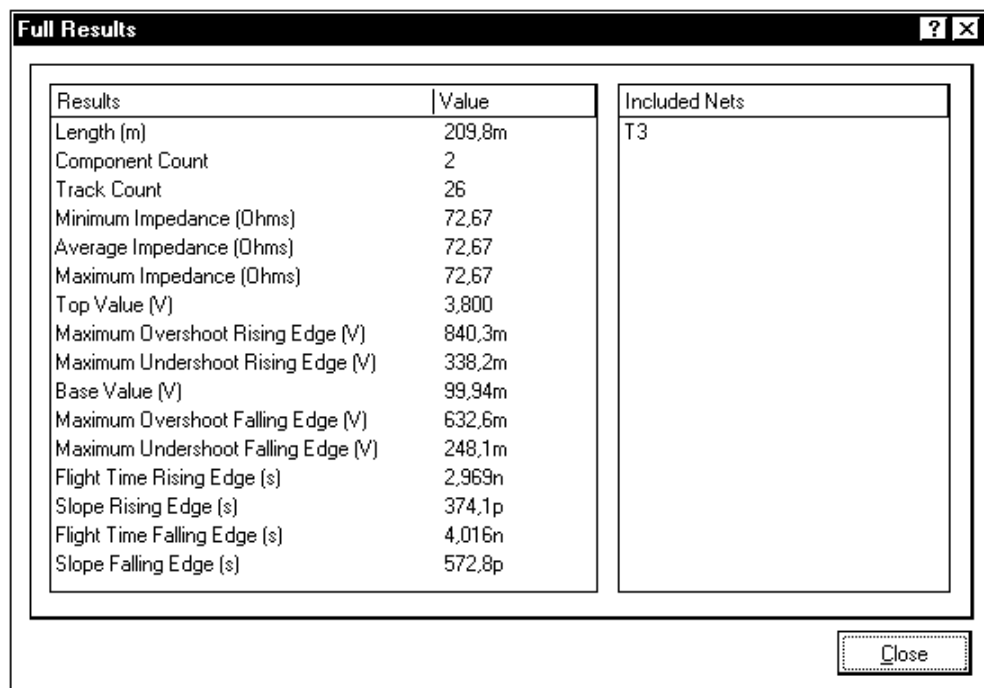


Рис. 10.9. Сводка результатов предварительного анализа выбранной цепи

- **Configuration** (Конфигурация) — на этой вкладке (рис. 10.10) задают следующие настройки:
 - ◇ **Ignore Stubs** — длину "отростков" печатных проводников, не принимаемую в расчет при анализе;
 - ◇ **Total Time** — полное время анализа;
 - ◇ **Time Step** — шаг по времени;
 - ◇ **Coupling Max Dist** — максимальное расстояние между проводниками, до которого они принимаются в расчет как связанные;
 - ◇ **Coupling Min Length** — минимальную длину связанных печатных проводников;
- **Integration** — методы интегрирования (по умолчанию назначен метод трапеций);
- **Accuracy** — точность вычислений;
- **DC Analysis** — анализ по постоянному току.

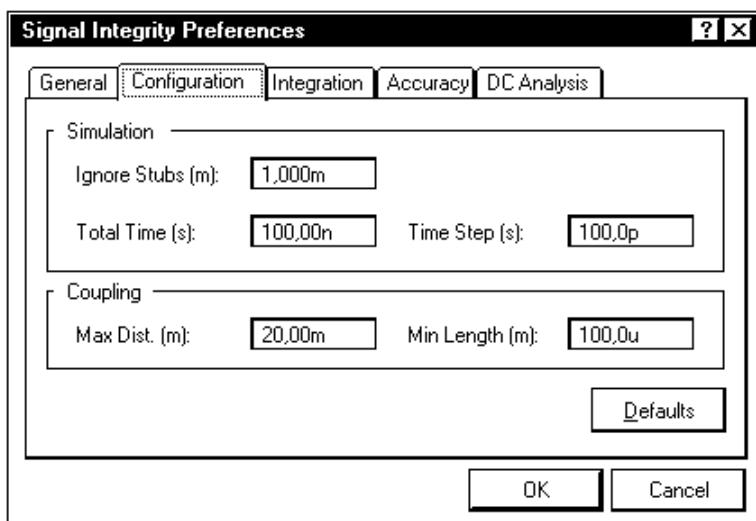


Рис. 10.10. Окно настройки параметров анализа

- ❑ **Find Coupled Nets** (Найти связанные цепи) — по этой команде в левой половине панели **Signal Integrity** выделяются темно-синей цветовой подсветкой связанные цепи печатной платы, отвечающие значениям параметров, назначенным в секции **Coupling** диалога **Signal Integrity Preferences** (см. рис. 10.10);
- ❑ **Cross Probe To Schematic/To PCB** — по этой команде цепь, выбранная в панели **Signal Integrity**, выделяется подсветкой в схеме или на плате.

Правая половина панели **Signal Integrity** служит для управления анализом отражений сигнала при прохождении его по трассам печатного монтажа (**Reflections**) и взаимных наводок с одних печатных проводников на другие (**Crosstalk**). Функции управления сгруппированы следующим образом:

- ❑ в поле **Net** составляется список цепей, представляющих интерес с точки зрения анализа отражений и наводок. Для составления этого списка необходимо указать в левом поле строки таблицы с именами необходимых цепей (по одиночке или группой), удерживая клавишу <Ctrl>, и щелчком на кнопке "стрелка вправо" образовать список имен цепей в поле **Net**;
- ❑ при указании курсором одной из цепей в поле **Net** в нижележащей таблице отображается список контактов компонентов, которые соединяет выбранная цепь: отображается позиционное обозначение компонентов (**Des-**

ignator), номера контактов этих компонентов (**Pin**) и направление передачи сигналов через контакты (**Direction**);

- в поле **Termination** располагается список из восьми способов подключения виртуальных согласующих элементов к входным или выходным IBIS-буферам микросхем (рис. 10.11):
 - **No Termination** — отсутствие согласующих элементов;
 - **Serial Res** — последовательный резистор;
 - **Parallel Res to VCC** — параллельный резистор к питанию;
 - **Parallel Res to VCC & GND** — параллельные резисторы к цепям питания и "земли";
 - **Parallel Cap to GND** — параллельный конденсатор к "земле";
 - **Res and Cap to GND** — последовательная RC-цепочка к "земле";
 - **Parallel Shottky Diodes** — диоды Шоттки параллельно, к цепям питания и "земли";
- в нижележащем поле (см. рис. 10.11) представляется пиктограмма, отображающая выбранный способ подключения согласующих элементов и численные значения параметров этих элементов, которые могут быть изменены непосредственно в диалоге, вводом с клавиатуры. Кнопка подсказки открывает текстовый комментарий к каждому выбранному способу согласования.

ЗАМЕЧАНИЕ

Следует отметить, что, в отличие от модуля Signal Integrity P-CAD 2000-2006, где можно назначить способ согласования индивидуально для каждого контакта компонентов, в Altium Designer выбранная согласующая цепочка присоединяется ко всем цепям проекта. При этом в случае, если в качестве согласующего элемента указан последовательный резистор, он присоединяется ко всем выходным контактам микросхем в цепях, образующих список в поле **Net** панели **Signal Integrity**. Все остальные виды согласующих цепочек программа присоединяет к входным контактам в цепях этого списка.

В отличие от P-CAD 2000-2006, где можно проанализировать искажения в цепи, содержащей последовательно соединенные выходной буфер микросхемы на печатной плате, печатный проводник, контакт соединителя, соединительный кабель и входной буфер микросхемы на дальнем конце кабеля, в Altium Designer искажения, вносимые соединительными кабелями, не анализируются.

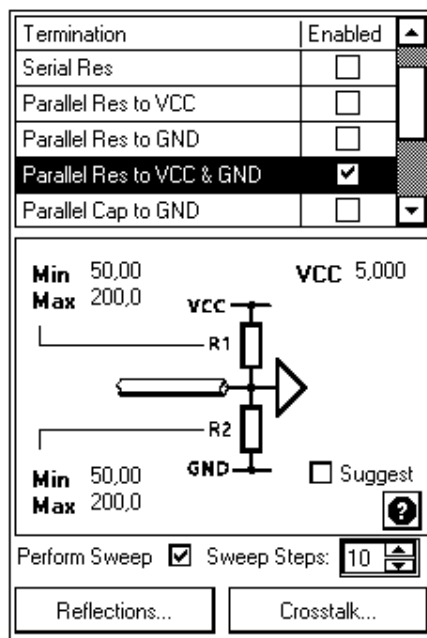


Рис. 10.11. Панель настройки режимов согласования

Рассмотрим процедуру и результаты анализа отражений и взаимных наводок для приведенного примера проекта печатной платы (см. рис. 10.7). Печатная плата имеет следующие характеристики:

- ☐ геометрические размеры 170×75 мм;
- ☐ четыре слоя: из них два наружных сигнальных слоя и два внутренних Plane-слоя сплошной металлизации, подключенные к цепям VCC и GND;
- ☐ толщина межслойного диэлектрика 0,32 мм;
- ☐ диэлектрическая проницаемость материала $\epsilon = 4,8$;
- ☐ ширина печатных проводников и зазоров между ними 0,25 мм;
- ☐ волновое сопротивление печатных проводников 72 Ом.

Микросхемы DD1 и DD2 K555ИЕ6 и K555ИД6 выполнены по технологии ТТЛШ (LS).

В список цепей для анализа включим две искусственно удлинненные цепи из параллельно расположенных сегментов с именами T3 и T4. Длина печатных проводников 205 и 208 мм.

В качестве элементов согласования назначим виртуальные согласующие цепочки из двух резисторов сопротивлением 140 Ом каждый — к цепи питания VCC и "земли" GND (см. рис. 10.11). Программа присоединяет эти цепи к входным буферам микросхемы DD2 — к контактам 12 и 13.

10.3.1. Моделирование отражений

Назначим в качестве цепи, подлежащей анализу, цепь T4. Анализ отражений запускается щелчком мыши на кнопке **Reflections** в панели **Signal Integrity** (см. рис. 10.8, 10.11). Программа анализирует форму сигнала на выходе 7 микросхемы DD1 и на входе 12 микросхемы DD2 в отсутствие и при наличии согласования. На рис. 10.12 результаты анализа приведены в виде графиков.

Рядом с графиком указаны имя цепи, контакт микросхемы и аббревиатура, обозначающая условия согласования, в нашем случае No Term — отсутствие согласования и RVG — резисторы к питанию и к "земле".

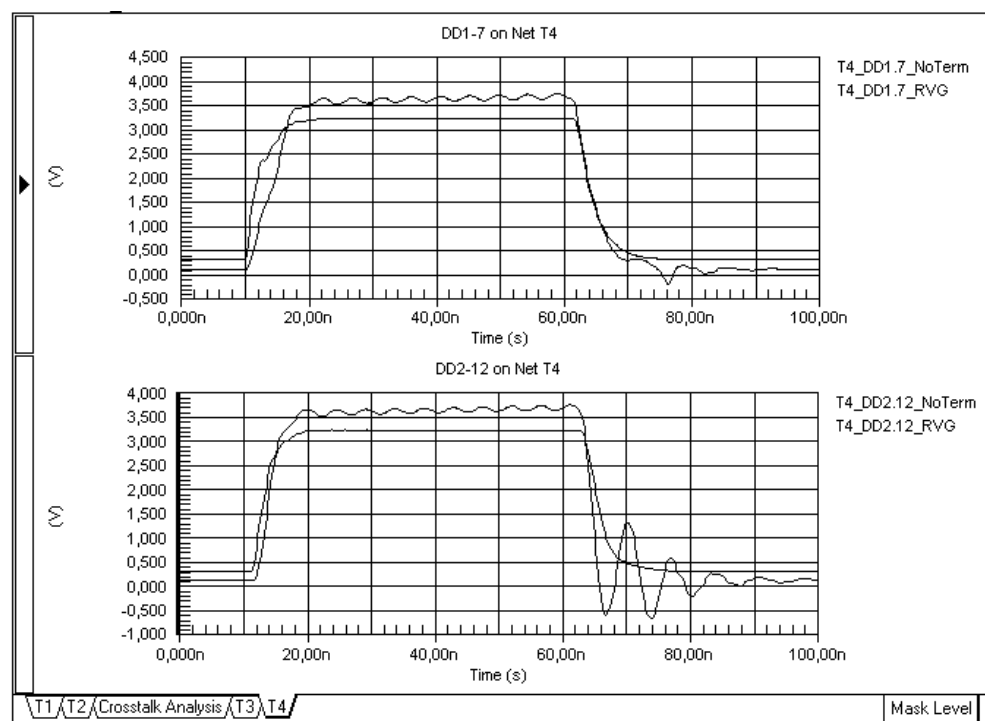


Рис. 10.12. Результаты анализа отражений на передающем и приемном концах микрополосковой линии (печатного проводника)

На нижнем графике, соответствующем отсутствию согласования, видно значительное искажение формы сигнала на плоской части импульса и в особенности "звон" по окончании действия импульса. При наличии согласующей цепочки искажения сигнала заметно сглаживаются.

Средства работы с графиками те же, что и при схемотехническом моделировании. Щелчком на имени графика он активизируется (выделяется утолщенной линией). После этого щелчком правой кнопки мыши можно вызывать контекстное меню с командами активизации электронных курсоров, редактирования, удаления кривой и т. д.

Используя электронные курсоры, можно установить такие сведения о цепях печатного монтажа, как время распространения сигнала по печатному проводнику от выхода микросхемы-источника до входа микросхемы-приемника импульсного сигнала. Так, в нашем случае это время по результатам замера курсором составляет 1,57 нс. Такой результат представляется правдоподобным: время распространения электромагнитных колебаний в свободном пространстве на расстояние 1000 мм составляет приблизительно 3,3 нс, а в микрополосковой линии это время пропорционально величине $\sqrt{\epsilon}$, т. е. примерно в 2,2 раза больше. При длине печатного проводника 200 мм время распространения должно составлять 1,45 нс.

На верхнем графике видно также, что при отсутствии согласования волна, отраженная от приемного конца линии передачи, приходя на передающий конец, заметно искажает форму импульса на контакте 7 микросхемы DD1 (график T4_DD1.7_No Term).

На рис. 10.13 приведены результаты анализа отражений при вариации значений сопротивления резисторов согласующей цепи.

Вариация параметров элементов согласующей цепи выполняется, если в панели **Signal Integrity** установлена опция **Perform Sweep** (см. рис. 10.11).

В зоне пиктограммы с изображением согласующей цепи отображаются пределы изменения сопротивления согласующих резисторов, а ниже пиктограммы, в поле **Sweep Step**, выбирается число шагов вариации параметра. На рис. 10.13 показаны значения пределов и шага вариации, назначенные по умолчанию: 10 шагов в пределах от 50 до 200 Ом. При внимательном изучении графиков видно, что наименьшие искажения формы импульса получают при значении сопротивлений обоих резисторов 140 Ом. Это естественно: результирующее сопротивление, подключенное к входному буферу микросхемы DD2, равно сопротивлению параллельного соединения верхнего и нижнего резисторов, т. е. 70 Ом. Это значение ближе всего к волновому со-

противлению печатного проводника, составляющему, по результатам предварительного анализа, 72,6 Ом (см. рис. 10.9).

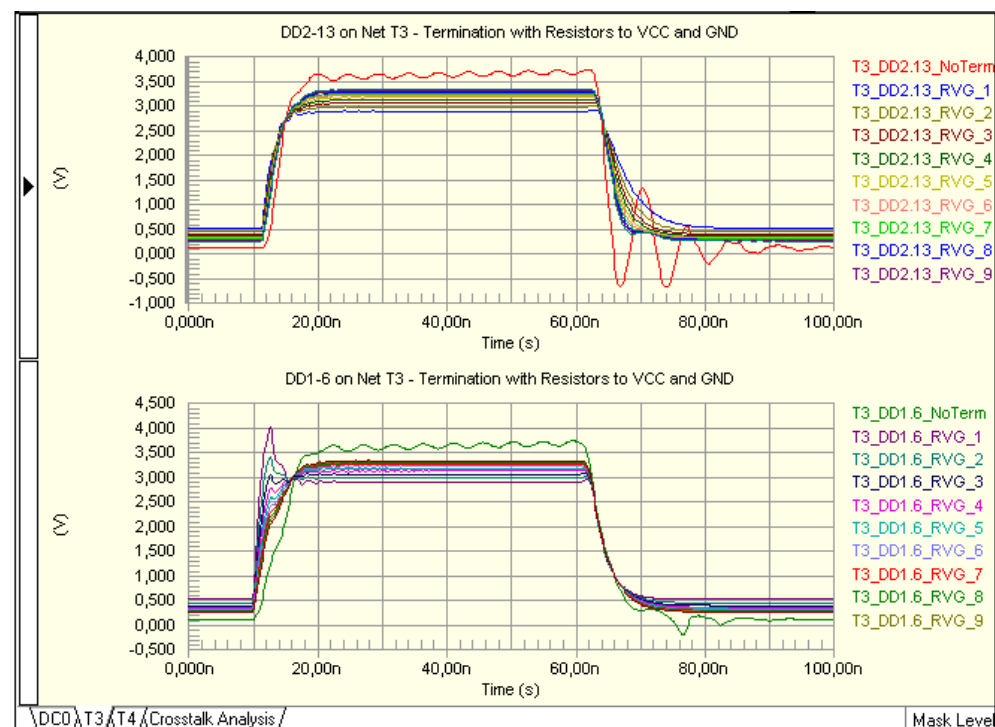


Рис. 10.13. Результаты анализа отражений при вариации параметров согласующей цепочки на приемном конце

10.3.2. Моделирование взаимных наводок

Перед анализом взаимных наводок необходимо задать цепь-источник наводки и цепи-приемники наводки. Для этого нужно выполнить еще одну специфическую настройку:

- ☐ указать курсором одну из цепей в списке **Net** панели **Signal Integrity** (рис. 10.14);
- ☐ щелчком правой кнопки мыши открыть контекстное меню и выбрать в нем команду **Set Aggressor** (Назначить агрессора) или **Set Victim** (Назначить жертву).

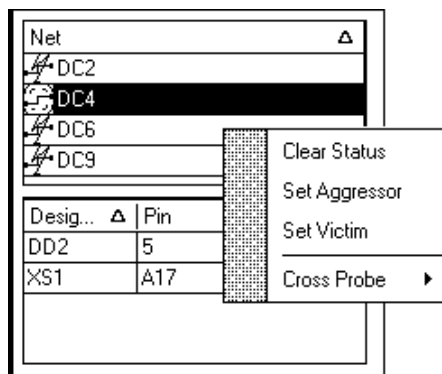


Рис. 10.14. Присвоение цепи статуса "агрессора" — источника наводок

В системе моделирования Altium Designer установлены следующие правила взаимодействия цепей "агрессора" и "жертвы":

- ☐ агрессором или жертвой можно назначить единственную цепь из списка в поле **Net**;
- ☐ при объявлении одной цепи агрессором все остальные цепи списка автоматически становятся жертвами, т. е. приемниками наводки;
- ☐ от выходных буферов микросхем цепей-жертв автоматически отключаются виртуальные источники импульсного сигнала;
- ☐ при объявлении одной цепи жертвой все остальные цепи списка автоматически становятся агрессорами, т. е. цепь-жертва принимает на себя наводки сразу от всех цепей-агрессоров.

На рис. 10.15 приведены результаты моделирования наводки, создаваемой цепью T3 нашего примера на цепь T4.

Верхние графики показывают напряжение наводки на передающем (T4_DD1.7), нижние — на приемном конце (T4_DD2.12) при отсутствии (признак `_No Term`) и наличии (признак `_RVG`) согласования. Согласование волнового сопротивления трассы печатного монтажа выполнялось, как и при анализе отражений, на входе микросхемы-приемника сигнала DD2, цепочкой из двух резисторов сопротивлением 140 Ом к питанию и "земле". Видим, что наводка на согласованную линию значительно (в 4...5 раз) меньше по амплитуде и полностью свободна от "звона".

На рис. 10.16 показаны результаты исследования взаимных наводок в печатных проводниках двух цепей, DC4 и DC6, идущих с выходов дешифратора DD2 на контакты соединителя XS1 (см. рис. 10.6 и 10.7). Длина печатных

проводников этих цепей 85,7 и 89,8 мм, ширина 0,25 мм, волновое сопротивление 72 Ом, расстояние между проводниками 0,25 мм.

Цепью-агрессором объявлена цепь DC4, цепью-жертвой автоматически назначается цепь DC6. По обоим цепям сигналы с выходов дешифратора поступают на контакты электрического соединителя XS1.

Результаты моделирования в отсутствие согласования показывают значительный "звон" на фронтах импульса как на передающем конце линии, так и на контакте соединителя.

Согласование с помощью последовательного резистора в выходной цепи микросхемы DD2 приводит к сильному ослаблению наводки и почти полному устранению "звона", особенно у контакта соединителя (нижний график DC6_XS1.A19_SerR). Полученные результаты соответствуют теоретическим положениям, сформулированным в теории цепей с распределенными параметрами.

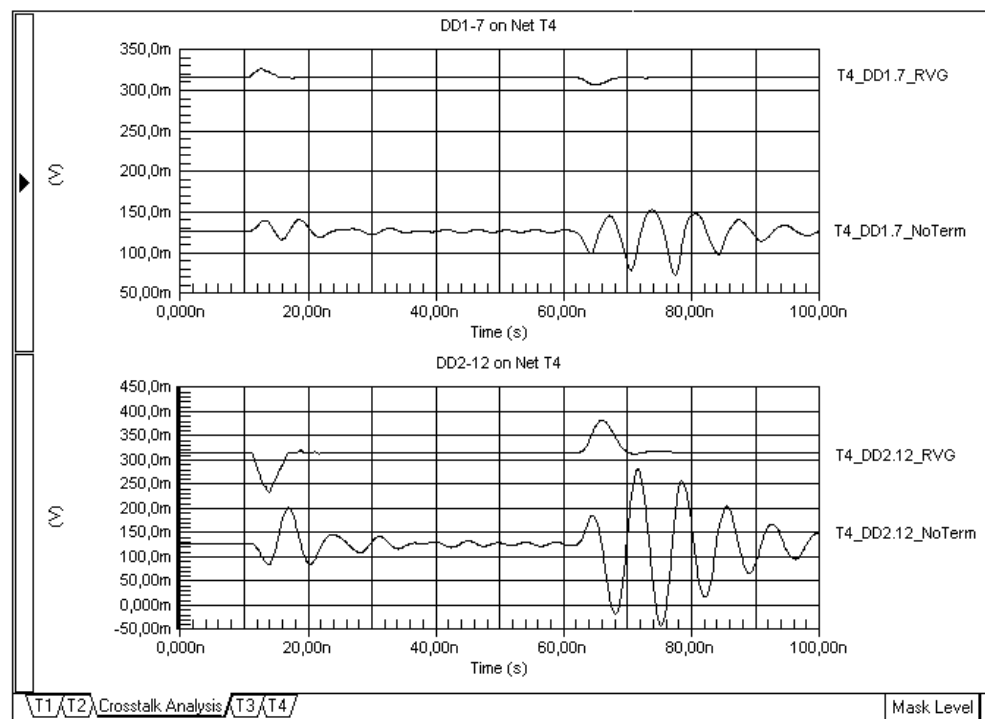


Рис. 10.15. Результаты анализа: наводка на передающий и приемный концы цепи-жертвы

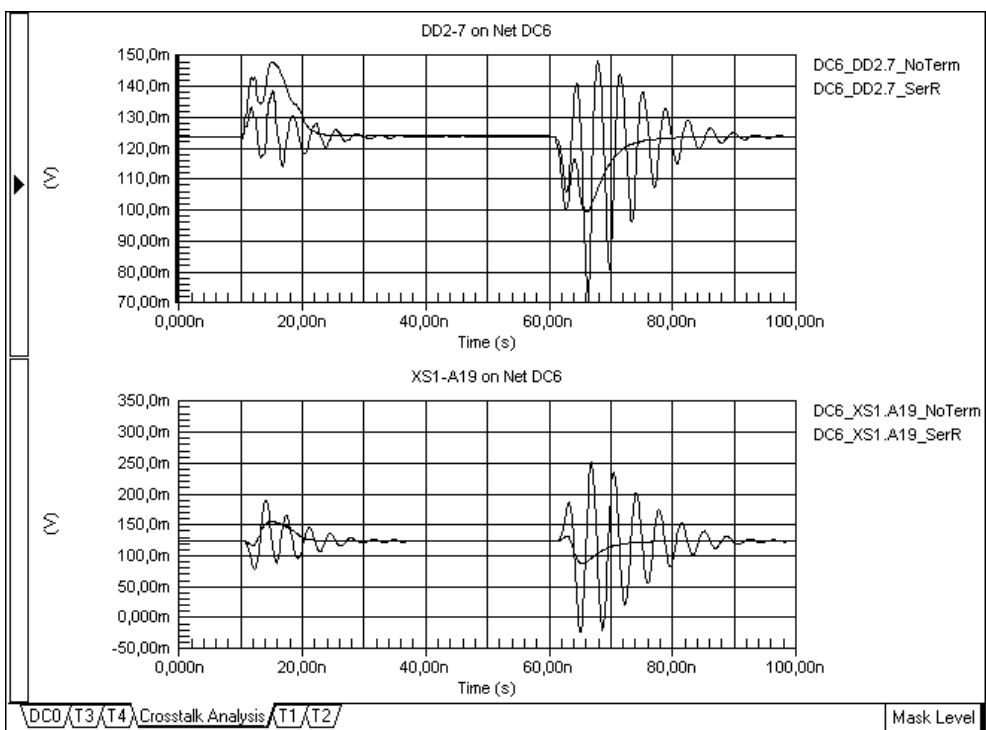
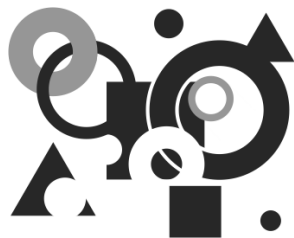


Рис. 10.16. Результаты анализа: наводка на передающий и приемный концы цепи-жертвы, отдающей сигнал на контакт соединителя

Проведенное рассмотрение позволяет утверждать, что целесообразно так планировать эксперимент, чтобы в нем каждый раз принимали участие одна цепь-агрессор и одна цепь-жертва и повторять прогон задачи, сохраняя одну цепь-жертву и меняя цепи-агрессоры. Только так удастся разделить вклады различных источников, выделить критические цепи, создающие наибольший вклад в общую наводку на одну выбранную цепь.

* * *

В заключение, оценивая заложенные в Altium Designer возможности схемотехнического моделирования и моделирования паразитных эффектов в печатном монтаже, следует отметить, что они являются важным средством верификации проекта на стадии разработки схемы и конструкции и позволяют снизить затраты на отладку функционального узла РЭС на макете или после выпуска его производством.



Связь Altium Designer с внешними базами данных

В условиях промышленной фирмы, оснащенной системой автоматизированного ведения документооборота, проектная документация хранится, как правило, в единой базе данных предприятия. Не составляет исключения и документация на функциональные узлы РЭС, разработанная в среде таких "электронных" САПР, как Altium Designer.

Кроме непосредственно проектной документации в базе данных предприятия радиоэлектронного профиля хранится также обширная библиотека компонентной базы РЭС, содержащая сотни тысяч записей с техническими данными на резисторы, конденсаторы, интегральные микросхемы и другие компоненты.

В соответствии с этим "электронные" САПР должны поддерживать возможность работы с базами данных в двух направлениях:

- ❑ устанавливать связь графических образов и атрибутов (в терминологии Altium Designer — параметров) электрорадиокомпонентов (ЭРК), хранящихся в библиотеках компонентной базы пакета САПР с записями базы данных предприятия¹;
- ❑ вести "историю" проектов — хранить документы в базе данных предприятия, отслеживать изменение версий и предоставлять возможность редактирования и модернизации как финальной, так и любых промежуточных версий проекта или библиотеки².

¹ Документ HELP-системы Altium Designer AP0133 Using Components Directly from Your Company Database.pdf Application Note AP0133 (v2.4) December 21, 2007.

² Документ HELP-системы Altium Designer TU0114 Working with a Version Control system.pdf Tutotial TU0114 (v2.3) July 24, 2007.

Задачи первого направления решаются встроенными средствами Altium Designer. В результате записи базы данных могут быть использованы при автоматическом составлении текстовых конструкторских документов — перечня элементов к схеме электрической принципиальной и спецификации сборочной единицы.

Для реализации второго направления в Altium Designer заложены средства интерфейса с поставляемыми независимо системами управления версиями Concurrent Versions System (CVS) или Subversion (SVN), позволяющие отслеживать изменение версий непосредственно через панель **Storage Manager**.

11.1. Прямая связь с библиотеками корпоративной базы данных

11.1.1. Предварительные замечания

На определенных стадиях выполнения проекта возникает задача формирования текстовых конструкторских документов, в которые вносятся записи, содержащие полную информацию о примененных компонентах — обозначение типа компонента, электрических параметров, вида конструктивного исполнения, показатели устойчивости к воздействию факторов внешней среды, обозначение документа на применение (стандарта, ТУ и т. п.), название фирмы-изготовителя или поставщика и др.

Средства "электронных" САПР позволяют составлять текстовые документы-отчеты, в которых размещаются списки компонентов, цепей и других элементов проекта. Этому служит процедура формирования выходных файлов проекта. Цепочка команд главного меню **File | New | Output Job Files** открывает файл <Имя_проекта>.OutJob (рис. 11.1).

В главном окне Altium Designer разворачивается оболочка файла <Имя_проекта>.OutJob, в которой отображается структура полного набора выходных документов. Нас интересует раздел **Report Outputs**, а в нем пункт **Bill of Materials**. Двойной щелчок мышью на поле **Bill of Materials** открывает диалоговое окно **Bill of Materials For Project** [<Имя проекта>.PrjPcb], в котором формируется список компонентов схемного документа проекта (рис. 11.2). В колонках таблицы для каждого компонента собраны сведения, составляющие "свойства" компонента (**Component Properties**), отображаемые при редактировании документов проекта по двойному щелчку кнопкой

мышли на изображении компонента в схемном документе или документе печатной платы.

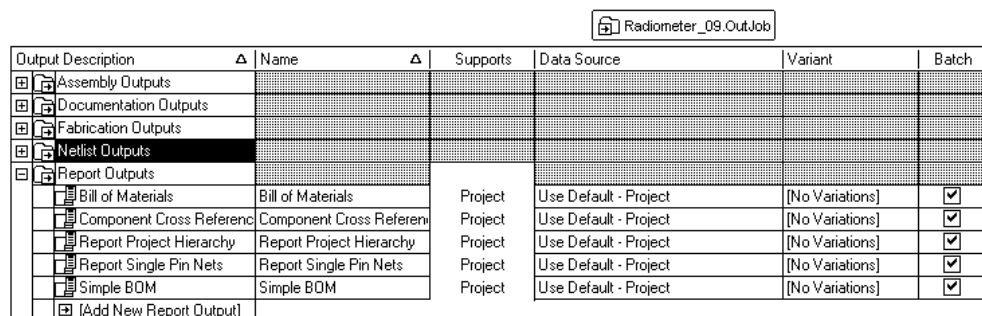


Рис. 11.1. Диалог формирования отчетов о результатах проекта

В поле **All Columns** (Все колонки) располагается список из более 90 позиций и флажки управления, включающие или исключающие колонки из таблицы. Это могут быть параметры компонентов, ссылки на модели, указатели на библиотеку или базу данных, реквизиты документа и т. п.

Компоненты в таблице сгруппированы по типам. Записи для однотипных компонентов собираются в одну строку таблицы. Список располагается в порядке возрастания латинской буквы в позиционном обозначении компонентов в колонке **Designator**.

В нижнем поле окна (рис. 11.2) сгруппированы функции экспорта — преобразования данных таблицы отчета в формат файлов, включаемых в структуру проекта. Всего программа поддерживает пять возможных форматов данных:

- ☐ CSV [Comma Delimited] — текстовый формат с запятой в качестве разделителя полей (*.csv);
- ☐ Microsoft Excel Worksheet — формат рабочего листа Excel (*.xls);
- ☐ Tab Delimited Text — текстовый формат с использованием в качестве разделителя полей знака табуляции (*.txt);
- ☐ Web Page — формат Web-страницы (*.htm, *.html);
- ☐ XML Spreadsheet — таблица в формате Internet Explorer (*.xml).

Эти форматы выходных данных, возможно, удовлетворяют требованиям стандартов ведения текстовой конструкторской документации, принятым в странах англоязычного ареала, однако ни один из них не соответствует ЕСКД ни по составу полей, ни по их содержанию.

Bill of Materials for Project [Radiometer_09.PriPcb] (No PCB Document Selected)

Grouped Columns	Show	C. Δ	Description	Designator	Footprint	LibRef	Q
Comment	<input checked="" type="checkbox"/>		SBM-21	SBM-21	SBM-21	SBM-21	4
Footprint	<input checked="" type="checkbox"/>		CAPC_32	Capacitor_Ceramics_Dielectric_HF	C1, C2, C3, C4, C6, C7, C9, C10, C11, C12	CAPC_3216_SM_3	CAPC_32 10
			CAPC_16	Capacitor_Ceramics_Dielectric_HF	C5, C13, C14, C15	CAPC_1608_SM_14	CAPC_16 4
			CAP_TH	CAP_TH_3216	C8	CAP_TH_3216_SM	CAP_TH 1
			HO-21C	HO-21C	DA1	HO-21C_HO-21C_F	HO-21C 1
			1564AF3	1564AF3	DD1	1564AF3_(Primary)	1564AF3 1
			UKZ1-4/8	UKZ1-4/8	HG1	UKZ1-4/8_UKZ1-4	UKZ1-4/8 1
			RC1608	Surface-mount chip-resistor	R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11	RC1608_SM_1608	RC1608 21
			PD9-2	PD9-2	SA1	PD9-2_PD9-2_Prim	PD9-2 1
			SW-PB	Switch	SB1	SPST-2	SW-PB 1
			TR_M150	TR_M1500_HM3	TV1	TR_M1500_HM3	TR_M150 1
			EPM7064	MAX 7000S 5V In-System Programmable CPLD, 64 Ma	U1	TOPF100_N	EPM7064 1
			KD512	KD512	VD1, VD2, VD5	KD512	KD512 3
			KD629AC	KD629AC	VD3, VD4	KD629AC_KD629A	KD629AC 2
			KT3117	KT3117	VT1, VT2	KT3117_KT3117(P)	KT3117 2
			IDC10RF	IDC10RF	XS1	IDC10RF_IDC10RF	IDC10RF 1

All Columns Show

Address1 ☐

Address2 ☐

Address3 ☐

Address4 ☐

ApprovedBy ☐

Author ☐

CheckedBy ☐

Export Options

File Format: Microsoft Excel Worksheet (*.xls)

☒ Add to Project ☐ Open Exported

Excel Options

Template:

☒ Relative Path to Template File

Menu Export...

☒ Force Columns to View ☐ Include Parameters From Database ☐ Include Parameters From PCB

OK Cancel

Рис. 11.2. Окно формирования списка компонентов проекта

В разд. 4.2.5 был приведен пример записи в перечень элементов к схеме электрической принципиальной для отечественного резистора типа С2-23 (см. рис. 4.13). В приведенном примере тип резистора и его допустимая мощность рассеяния могут быть обозначены в имени библиотечного элемента, номинальное значение сопротивления — в атрибуте {value}, а для остальных элементов полной записи в перечне и спецификации должен быть составлен список параметров, значения которым присваиваются в процессе работы со схемой.

Известны попытки автоматизации формирования текстовых конструкторских документов¹ с использованием данных о компонентной базе, сосредоточенных в документах проекта, главным образом в схемном документе.

В принципе, можно составить полные списки параметров для каждого компонента схемной библиотеки и хранить их в библиотечных файлах. Но это непомерно повышает нагрузку на библиотеки, поскольку полный список компонентов, по всей номенклатуре типов, по всей шкале номиналов, допусков, исполнений и т. п. может составлять сотни тысяч позиций. При формировании схемного образа компонента, его топологического посадочного места, прочих моделей обычно стараются обойтись лаконичными именами, достаточными для идентификации компонента при поиске его в библиотеке и включения в схему.

¹ См. например, материалы И. В. Брагина на интернет-сайте www.verzak.ru.

В схемном документе проекта формировать списки параметров для каждого компонента тоже нерационально: настройки параметров требуют большого ручного труда, а действуют только в текущем документе и теряются при переходе к работе с другими документами и проектами.

В то же время существуют справочники по компонентной базе РЭС, в том числе и "электронные" базы данных, в которых хранятся все эти сведения. Если САПР функциональных узлов РЭС содержит функции связи с базами данных, это обеспечивает доступ к сведениям о компонентах из базы данных, не перегружая библиотеки программного пакета.

Altium Designer включает в свой состав средства так называемой "горячей" связи с базами данных. Есть два вида файлов для такой связи:

- Database Link file, (расширение DBLink);
- Database Library file (расширение DBLib).

В первом случае устанавливается и поддерживается связь компонентов схемной библиотеки или компонентов, используемых в проекте с информацией о них, содержащейся в базе данных. Во втором случае полная информация о компоненте формируется динамически, при вызове компонента из библиотеки и применении его в проекте, в соответствии с информацией, содержащейся в базе данных.

В поставку Altium Designer входит небольшая база данных в формате Microsoft Access, в записях которой, сгруппированных в 7 таблиц, содержатся параметры и ряд других сведений об ограниченной номенклатуре компонентов производства зарубежных фирм — всего немногим более 2500 резисторов, конденсаторов, транзисторов, ключей, интегральных микросхем, электрических соединителей, диодов, кварцевых резонаторов. Этих данных хватит для освоения технологии ведения проекта с использованием "горячих" связей с базой данных, но будет недостаточно для полноценной профессиональной работы. Пользователь может расширить базу. Кроме того, в поставку Altium Designer включена профессиональная база данных по конденсаторам фирмы Vishay Telefunken, содержащая около 10 000 записей и связанные с ней библиотеки схемных компонентов и посадочных мест.

В Altium Designer есть также возможность установить связь с базами данных в формате Microsoft Excel и другими внешними базами данных, поддерживающими технологию OLE DB.

Рассмотрим подробнее оба способа связи проекта Altium Designer с базами данных.

11.1.2. Создание файла связи с базой данных Database Link file

Этот способ обеспечивает связь компонента схемной библиотеки с соответствующей записью в базе данных. Для связи используется одно ключевое поле (например, кодовое обозначение компонента или номер записи в базе данных) или совокупность ключевых полей. В последнем случае должен быть построен специальный запрос.

При данном методе такая информация о компоненте, как тип, некоторые параметры и информация о связи с моделями, должна быть предопределена в библиотечном компоненте Altium Designer. Ключевое поле для связи с базой данных также нужно определить среди параметров компонента.

После того как вся эта подготовка выполнена, к библиотечному пакету (*.LibPkg) или к PCB-проекту (*.PrjPcb) командой **File | New | Database Link File** или командой **Project | Add New to Project | Database Link File** файл связи с базой данных подключается к проекту. В главном окне программы разворачивается оболочка файла (рис. 11.3).

- ☐ В поле **Source of Connection** одной из трех "радиокнопок" выбирается база-объект, с которой устанавливается связь:
 - **Select Database Type** — выбрать тип базы данных: в раскрывающемся поле доступны два типа (формата): Microsoft Access или Microsoft Excel; здесь же кнопкой **Browse** активизируется выбор пути к базе выбранного формата, а в поле **Path** отображается найденный путь. По умолчанию предлагается путь к упомянутой ранее базе Microsoft Access;
 - **Use Connection String** — использовать строку связи: по умолчанию в нижележащем поле указывается тот же путь к выбранной базе; кнопкой **Build** (Построить) активизируется диалоговая процедура построения вызова внешней базы данных (рис. 11.4);
 - **Use Data Link File** — если база-источник определена в файле связи Microsoft Data Link File (*.udl), кнопкой **Browse** выполняется поиск этого файла связи в памяти компьютера.

Диалоговое окно **Свойства связи с данными** (в локализованной ОС Windows текстовые сообщения пишутся по-русски) содержит четыре вкладки. На вкладке **Поставщик данных** (рис. 11.4, а) приводится список поставщиков баз данных, поддерживающих технологию OLE DB, связь с которыми под-

держивается в Altium Designer. Щелчок на кнопке **Далее** переводит диалог на вкладку **Подключение**, на которой назначается путь к базе, определенной на предыдущем шаге, вводится имя и пароль пользователя. По умолчанию здесь определен путь к той же базе данных Microsoft Access.

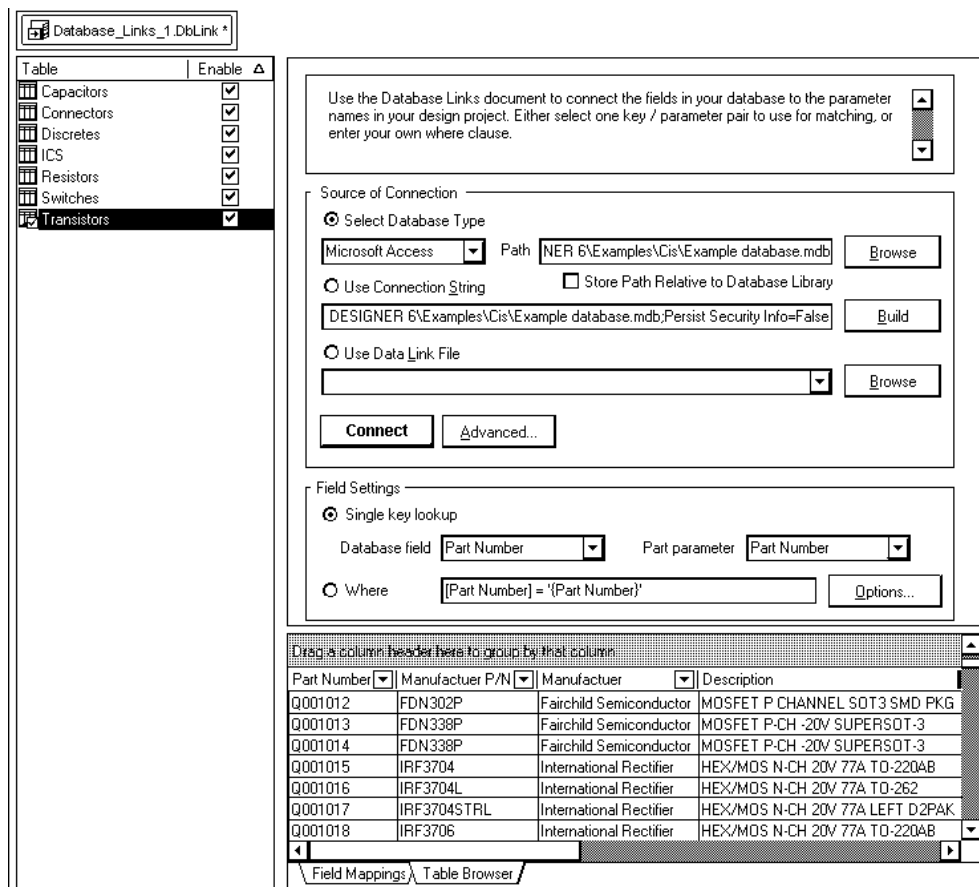
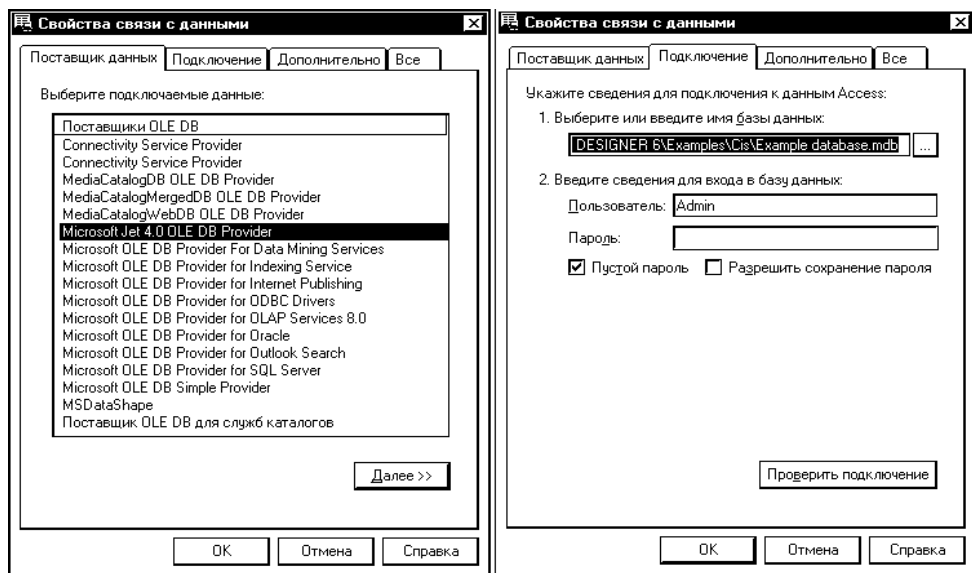


Рис. 11.3. Диалоговая оболочка файла связи с базой данных Database Link File

После того как определен путь к базе данных, связь устанавливается щелчком на кнопке **Connect**. При успешном соединении кнопка становится пассивной и надпись на кнопке **Connect** затеняется серым фоном. В случае невозможности установления связи выводится сообщение об ошибке (красным цветом). Если в процессе работы связь переопределяется, надпись на кнопке меняется с **Connect** на **Reconnect** (Соединить заново).



а

б

Рис. 11.4. Окно свойств связи с базами данных:

а — список поставщиков баз данных; б — указание пути к выбранной базе данных

После успешного соединения с базой данных в диалоговом окне-оболочке файла связи отображается содержимое подключенной базы данных (см. рис. 11.3):

- список в левом поле окна показывает состав таблиц компонентов базы данных (на рис. 11.3 отображен состав таблиц базы Microsoft Access, входящей в поставку Altium Designer);
- установка флажков в колонке **Enable** активизирует или отключает связь с той или иной таблицей компонентов;
- при указании курсором (со щелчком левой кнопки мыши) в одну из строк списка таблиц в нижнем поле окна выбранная таблица полностью отображается.

Формат и содержание таблицы повторяют таблицу в связанной базе данных (в нашем случае базы Microsoft Access). Редактирование в полях таблицы невозможно.

В заголовках колонок таблицы расположены кнопки, при активизации которых открывается возможность фильтрации компонентов по соответствующему признаку — порядковому номеру в списке, номиналу, допуску, названию фирмы-производителя и т. п.

Для того чтобы параметры компонента из базы данных были переданы компоненту проекта или схемной библиотеки, необходимо связать каждый компонент проекта с единственной строкой таблицы базы данных. Критерий отбора этой единственной строки формируется в поле **Field Settings** окна оболочки файла связи **Data Link File** (рис. 11.5).

Поиск компонента по единственному признаку

Если мы располагаем базой данных, о которой заведомо известно, что в ее записях можно обнаружить компоненты с полным набором необходимых нам параметров (тип, номинал, допуск, производитель, цена, документ на поставку и др.), отобрать компоненты в базе можно по единственному простому признаку, например номеру записи в базе данных.

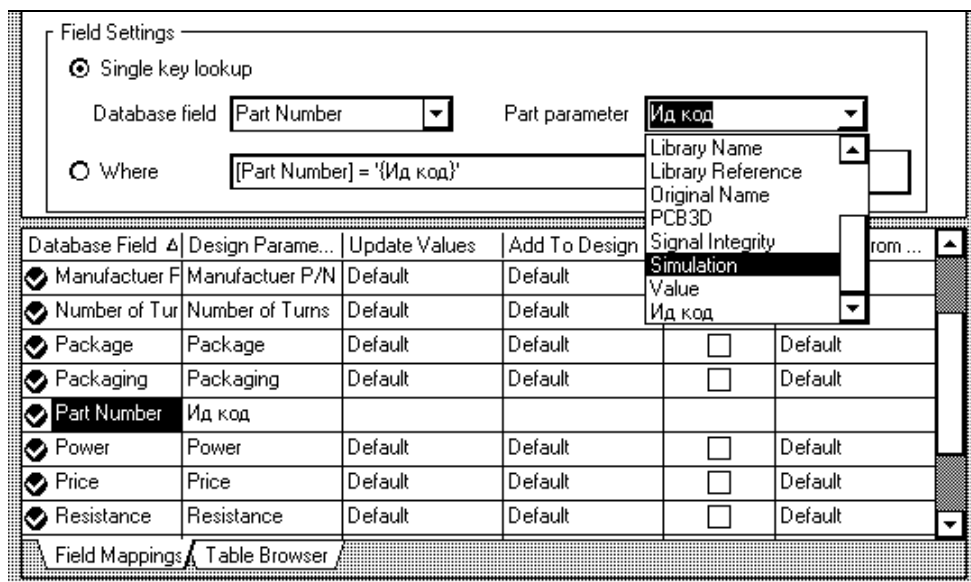


Рис. 11.5. Настройка связи параметров компонентов проекта с записями таблицы базы данных

Для отбора по единственному критерию следует активизировать опцию **Single Key Lookup**. При этом открывается доступ к нижележащим полям:

- ☐ **Database Field** — открывает выпадающий список заголовков всех колонок таблицы базы данных (см. нижнее поле на рис. 11.3). Эти же заголов-

ки перечислены в колонке **Database Field** таблицы **Field Mappings** того же окна-оболочки (рис. 11.5);

- **Part Parameter** — открывает выпадающий список всех параметров всех компонентов активного проекта или библиотеки Altium Designer. Список наполняется только после компиляции активного проекта.

Если связь компонентов проекта с базой данных устанавливается по идентификационным кодам, присвоенным компонентам в результате предварительного просмотра таблиц базы данных, в списке параметров следует выбрать параметр **Ид. код**. Имя параметра в колонке **Design Parameter** таблицы **Field Mappings** "расфокусируется" (термин MS Windows) — делается бледно-серым.

Поиск компонентов по совокупности признаков

При неизвестном заранее идентификационном коде компонента в базе данных компонента с необходимым набором параметров можно найти, составив сложный запрос. Запрос можно составить вручную в таблице на вкладке **Table Browser** (см. рис. 11.3) либо автоматически с помощью опции **Where** (дословно — "где?", а по существу — в какой области значений совокупности признаков).

Чтобы составить сложный запрос вручную, следует выполнить расположенное на темно-сером фоне в заголовке таблицы указание **Drag a column header here to group by that column** — "Перетащите сюда заголовок колонки, чтобы сгруппировать (данные) по названию данной колонки". Стандартным приемом Drag-and-Drop в данном поле составляется цепочка имен колонок, определяющая иерархию поиска компонента (рис. 11.6).

Строки таблицы затеняются светло-серым фоном и выстраиваются в порядке возрастания значения первого признака (в приведенном на рис. 11.6 примере — по увеличению номинальной емкости конденсаторов). Для каждого компонента строится дерево параметров, которое можно развернуть щелчком мышью на знаке + в ячейках управления (**Checkbox**), расположенных в начале каждой строки. Когда таким способом доходим до последнего узла дерева поиска, в таблице отображаются строки базы данных для компонентов с заданным набором параметров. Если найдено более одного компонента, следует выбрать единственный из них и использовать его для установления связи с компонентами проекта или библиотеки Altium Designer.

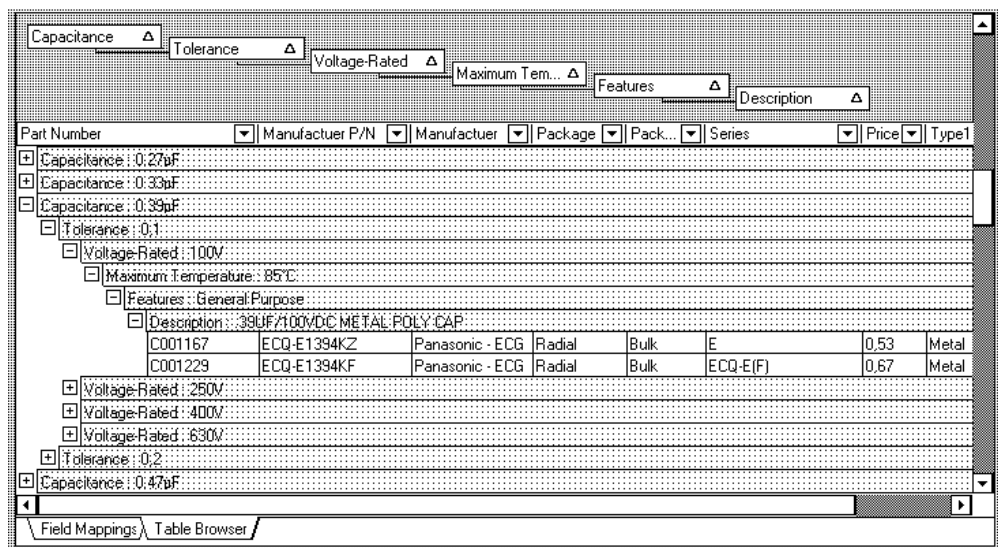


Рис. 11.6. Построение сложного запроса на поиск компонента

Для автоматизации поиска по сложному запросу прежде всего следует активизировать в поле **Field Settings** диалогового окна (см. рис. 11.3 и 11.5) функцию построения запроса **Where** (дословно — где, а точнее — каково сочетание параметров искомого компонента в базе). Запрос **Where** составляется в формате SQL-языка построения запросов. Простейшая форма запроса, формируемая при поиске, построенная ранее по единственному признаку автоматически, отображается в поле напротив кнопки **Where**: `[Part Number] = '{Ид код}'`.

Сложные запросы составляют из полей, заключаемых в квадратные или фигурные скобки и соединяющих их логических или арифметических операторов. В квадратные скобки заключают имена параметров (имена колонок таблицы базы данных), для которых составляется маска поиска. В фигурные скобки заключают значения параметров, которые предстоит отыскать в базе. При этом переменные строчного типа, в отличие от переменных числового типа, заключают еще и в кавычки-апострофы.

По окончании формирования запроса следует внести параметры найденных компонентов в схемный документ проекта. Для этого в активном схемном документе выполнить команду главного меню **Tools | Update Parameters From Database**. Откроется одноименное диалоговое окно (рис. 11.7).

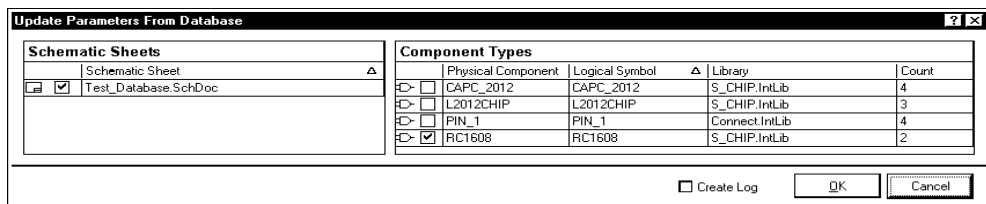


Рис. 11.7. Восстановление параметров компонентов из базы данных

- ☐ В поле **Schematic Sheets** отображается список схемных документов (листов) проекта, для которых выполняется обновление параметров компонентов. Установкой флажков в строках списка выбирают документы, подлежащие обработке.
- ☐ В поле **Component Types** отображается список всех типов компонентов, использованных в проекте. Флажками в левой части строк составляют список компонентов, подлежащих обновлению.

Щелчком на кнопке **OK** закрывают диалог. Процедура обновления переходит на следующий шаг: открывается диалоговое окно выбора параметров, подлежащих обновлению — **Select Parameter Changes** (рис. 11.8).

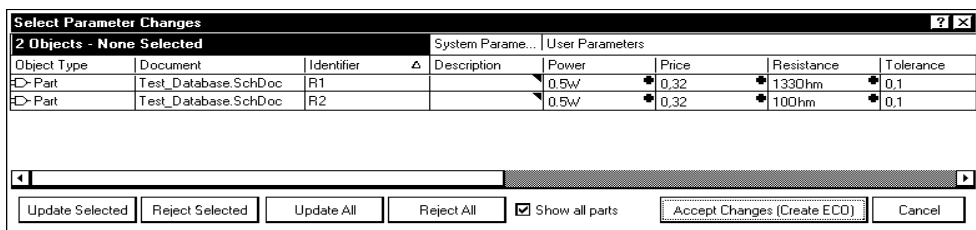


Рис. 11.8. Выбор параметров для передачи в проект

В трех левых полях таблицы отображается список отобранных на предыдущем шаге компонентов схемного документа. В правой половине таблицы отображается список параметров компонентов, отобранных по запросу в базе данных.

Выбирая курсором ячейки этой таблицы (удерживая при этом нажатой клавишу регистра <Shift>), составляют список параметров, подлежащих передаче компонентам в схемный документ. Щелчком на одной из четырех кнопок, **Update Selected**, **Update All**, **Reject Selected** или **Reject All**, назначают передачу или отказ от передачи в схемный документ выбранных или всех пара-

метров, входящих в таблицу. Обновление параметров в схемном документе выполняется как стандартная ECO-процедура.

Щелчок на кнопке **Accept Changes (Create ECO)** запускает процедуру обновления параметров. Открывается диалоговое окно **Engineering Changes Order** (рис. 11.9).

Последовательными щелчками на кнопках **Validate Changes** и **Execute Changes** активизируется проверка правильности предлагаемых изменений и исполнение ECO-процедуры. В случае отсутствия ошибок строки списка помечаются в поле **Status** метками зеленого цвета. При обнаружении ошибок ставится метка-крестик на фоне красного кружка, и соответствующее изменение не вносится в схемный документ. Для окончательной проверки внесенных обновлений следует выбрать курсором компонент в схеме и двойным щелчком левой кнопки мыши вызвать окно свойств **Component Properties**.

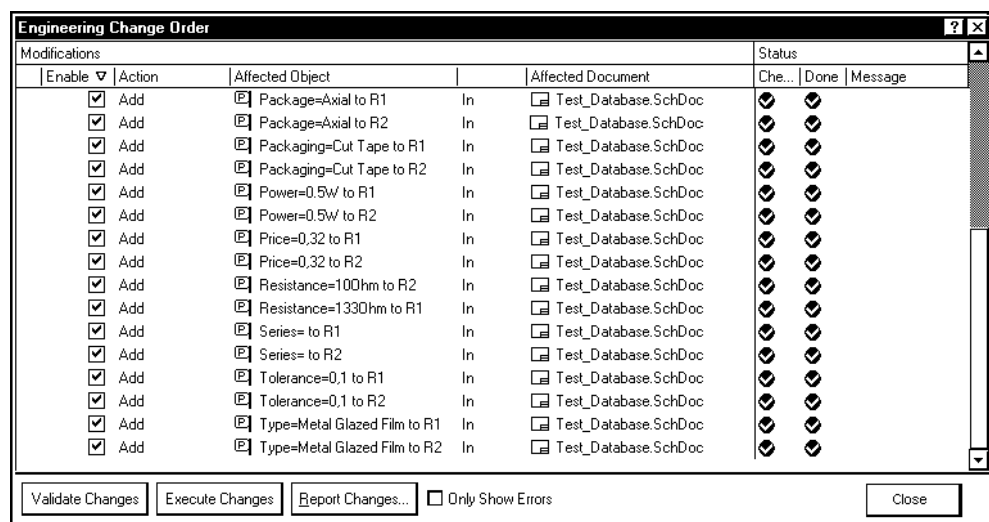


Рис. 11.9. Диалог внесения изменений в проект

11.2. Связь с базой данных через файл библиотеки Database Library file

Этот метод также устанавливает связь компонентов схемного документа с записями во внешней базе данных. Файл Database Library File (*.DBLib)

представляет собой файл-указатель на внешнюю базу данных, в записях которой установлены все связи схемных символов, ссылки на модели и информация о параметрах компонентов. Связь возможна с любой базой данных, в которой поддерживается технология OLE DB. Для связи служит ключевое поле в параметрах схемного компонента. Это поле, как и при осуществлении связи через файл DBLink, может содержать единственный признак или сложный запрос Where в конструкциях языка SQL.

Различие состоит в способе передачи информации из полей записи базы данных при извлечении компонентов схемной библиотеки и использовании их в схемном документе проекта. Компонент схемной библиотеки (*.SchLib) при этом способе связи с базами данных представляет собой пустую оболочку, в которой определен только схемный символ (УГО), а вся информация о моделях и параметрах компонента динамически извлекается из записей базы данных или из других источников, путь к которым указывается при настройке, и компонент составляется "на лету" при формировании схемного документа.

В отличие от метода связи через DBLink-файл, который обязательно должен быть принадлежностью проекта, файл DBLib может не входить в состав документов проекта. Он должен быть включен в состав доступных библиотек и отображаться в рабочей панели Libraries.

11.2.1. Образование DBLib-файла

Для образования оболочки DBLib-файла нужно выполнить команду главного меню **File | New | Library | Database Library**. В главном окне Altium Designer откроется оболочка с предопределенным по умолчанию именем файла Database_Libs1.DBLib (рис. 11.10).

Командой **File | Save As** сохранить DBLib-файл в памяти компьютера.

Левая панель окна-оболочки DBLib-файла в момент его образования пуста. В правой панели сосредоточены элементы диалога установления связи с внешними базами данных и функции вызова необходимой записи подключенной базы данных.

В поставку Altium Designer входит пример DBLib-файла, расположенный по адресу ...\\Altium Designer\\Examples\\Cis\\ExampleDBLib\\VishayCapacitor.DBLib и устанавливающий связь схемных документов с базой данных по конденсаторам фирмы Vishay — VishayCapacitor.mdb (в формате Microsoft Access).

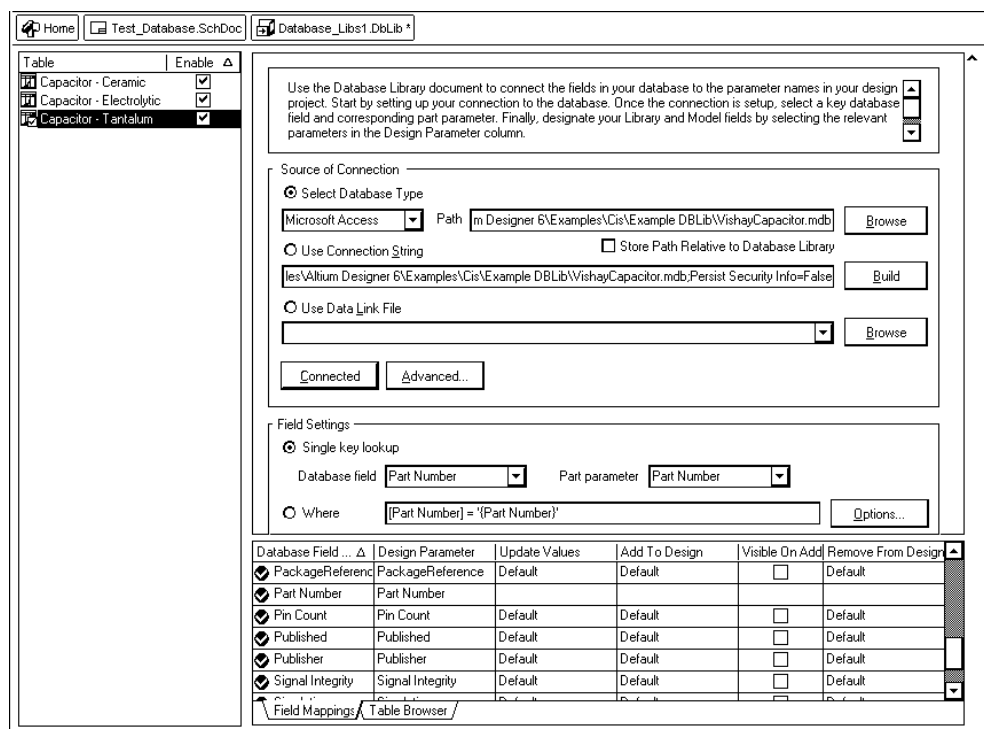


Рис. 11.10. Диалоговая оболочка Database Library

В поле **Source of Connection** сгруппированы три опции поиска связи с внешними базами данных:

☐ **Select Database Type** — выбрать тип базы данных:

- в нижележащем поле выбирают один из двух predeterminedных форматов внешней базы данных — Microsoft Access (по умолчанию) или Microsoft Excel;
- в поле **Path** отображается путь к выбранной базе данных в файловой системе компьютера: по умолчанию это путь к установленной вместе с программой Altium Designer базе данных по конденсаторам фирмы Vishay;
- кнопка **Browse** запускает стандартную процедуру поиска пути к необходимой базе данных.

☐ **Use Connection String** — использовать строку связи: по умолчанию в нижележащем поле отображается строка связи — указывается провайдер

Microsoft Jet 4.0 OLE DB, предоставляющий связь с базами данных в формате Microsoft Access, путь к базе конденсаторов VishayCapacitor.mdb и состояние защиты данных. Кнопка **Build** (Построить) активизирует диалог построения строки для установления связи с другими внешними базами данных (рис. 11.11). Диалог открывается на вкладке **Подключение**: предлагается указать путь и имя необходимой базы данных, а также определить имя пользователя и пароль (рис. 11.11, а).

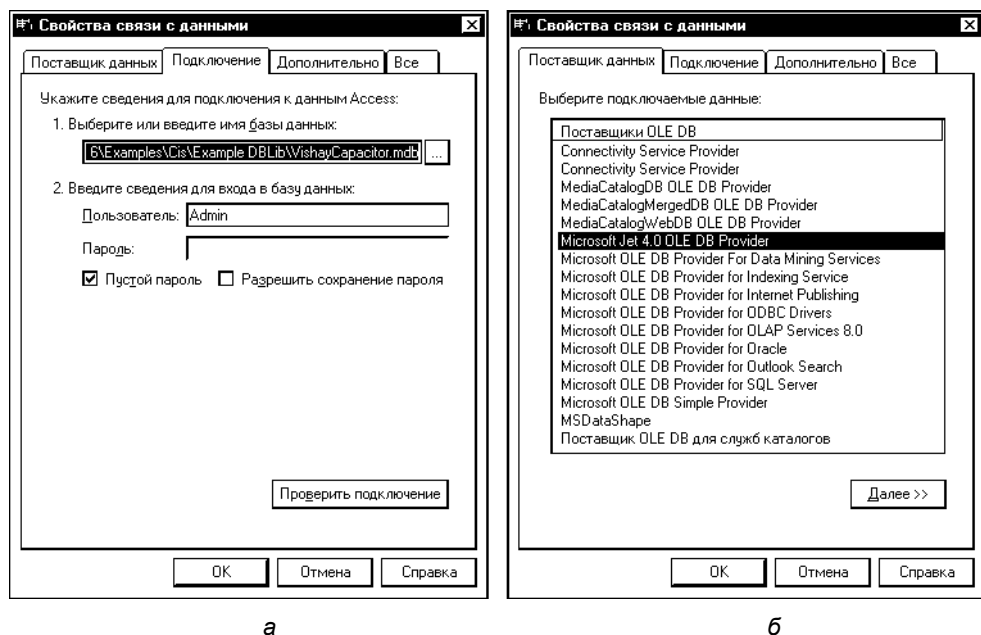


Рис. 11.11. Окно свойств связи библиотечной структуры Database Library с базами данных: а — указание пути к выбранной базе данных; б — список поставщиков баз данных

На вкладке **Поставщик данных** предоставляется возможность выбора из 19 поставщиков. По умолчанию предлагается тот же Microsoft Jet 4.0 OLE DB Provider (рис. 11.11, б) и все та же база конденсаторов VishayCapacitor.mdb.

□ **Use Data Link File** — данную опцию следует выбрать, если связь с базой данных устанавливается с помощью файла связи Microsoft Data Link File (*.udl). Кнопка **Browse** запускает стандартную процедуру поиска данного файла, в котором хранится строка связи с необходимой базой данных.

В любом из трех рассмотренных случаев связь с базой данных устанавливается щелчком мыши на кнопке **Connect**. При успешном подключении надпись **Connect** на кнопке меняется на **Connected** (Присоединено) и затеняется серым фоном. При обнаружении ошибок программа выдает (красным цветом) сообщение **Connection Failed. Check your connection settings** (Соединения не получилось. Проверьте ваши настройки).

При переопределении пути, имени базы данных и других изменениях настроек кнопка **Connect** снова становится активной, а надпись на ней меняется на **Reconnect** — присоединить заново.

В результате успешного соединения в левом поле окна-оболочки DBLib-файла отображается список таблиц подключенной базы данных. При выборе курсором одной из таблиц ее содержание отображается на вкладке **Table Browser** в нижней части правого поля. Устанавливая флажки в колонке **Enable**, определяют состав таблиц, которые будут включены в окончательно сохраняемый DBLib-файл.

- На вкладке **Field Mappings** отображается соответствие обозначений колонок таблицы базы данных и параметров документов проекта, а также определяются действия, которые производятся над параметрами компонента при их передаче из базы данных в документ проекта и исключении из этого документа.
- В колонке **Database Field Name** содержатся имена всех полей (заголовки колонок) активной таблицы присоединенной базы данных.
- В колонке **Design Parameter** перечисляются параметры компонентов, входящих в схемный документ активного проекта. По умолчанию имена параметров совпадают с именами полей базы данных. В случае необходимости изменить соответствие имен параметра и поля базы данных следует селектировать курсором соответствующую ячейку в колонке **Design Parameter**, после чего повторным щелчком левой кнопки мыши в той же ячейке активизировать кнопку вызова выпадающего списка реальных параметров компонентов активного схемного документа и выбрать в этом списке необходимое имя.

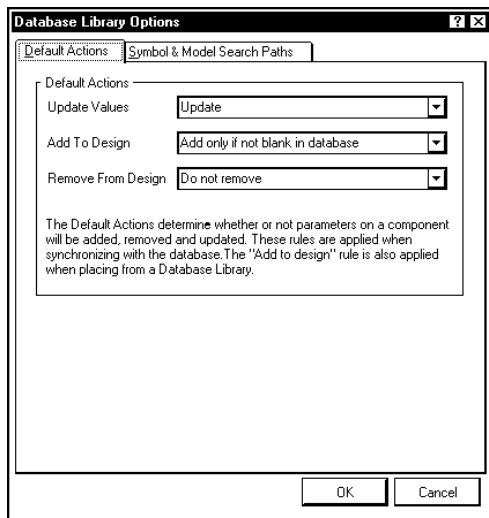
Имя параметра проекта можно изменить прямым вводом с клавиатуры. Как в предыдущем, так и в этом случае при помещении компонента из базы данных в документ проекта ему будет присвоен параметр с новым именем.

Если включать какой-либо параметр из поля базы данных в документ проекта не нужно, следует выбрать в выпадающем списке позицию **[None]** — пусто.

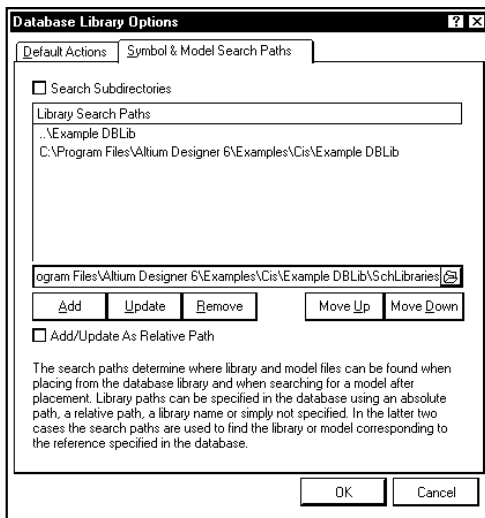
Поля таблицы, которые передаются в параметры документа проекта, помечаются зеленым значком, поля, которые не передаются — красным.

- ❑ В колонке **Visible On Add** флажками в случае необходимости устанавливается видимое отображение параметра, присваиваемого в активном схемном документе компоненту из базы данных.
- ❑ В колонках **Update Values**, **Add To Design** и **Remove From Design** по умолчанию назначено действие **Default**, выполняемое автоматически при обновлении значений параметров в активном документе проекта по команде **Tools | Update Parameters From Database**, добавлении параметров или их исключении из документа проекта. Эти действия, опять же по умолчанию, назначены на вкладке **Default Actions** диалогового окна **Database Library Options** (рис. 11.12, а), вызываемого по команде **Tools | Options** в активном DBLib-документе.

Данная настройка распространяется на все без исключения параметры компонентов, поступающих из присоединенной базы данных. Если требуется назначить для отдельных параметров другие правила обращения с данными, то следует выбрать соответствующие ячейки в колонках таблицы, открыть в каждой ячейке выпадающий список и выполнить в каждом случае необходимые действия.



а



б

Рис. 11.12. Диалог Database Library Options: а — настройка действий по умолчанию; б — указание пути к библиотекам моделей

Пути к библиотекам и файлам моделей для компонентов базы данных назначают на вкладке **Symbol & Model Search Paths** того же диалога **Database Library Options** (см. рис. 11.12, б). Ссылки на эти пути должны быть занесены в поля таблиц базы данных.

11.2.2. Использование компонентов DBLib-библиотеки

После успешного соединения с базой данных и сохранения файла в памяти компьютера соединение с выбранной базой данных будет выполняться автоматически при каждом следующем открытии данного файла, если только путь к базе данных оставался неизменным. Образованный файл *.DBLib включается в ветвь **Libraries** дерева документов текущего проекта и в число библиотек, доступных для использования.

При этом собственно сам этот файл не содержит в своем составе ни базы данных, ни каких-либо библиотек, а является файлом конфигурации, управляющим присоединением компонентов связанной базы данных к схемному документу активного проекта. Сами же таблицы базы данных отображаются в плавающей панели **Libraries** как библиотеки компонентов. Все эти таблицы-библиотеки имеют общее имя и расширение имени файла и различаются по дополнительному обозначению, повторяющему заголовок соответствующей таблицы в базе данных, например, Dbase_Lib.DBLib-Capacitor-Electrolytic (рис. 11.13).

Выбирая в панели **Libraries** необходимую библиотеку в списке доступных, разработчик видит список компонентов соответствующей таблицы базы данных. В двух колонках этого списка размещаются:

- ❑ **Part Number** — условное обозначение (номер) компонента, принятое в базе данных;
- ❑ **Library Ref** (Reference) — ссылка на имя компонента в библиотеке.

При указании курсором компонента из данного списка в полях панели **Libraries** отображаются УГО схемного элемента, имена присоединенных моделей и графическое изображение его посадочного места (**Footprint**).

Щелчком на кнопке **Place <Library Ref>** или с помощью приема Drag-and-Drop выбранный компонент помещают в схемный документ проекта.

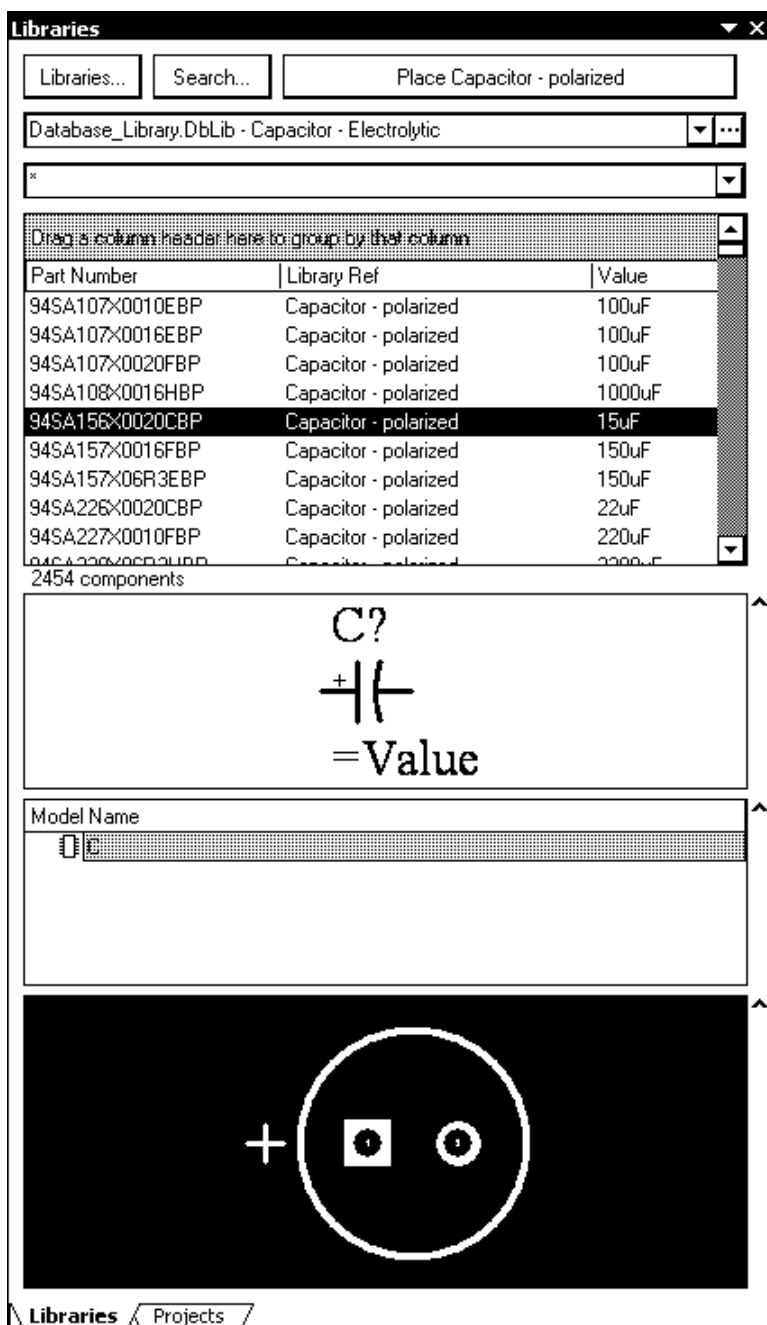


Рис. 11.13. Панель библиотек **Libraries** со списком компонентов из базы данных

При этом на физическом уровне происходит невидимая пользователю работа — компонент, помещаемый в документы проекта, формируется динамически, в процессе его размещения: схемный символ извлекается из ассоциированной с базой данных библиотеки *.SchLib, путь к которой указан в таблице базы данных в графе **Library Path**, конструктивный образ (посадочное место) присоединяется из ассоциированной библиотеки *.PcbLib, путь к которой указан в графе **Footprint Path**, сюда же подключается весь набор параметров и ссылок на модели, содержащихся в записях базы данных.

Однако одно дело — иметь возможность извлечь из базы данных и поместить в схемный документ любой из хранящихся в базе компонентов и совсем другое — отыскать в базе данных и поместить в схему компонент с заранее точно известными параметрами (коммерческим обозначением, номиналом, допуском, ТКЕ или ТКС, рабочим напряжением, уровнем собственного шума, климатическим исполнением и т. п.).

Само по себе обозначение компонента в панели **Libraries** мало что говорит о его параметрах. Для обнаружения в библиотеке базы данных компонентов с заданным набором параметров служит заложенная в Altium Designer функция поиска компонентов по набору признаков-параметров.

Последовательность поиска такова.

1. Активизировать поиск щелчком на кнопке **Search** в панели **Libraries**. Откроется диалоговое окно **Libraries Search** (рис. 11.14).
2. В пустое главное поле окна ввести запрос в конструкторах языка поисковых процедур Altium Designer (Query Language). Для проверки синтаксиса запроса и исправления ошибок следует пользоваться функциями помощника Query Helper, вызываемого щелчком на кнопке **Helper**. Кнопка **History** открывает окно просмотра списка предыдущих запросов. Выбрав один из предыдущих запросов, кнопкой **Apply Expression** можно передать его в поле запросов окна (рис. 11.14) и повторить поиск.
3. В поле **Options** нужно выполнить настройку области поиска (Search in):
 - в первом выпадающем списке указать опцию **Database Components** (Поиск компонентов в базе данных);
 - во втором выпадающем списке выбрать поиск во всех таблицах подключенных баз данных **<Search All Tables>** или указать одну из предлагаемых таблиц, если заведомо известно, что искомый компонент находится именно в ней.

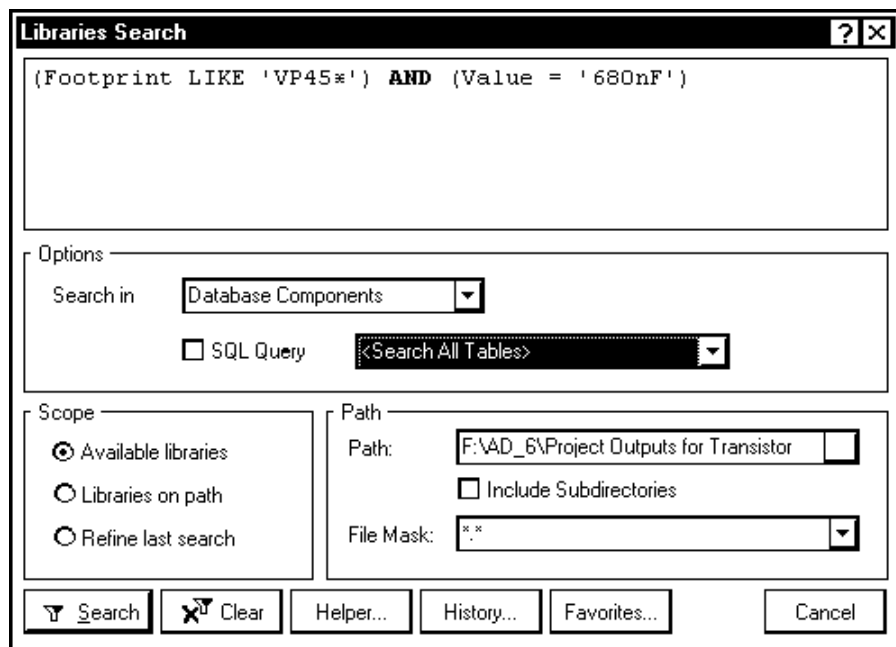


Рис. 11.14. Формирование запроса на поиск компонента в библиотеке

4. Щелчком на кнопке **Search** запустить поиск. Программа обнаруживает в базе данных компоненты, соответствующие составленному запросу, и выводит результат поиска в панель **Libraries** (рис. 11.15).

Так, по запросу, приведенному на рис. 11.14, в базе данных VishayCapacitor.mdb будут найдены шесть конденсаторов емкостью 680 нФ с посадочным местом, обозначение которого начинается с символов VP45.

По умолчанию в поле имен найденных компонентов панели **Libraries** показаны только номера записей базы данных (**Part Number**) и ссылки на имя компонента (**Library Reference**) в ассоциированной библиотеке *.SchLib. Для размещения в этом поле других сведений о компонентах (номинала, допуска и т. п.) следует щелчком правой кнопки мыши вызвать контекстное меню и указать в нем команду **Select Columns** — выбрать колонки (см. рис. 11.15). Откроется окно выбора параметров, подлежащих отображению в панели **Libraries** (рис. 11.16).

В окне отображается список полей базы данных (параметров компонентов). Выбирая параметры в левом списке (**Known Parameters**), кнопкой **Add** пе-

ремещают их в правый список выбранных (**Selected Parameters**). После щелчка на кнопке **OK** выбранные параметры отображаются в поле панели **Libraries**.

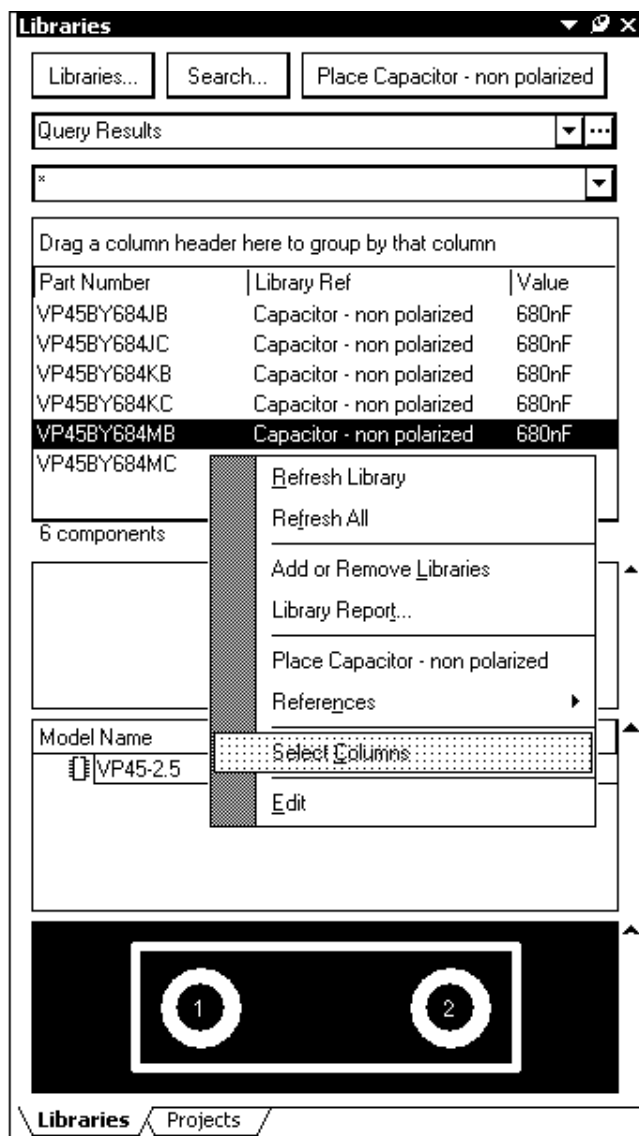


Рис. 11.15. Панель библиотек **Libraries** с результатами поиска компонентов по набору параметров

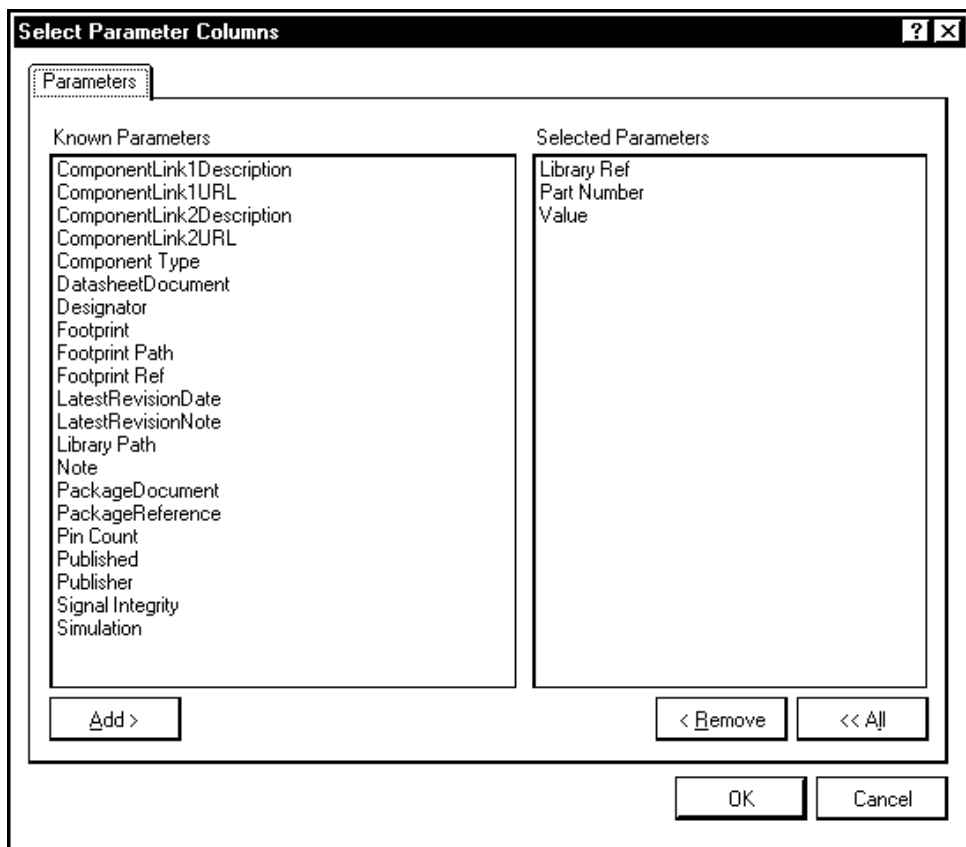


Рис. 11.16. Формирование набора параметров
для представления в панели **Libraries**

Дальнейший выбор единственного компонента для применения в проекте зависит от того, насколько хорошо разработчик сумеет воспользоваться справочными сведениями, ссылки на которые, в частности, могут находиться в полях той же базы данных.

Если нужный компонент выбран, его можно поместить на лист активного схемного документа щелчком на кнопке **Place**, двойным щелчком левой кнопки мыши на имени компонента в панели **Libraries**, либо с помощью приема Drag-and-Drop. Как обычно, перед фиксацией компонента на листе схемы клавишей <Tab> можно выполнить "горячее" редактирование свойств компонента. Откроется диалоговое окно свойств **Component Properties** (рис. 11.17).

Component Properties

Properties

Designator: ☒ Visible ☐ Locked

Comment: ☒ Visible

Part 1/1 ☐ Locked

Description:

Unique Id:

Type:

Library Link

Design Item ID:

☒ Library Name:

☒ Table Name:

Sub-Design Links

Sub-Project:

Configuration:

Graphical

Location X: Y:

Orientation: ☐ Locked

Mode: ☒ Lock Pins ☐ Mirrored

☐ Show All Pins On Sheet (Even if Hidden)

☐ Local Colors

Parameters for C? - Capacitor - non polarized

Visible	Name	Value	Type
<input type="checkbox"/>	ComponentLinkURL	http://www.vishay.com/	STRING
<input type="checkbox"/>	DatasheetDocument	2-Jan-2002	STRING
<input type="checkbox"/>	Designator	VP45BY684MB	STRING
<input type="checkbox"/>	Footprint	VP45-2.5	STRING
<input type="checkbox"/>	PackageDocument	2-Jan-2002	STRING
<input type="checkbox"/>	PackageReference	VP45-2.5	STRING
<input type="checkbox"/>	Pin Count	2	STRING
<input type="checkbox"/>	Published	17-Dec-2003	STRING
<input type="checkbox"/>	Publisher	Altium Limited	STRING
<input type="checkbox"/>	Signal Integrity	Cap	STRING
<input type="checkbox"/>	Simulation	CAP	STRING
<input type="checkbox"/>	Value	680nF	STRING
<input type="checkbox"/>	Ид. код	VP45BY684MB	STRING

Models for C? - Capacitor - non polarized

Name	Type	Description
VP45-2.5	Footprint	Capacitor, Ceramic, Thru-Hole, Dipped Radial; B...

Рис. 11.17. Диалоговое окно управления свойствами компонента с набором параметров, полученных из базы данных

- ☐ В поле **Designator** вводят позиционное обозначение компонента.
- ☐ В поле **Parameters for <обозначение компонента>** отображается набор параметров, переданных с компонентом из базы данных.
- ☐ В поле **Library Link** отображается обозначение компонента в базе данных, имя DBLib-библиотеки и имя таблицы в базе данных, из которой извлекался компонент.

Информация о параметрах компонентов, хранящаяся в базе данных, с течением времени может меняться, и может потребоваться обновить эти сведения в документах проекта¹. Для поддержания синхронности параметрической

¹ AP0144 Keeping Components Up-To-Date.pdf Application Note AP0144 (v1.0) June 09, 2006.

информации компонентов схемного документа и параметров, хранящихся в записях базы данных, следует использовать команду главного меню **Tools | Update Parameters From Database**.

11.2.3. Формирование базы данных из интегрированной библиотеки

Интегрированные библиотеки Altium Designer являются удачным решением, обеспечивающим защиту информации о компонентах от случайного повреждения и возможность переноса проектов (вместе с библиотеками) с одних компьютеров на другие. Такая библиотека может служить для пополнения базы данных предприятия и адаптации ее к используемым на предприятии средствам САПР. С этой целью в Altium Designer заложены так называемые средства миграции библиотек*, позволяющие, в частности, конвертировать интегрированную библиотеку в таблицу базы данных в формате Microsoft Access и связать ее через DBLib-файл с проектом Altium Designer. Конверсия выполняется в среде мастера конверсии **Integrated Library to Database Library Translation Wizard** в несколько шагов.

1. Во вновь образованном или текущем активном DBLib-файле нужно выполнить команду главного меню **Tools | Import from Integrated Libraries**. Откроется первая страница мастера конверсии (рис. 11.18).

На этой странице следует определить, хотим ли мы открыть новую базу данных (**New Access Database**) или дополнить существующую (**Existing Access Database**) и указать путь к этой базе (**Database Location**). В нашем примере образуем новую базу данных с именем `Miscellaneous_DB.mdb` из входящей в поставку пакета интегрированной библиотеки `Miscellaneous Devices.IntLib`. Возможна конверсия сразу нескольких интегрированных библиотек. Каждая из них преобразуется в отдельную таблицу в назначенной базе данных или присоединяется к существующей таблице, если совпадают имена конвертируемой библиотеки и таблицы, уже существующей в базе данных.

2. Щелчком на кнопке **Next** перейти на следующую страницу мастера (рис. 11.19). На данной странице следует указать адрес соответствующего DBLib-документа. В нашем случае следует образовать новый файл библиотеки с именем `Misc_DB.DBLib`.

* AP0143 Database Library Migration Tools.pdf Application Note (v1.0) June 20, 2006.

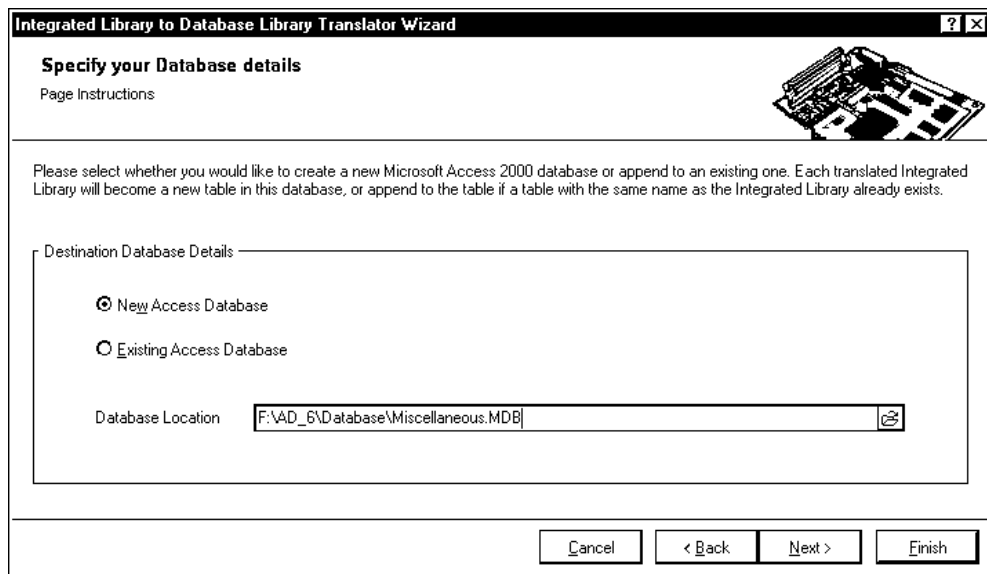


Рис. 11.18. Мастер конверсии интегрированной библиотеки Altium Designer в библиотеку базы данных: объявление пути и имени базы данных для размещения результатов

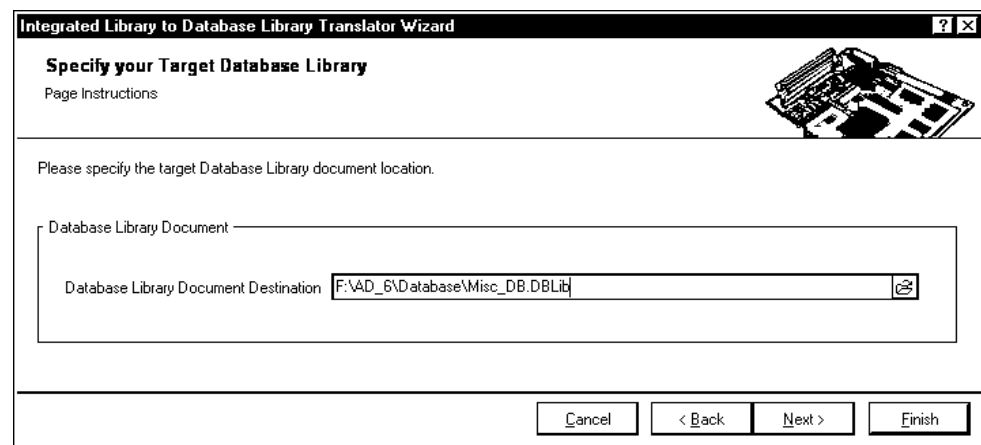


Рис. 11.19. Мастер конверсии интегрированной библиотеки Altium Designer в библиотеку базы данных: объявление пути и имени структуры DBLib

3. Щелчком на кнопке **Next** перейти на следующую страницу (рис. 11.20), на которой нужно составить список интегрированных библиотек, образуя-

щих новую или подключаемых к существующей базе данных. Кнопка **Add** запускает стандартную процедуру поиска файлов интегрированных библиотек. Выбранные библиотеки образуют список в поле **Source Integrated Libraries**. Программа также автоматически образует библиотечный подкаталог и отображает путь к нему в поле **Destination Folder**.

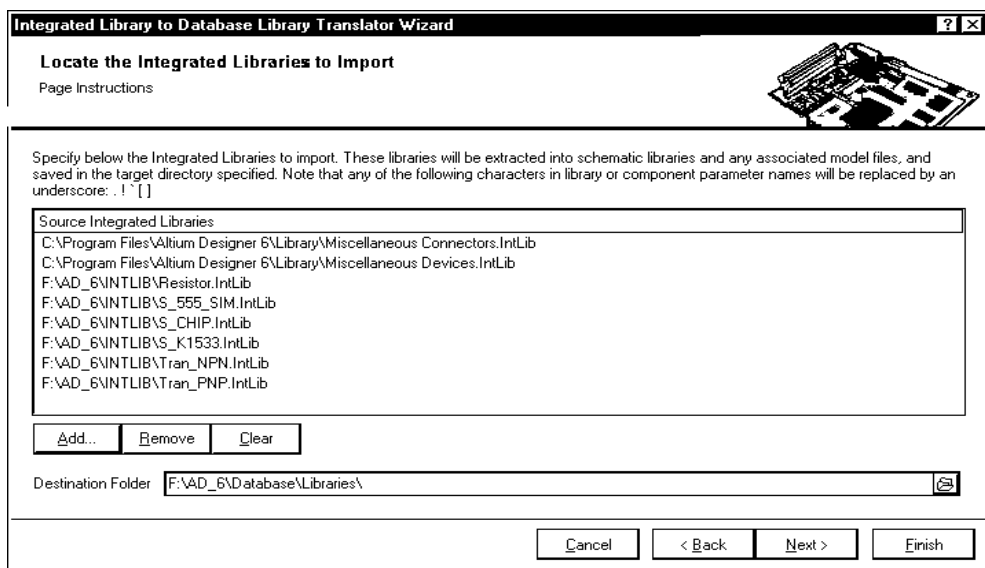


Рис. 11.20. Мастер конверсии интегрированной библиотеки Altium Designer в библиотеку базы данных: составление списка библиотек, подлежащих конверсии

В комментарии, размещенном под заголовком страницы, указывается, что интегрированные библиотеки разделяются на библиотеки схемных символов (*.SchLib) и связанные с ними файлы моделей, и все эти файлы сохраняются в назначенном библиотечном подкаталоге.

4. Щелчком на кнопке **Next** перейти на следующую страницу (рис. 11.21), на которой отображается процесс конверсии набора интегрированных библиотек, составленного на предыдущем шаге: выводится текстовое сообщение о ходе конверсии, сообщения с указанием адреса размещения преобразованных библиотек и базы данных и линейка прогресса, также иллюстрирующая ход конверсии. Вмешательство в ход конверсии невозможно: все кнопки управления в диалоговом окне затенены.

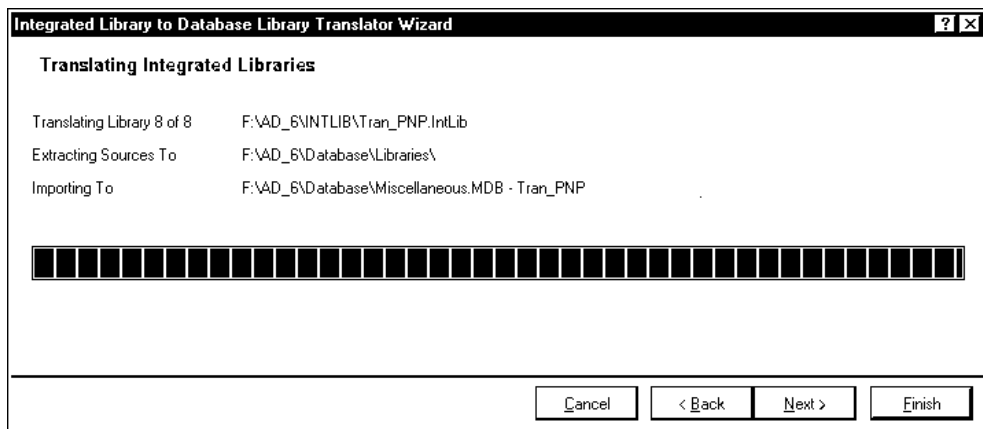


Рис. 11.21. Мастер конверсии интегрированной библиотеки Altium Designer в библиотеку базы данных: отображение прогресса конверсии

5. По окончании преобразования данных мастер переходит на заключительную страницу, на которой предлагается завершить процедуру щелчком на кнопке **Finish**.

Преобразованные данные подключаются в виде таблиц к базе данных и отображаются в оболочке активного DBLib-документа, а сама DBLib-библиотека включается своими новыми таблицами в панель **Libraries**, откуда новые компоненты могут, как обычно, использоваться в схемных документах активного проекта.

11.2.4. Преобразование DBLib-структуры в интегрированную библиотеку

Когда проект разрабатывают несколько организаций, располагающих различным составом баз данных или вовсе не имеющих таких баз, возникает необходимость обратного преобразования DBLib-библиотеки, связанной с базой данных, в обычную интегрированную библиотеку. Средства миграции библиотек Altium Designer позволяют выполнить такое преобразование. При этом каждая таблица связанной базы данных преобразуется в самостоятельный библиотечный пакет с тем же именем, что таблица и стандартным расширением LibPkg.

Конверсию выполняют за три шага в оболочке мастера **Offline Integrated Library Maker** (Построитель офф-лайновой интегрированной библиотеки),

активизируемого из открытого DBLib-документа одноименной подкомандой главного меню **Tools**.

1. На первом шаге в диалоговом окне мастера (рис. 11.22) по умолчанию предлагается конвертировать активную DBLib-библиотеку и связанные с ней записи базы данных в интегрированную библиотеку Altium Designer. При необходимости щелчком на кнопке в правой части поля выбора можно выполнить поиск другой DBLib-библиотеки, подлежащей конверсии. Нажав кнопку **Next**, перейти на следующий шаг.
2. На следующем шаге в поле **Database Tables** диалогового окна мастера (рис. 11.23) отображается список таблиц связанной базы данных, которые конвертируются в офф-лайн-овые библиотечные пакеты. Флажками в колонке **Convert** обозначить таблицы, подлежащие преобразованию.

В поле выбора пути **Specify the output directory in which the newly created integrated libraries will be placed** (Определите выходную директорию, в которой будут размещены вновь созданные интегрированные библиотеки) выбирается путь к выходному каталогу, в который отправляются конвертированные библиотечные пакеты.

Составив необходимый список таблиц, подлежащих конверсии, щелчком на кнопке **Next** перейти на заключительный, третий шаг.

3. На третьем шаге выполняется обработка выбранных таблиц и связей DBLib-документа. Текстовые сообщения и линейка прогресса в диалоговом окне мастера (рис. 11.24) отображают ход преобразования. Вмешательство в ход конверсии невозможно: все кнопки управления в диалоговом окне затенены.

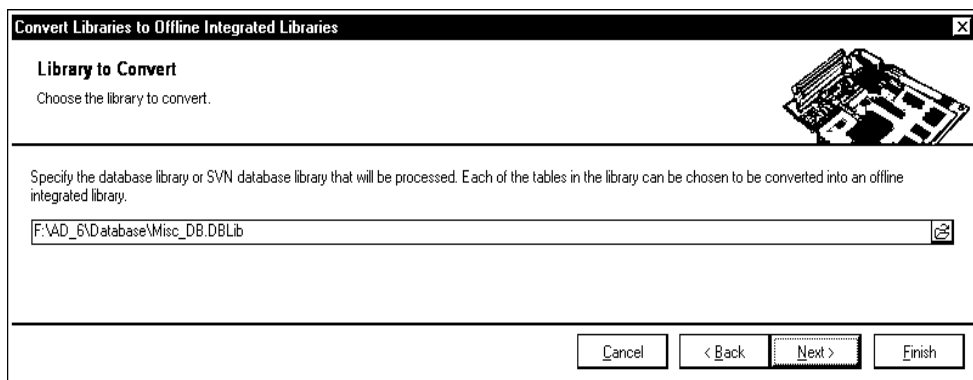


Рис. 11.22. Мастер конверсии библиотек DbLib-структуры в интегрированные библиотеки Altium Designer: указание пути и имени DbLib-файла

*.PcbLib, трехмерных образов *.Pcb3DLib (если трехмерные модели используются в DBLib-библиотеке). Компонентам схемной библиотеки передаются ссылки на SPICE-модели и модели Signal Integrity, установленные в базе данных, порождающей библиотеку.

Образованные библиотечные пакеты отображаются в панели **Projects** (рис. 11.25).

В ходе конверсии программа также компилирует интегрированные библиотеки и размещает их в папке, назначенной в диалоге мастера (см. рис. 11.23).

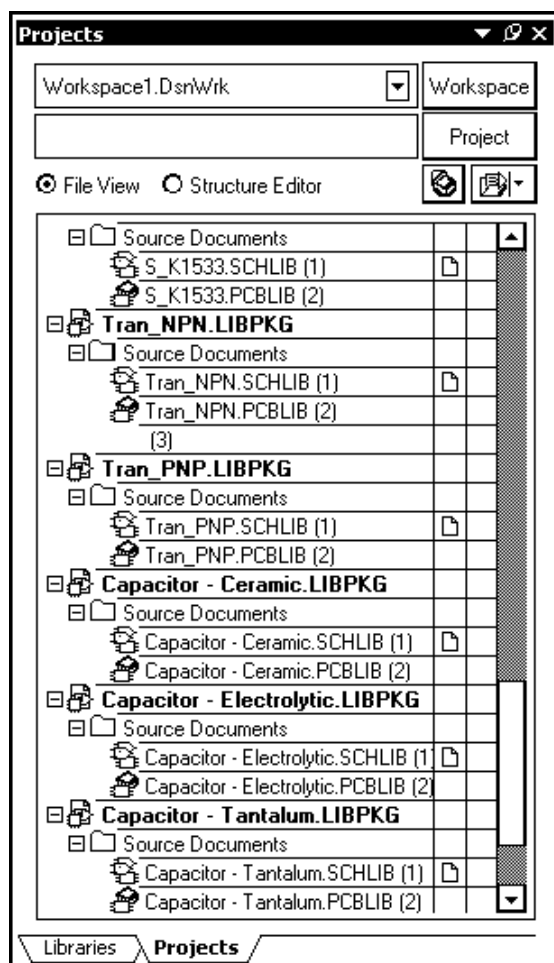
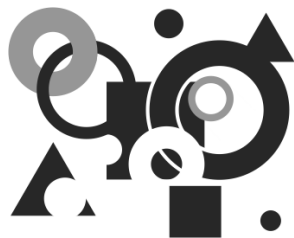


Рис. 11.25. Панель **Projects** со списком конвертированных библиотек



Работа Altium Designer под управлением систем контроля версий

Появление на рынке программных продуктов систем управления версиями вызвано необходимостью отслеживать изменения, вносимые разработчиками электронной документации по мере редактирования и модернизации проектной документации. Эти системы позволяют не только следить за изменениями версии электронных документов некоторой предметной области, но и предоставляют возможность извлекать любую хранящуюся в базе данных версию документа для редактирования, возвращать отредактированный документ в базу данных и сохранять информацию о внесенных изменениях и различиях версий. Они могут действовать независимо от среды, в которой создаются и редактируются документы.

Обычно система управления версиями содержит инструменты, позволяющие избежать конфликтов и потери данных в случаях, когда один и тот же документ вызывают из базы данных и работают над ним одновременно несколько исполнителей, а также интерактивные средства разрешения конфликтов и объединения внесенных изменений в обновленную версию документа.

В Altium Designer возможна работа с системами управления версиями Concurrent Versions System (CVS), Subversion (SVN) и другими, в которых поддерживается интерфейс Microsoft Source Code Control Interface (SCCI). Подробное рассмотрение возможностей и функций, имеющихся на рынке систем контроля версий в системах автоматизированного документооборота¹, не входит в наши

¹ Документ TU0114 Working with a Version Control System.pdf Tutorial TU0114 (v2.3) July 24, 2007.

задачи. Ограничимся рассмотрением заложенных в Altium Designer функций связи с внешними базами данных и ведения библиотек под контролем системы управления версиями электронных документов Subversion 1.4.3.

Прямая поддержка системы Subversion 1.4.3 позволяет непосредственно из среды Altium Designer выполнять такие действия Subversion, как создание пространства для размещения собственной базы данных системы контроля версий (Repository) для хранения исходных и модифицированных проектных документов и библиотек, извлечение документов, редактирование и помещение их обратно в базу, и обеспечивает возможность доступа к любой версии документов. В отличие от традиционных баз данных, в которых хранится информация об электрорадиокомпонентах, будем называть эту базу де-позиторием.

12.1. Активация связи с системой управления версиями

Поддержка связи с системами управления версиями активизируется командой главного меню **DXP | Preferences**. В открывающемся диалоговом окне настройки главной оболочки программы **Preferences** нужно открыть страницу **Version Control | General** (рис. 12.1).

Приведем перевод комментария вверху правого поля окна: "Система управления версиями — это система обращения с множественными вариантами одной и той же единицы информации. Она используется для отслеживания эволюции таких цифровых документов, как исходный код программного приложения, проекты и модели в электронной форме и другая критическая информация, над которой работает группа исполнителей".

В этой же зоне диалогового окна располагается поле для выбора системы контроля версий, с которой предполагается установить взаимодействие. Программа предлагает следующие варианты выбора:

- ☐ **CVS** (Concurrent Versions System) — для одноименной системы управления версиями;
- ☐ **SVN** (Subversion) — для одноименной системы управления версиями Subversion;
- ☐ **SCCI** (Source Code Control Interface) — для других систем управления версиями, поддерживающих интерфейс SCC.

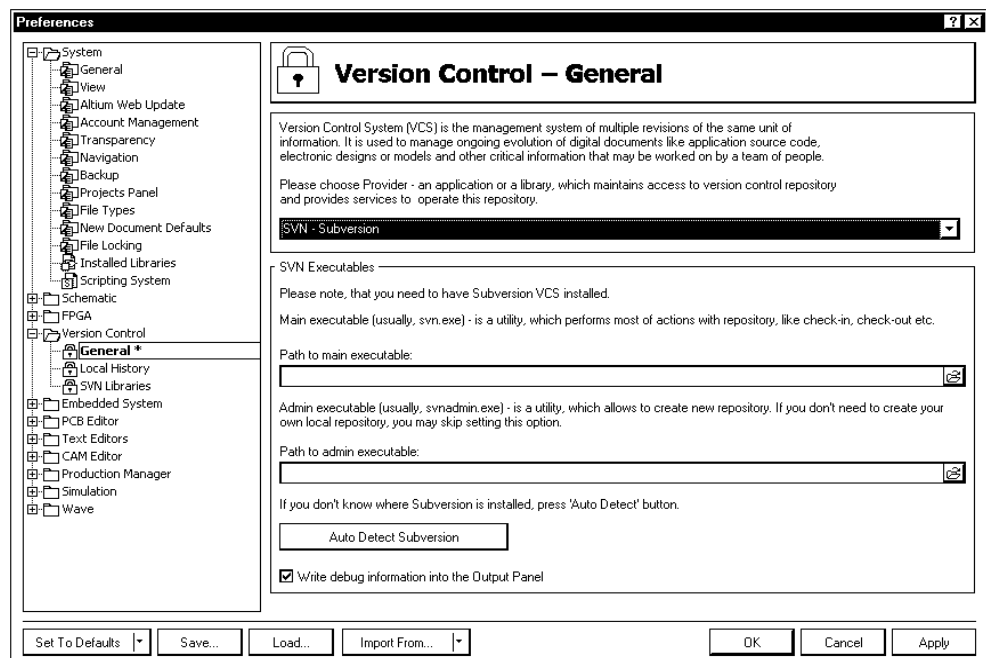


Рис. 12.1. Настройка связи Altium Designer с системой контроля версий

Состав информационных полей нижележащей зоны диалогового окна (рис. 12.1) зависит от выбора системы управления версиями. При выборе системы Subversion под заголовком **SVN Executables** открываются поля для назначения пути к исполняемым модулям:

- ❑ **Path to main executable** — путь к главному исполняемому модулю, позволяющему выполнять основные операции с данными в собственной базе системы управления версиями документов;
- ❑ **Path to admin executable** — путь к исполняемому модулю администратора, под управлением которого создается база данных системы. Если такая база уже существует, данную опцию можно пропустить.

Само собой разумеется, что все эти действия и вся дальнейшая работа возможны, если та или иная система контроля версий установлена на компьютере исполнителя.

После настройки путей систему Altium Designer нужно перезапустить. При этом активизируется поддержка связи Altium Designer с системой управления версиями.

12.2. Управление версиями проектных документов

Вся дальнейшая работа по контролю модификаций, вносимых в проектные документы, производится из панели **Storage Manager**.

12.2.1. Активизация панели *Storage Manager*

Для активизации панели в нижнем обрамлении главного окна Altium Designer командой **System** нужно вызвать контекстное меню и указать в нем подкоманду **Stogage Manager**. В правом верхнем поле панели (рис. 12.2) отображается список документов активного проекта с указанием имени файла, типа документа (проект, схемный документ и т. п.), размера файла, даты последней модификации и статуса документа.

Сразу после активизации панели, пока не создана база данных системы контроля версий, документы имеют статус "неприсоединенных" (Not in version control).

12.2.2. Образование депозитария системы контроля версий

Если на предыдущей стадии была установлена связь Altium Designer с системой контроля версий, то щелчок правой кнопки мыши в панели **Storage Manager** открывает контекстное меню, включающее команды формирования депозитария, вызова документов для редактирования и возврата их обратно с присвоением нового номера версии.

1. Для образования новой базы данных системы контроля версий щелчком правой кнопки мыши в панели **Storage Manager** необходимо вызвать контекстное меню и указать в нем команду **Create Repository**. Откроется диалоговое окно **Create Subversion Repository** (рис. 12.3).
2. В поле **Repository Folder** ввести с клавиатуры путь и имя каталога, предназначенного для организации и хранения депозитария системы контроля версий, или отыскать путь к существующему каталогу.
3. Переключателем **Repository Type** назначить тип файловой системы для создаваемой базы данных. Рекомендуется установить "туземный" (**Native**), действующий на компьютере тип файловой системы.

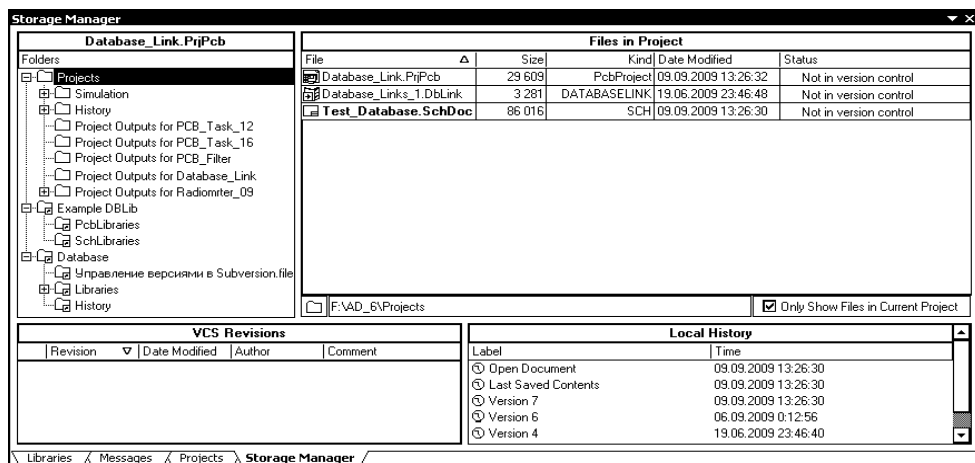


Рис. 12.2. Панель менеджера памяти с деревом документов проекта

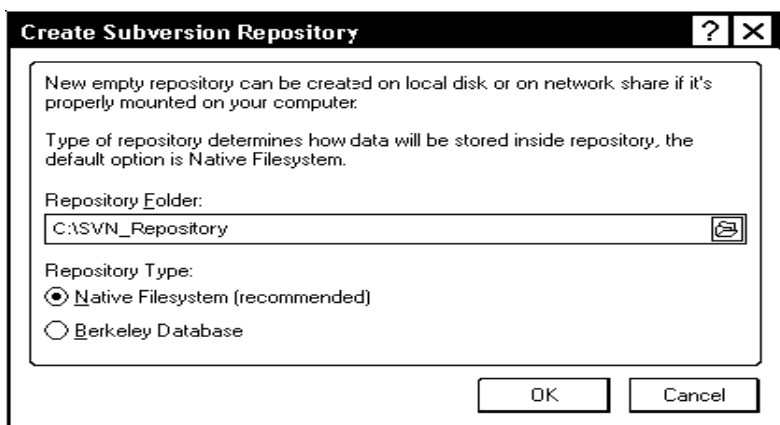


Рис. 12.3. Окно формирования депозитария системы контроля версий

12.2.3. Присоединение документов к базе данных системы контроля версий

Система контроля версий отслеживает модификацию документов в определенном каталоге, для присоединения которого к базе данных системы следует:

1. Открыть проект или свободный документ.

2. Активизировать панель **Storage Manager**.
3. В поле **Folders** панели **Storage Manager** указать каталог, подлежащий присоединению к депозиторию. В правом поле панели отображается состав выбранного каталога.
4. Щелчком правой кнопки мыши открыть контекстное меню и указать в нем команду **Link Folder to repository folder** — установить связь выбранного каталога с каталогом депозитория. Видим, что физического перемещения выбранного каталога в депозиторий не происходит — достаточно установить его связь с депозиторием через систему ссылок.
5. Открывается диалоговое окно **SVN Link Setup** (рис. 12.4).

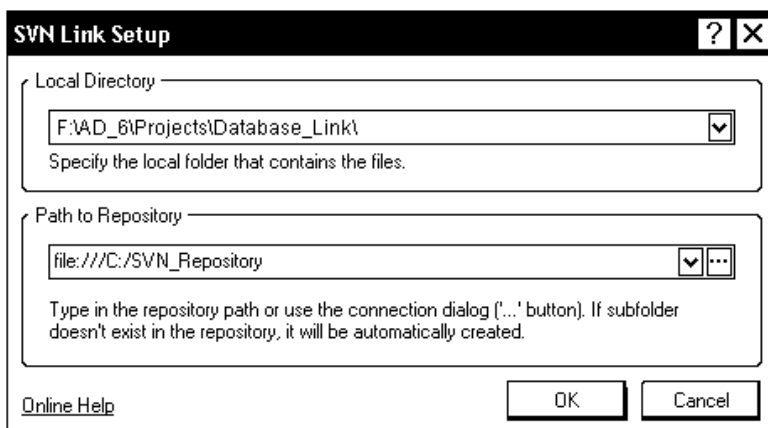


Рис. 12.4. Настройка связи документов проекта с депозиторием системы контроля версий

- В поле **Local Directory** ввести путь к каталогу, выбранному на третьем шаге. Убедиться, что именно в нем подлежат контролю версии проектных документов.
 - В поле **Path to Repository** отображается путь к каталогам депозитория системы контроля версий.
6. Щелчком на кнопке **ОК** завершить подключение проекта к системе контроля версий. Обозначение пути к выбранному депозиторию переносится в нижнюю строку правого поля панели **Storage Manager**. Слева от него программа ставит метку в виде звеньев цепочки, являющуюся признаком установления связи.

12.2.4. Подключение файлов к депозиторию

Для того чтобы окончательно передать документы выбранного каталога под контроль системы управления версиями, нужно выполнить действия:

1. Выбрать один или несколько необходимых файлов в панели **Storage Manager**.
2. Щелчком правой кнопки мыши открыть контекстное меню и указать в нем команду **Add to Version Control**.
3. Добавить комментарий, поясняющий причину подключения файлов к системе контроля, и закрыть диалог щелчком на кнопке **OK**.

В колонке **Status** правого поля панели **Storage Manager** появляется метка в виде "птички" белого цвета поверх зеленого кружка и выводится сообщение **No modification** — модификации не было (оно и естественно: файлы только что подключены к системе контроля).

12.2.5. Работа с проектными документами

Режим работы с документами под контролем системы управления версиями определяется, с одной стороны, правилами, установленными в самой системе контроля, а с другой — политикой, принятой в фирме. Сюда могут относиться такие требования, как допуск к документу только одного или сразу нескольких исполнителей, возможность обращения к уже открытому документу, требование возвращать отредактированные документы в депозиторий в определенные моменты времени, например, в конце рабочего дня и т. п. В нашей книге коснемся только технической стороны — порядка обращения к документу, хранящемуся в депозитории, возврата или удаления документа из депозитория.

- Для обращения к документу, хранящемуся в депозитории, нужно щелчком правой кнопки мыши на имени файла в панели **Storage Manager** вызвать контекстное меню и указать в нем команду **Update**. Если к этому моменту в рабочем каталоге исполнителя уже идет работа с документом, программа извлекает из депозитория последнюю сохраненную версию документа, сравнивает ее с рабочей копией исполнителя и помещает эту

версию документа в рабочий каталог исполнителя. Если рабочая копия исполнителя полностью совпадает с извлеченной из депозитория, ничего не происходит, а программа закрывает доступ к команде **Update**.

В левом нижнем поле **VCS Revisions**¹ выводится список заданного при настройке **DXP | Preferences** числа редактировавшихся прежде версий документа с указанием даты и времени соответствующего обновления, имени исполнителя, вносившего изменения, и комментариев. Версия, вызванная из депозитория (по умолчанию это последнее сохраненное обновление), отмечается стрелкой голубого цвета.

- Для обращения к любой промежуточной версии документа следует указать курсором эту версию в списке поля **VCS Revisions**, вызвать щелчком правой кнопки мыши контекстное меню и выполнить в нем команду **Open** (Открыть). Выбранная версия открывается в Altium Designer.
- Для возврата отредактированного документа обратно в депозиторий следует щелчком правой кнопки мыши на имени документа в панели **Storage Manager** вызвать контекстное меню и указать в нем команду **Commit** (Вернуть документ). При передаче документа в депозиторий ему присваивается следующий номер версии. Возврат документа обусловлен рядом требований:
 - если документ в этот момент был открыт в среде Altium Designer, перед возвратом в депозиторий его нужно сохранить в файловой системе, в противном случае в депозиторий будет отдана неотредактированная версия;
 - если попытка возврата происходит в момент, когда над документом работает еще один исполнитель, программа требует сначала выполнить обновление командой **Update**, а затем уже разрешает возврат документа;
 - при использовании систем контроля SVN или CVS и работе с двоичными документами, такими как схемный или PCB-документ Altium Designer обновление документа командой **Update** не выполняется. Вместо сравнения система контроля версий выдает из депозитория сохраненную версию документа, добавляя к имени файла знак #. Исполнителю приходится самому сравнивать извлеченную версию со

¹ VCS — от Version Control System — система управления версиями.

своей рабочей копией документа и принимать решение, что делать с вариантами документа.

- Сравнение версий схемного документа или PCB-документа при использовании систем контроля версий CVS или SVN выполняется следующим образом:
 - указать в поле **VCS Revisions** панели **Storage Manager** две версии документа, выполняя щелчок левой кнопкой мыши при нажатой клавише <Ctrl>;
 - щелчком правой кнопки мыши открыть контекстное меню и указать в нем команду **Compare** (Сравнить).

Программа проводит сравнение версий документа и открывает панель **Differences** (Различия), в которую выводит в форме дерева список обнаруженных отличий (**Differences Detected**).

12.3. Библиотеки под контролем версий

Как и другие документы, библиотеки Altium Designer, в том числе и связанные с внешними базами данных, могут работать под контролем систем управления версиями¹. Для обеспечения возможности работы под контролем версий в Altium Designer формируется еще один вид управляющего файла — так называемой SVNDBlib-библиотеки (файла *.SVNDBlib). Это такой же файл связи с внешней базой данных, как рассмотренная DBLib-библиотека, с той разницей, что символы ассоциированной схемной библиотеки (*.SchLib) и библиотеки топологических моделей (*.PcbLib) хранятся в депозитории системы контроля версий. Еще одно отличие — в версиях Altium Designer вплоть до 6.9.xxxxx библиотека SVNDBlib не поддерживает работу с трехмерными моделями компонентов (PCB3D) и моделями, обеспечивающими схемотехническое моделирование.

Здесь различают подходы разработчика библиотек и разработчика проектов. Далее рассмотрим особенности обоих подходов.

¹ Документ AP0145 Working with Version-Controlled Database Libraries.pdf Application Note AP0145 (v1.1) June 14, 2007.

12.3.1. Ведение библиотек

Разработчик библиотек кроме формирования непосредственно библиотеки схемных символов и посадочных мест должен:

- ❑ образовать библиотечную структуру *.SVNDBLib, подчиненную системе контроля версий, и установить связь этих библиотек с внешней базой данных;
- ❑ определить ключевую запись в таблице для привязки параметров и моделей компонента;
- ❑ задать связь библиотечных элементов с депозиторием системы контроля версий.

12.3.2. Расщепление библиотек

Важная особенность библиотек, которые отдаются под контроль системы управления версиями, — наличие отдельного библиотечного файла для каждого символа и каждой модели посадочного места. Если этого нет (такое тоже возможно), то при модификации единственного символа или модели посадочного места придется образовать новую версию библиотеки в целом. При этом неоправданно раздувается число версий библиотек: вместо нескольких версий каждого модифицированного компонента придется держать в депозитории столько версий большой библиотеки, сколько компонентов и сколько раз каждый из них подвергался модификации.

Если библиотеки формируются с чистого листа, никаких проблем это не вызывает. Однако часто библиотеки компонентов ранее уже были сформированы. На этот случай Altium Designer содержит функцию дробления библиотек на отдельные библиотечные файлы для каждого символа или модели. Расщепление библиотек выполняется в несколько шагов в среде мастера **Library Splitter Wizard**.

1. Активизировать функции мастера командой главного меню программы **Tools | Library Splitter Wizard**. Откроется диалоговое окно — первая страница мастера (рис. 12.5).
2. Щелчком на кнопке **Add** запустить поиск библиотек, подлежащих расщеплению. В окне строится иерархическая структура из двух ветвей — для библиотек схемных символов и посадочных мест.
3. Кнопкой **Remove** выбранные библиотеки можно удалить из списка.

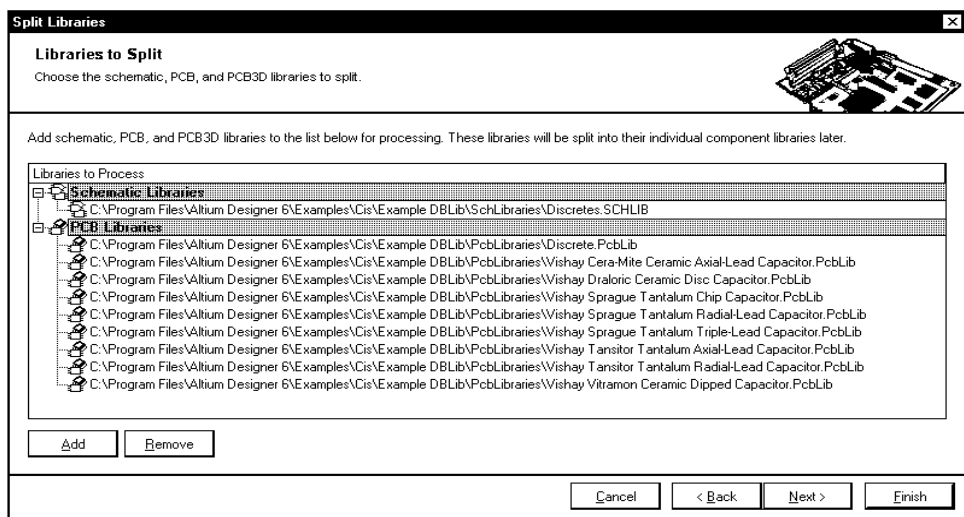


Рис. 12.5. Мастер расщепления интегрированных библиотек: объявление списка библиотек, подлежащих расщеплению

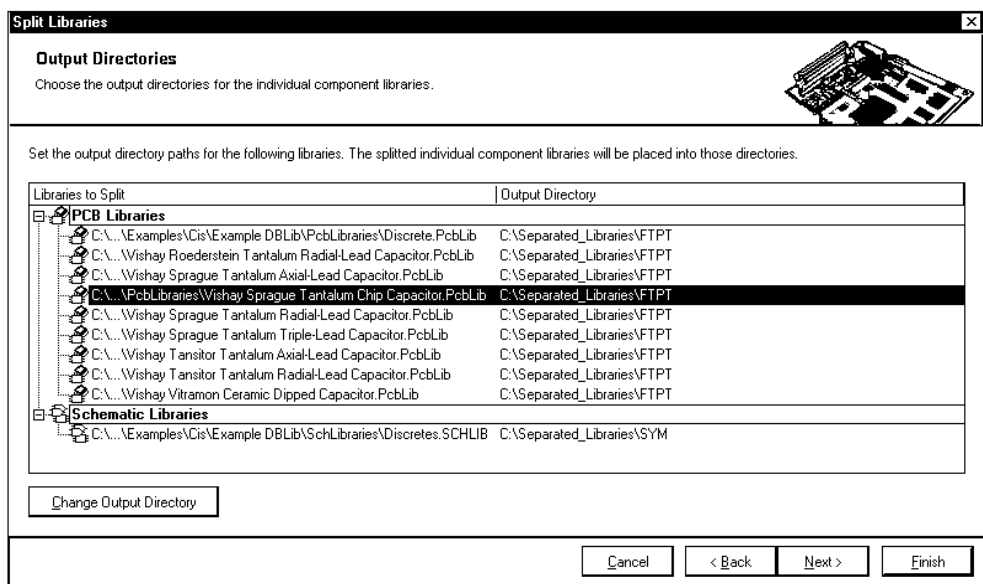


Рис. 12.6. Мастер расщепления интегрированных библиотек: объявление выходного каталога для расщепленных библиотек

4. Щелчком на кнопке **Next** перейти на вторую страницу мастера (рис. 12.6). На данной странице к списку выбранных библиотек (**Libraries to Split**) добавляется список каталогов для размещения расщепленных библиотек (**Output Directory**). По умолчанию предлагается разместить расщепленные библиотеки в тех же каталогах, где находятся исходные библиотеки.

При необходимости кнопкой **Change Output Directory** следует запустить стандартную процедуру поиска новых путей для размещения результатов расщепления.

5. Щелчком на кнопке **Next** перейти на следующую страницу (рис. 12.7). На данной странице выбрать одну из двух опций на случай, если назначенный выходной каталог уже содержит библиотечные файлы:

- **Overwrite Existing Files** — заменить существующие библиотечные файлы новыми, если совпадают их имена;
- **Append Incrementing Numbers to File Names** — добавить к именам файлов с совпадающими именами инкрементируемые по мере присоединения новых библиотек суффиксы "_n", где n — числа, начиная с единицы.

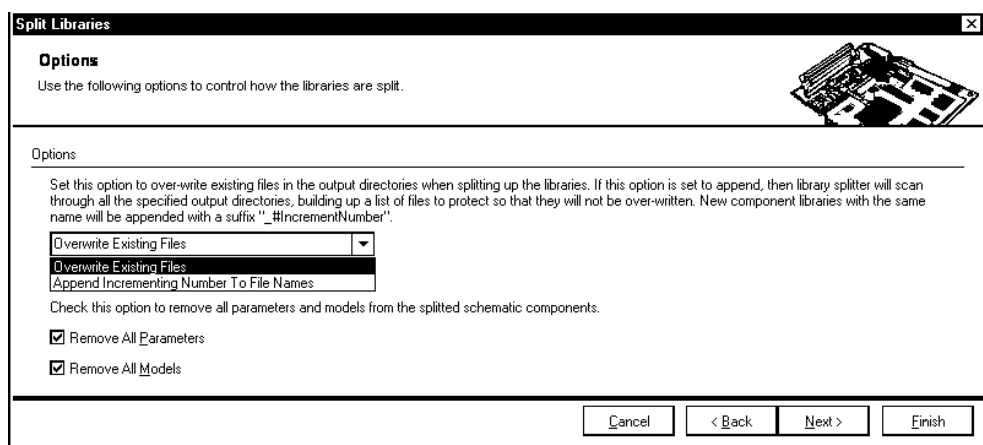


Рис. 12.7. Мастер расщепления интегрированных библиотек: объявление способа расщепления и особенностей размещения результатов

Кроме того, установкой флажков **Remove All Parameters** (Удалить все параметры) и **Remove All Models** (Удалить все модели) можно удалить из расщепленных символьных библиотек все параметры и все ссылки на мо-

дели для того, чтобы в них остались только чистые символы, а параметры и ссылки на модели присоединялись из внешней базы данных.

- Щелчком на кнопке **Next** открыть следующую страницу (рис. 12.8), на которой выполняется анализ исходных библиотек и формируется список имен расщепленных библиотечных файлов.

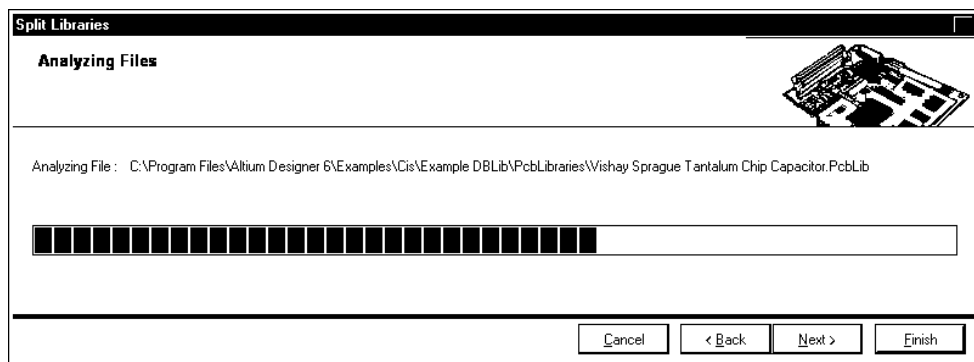


Рис. 12.8. Мастер расщепления интегрированных библиотек: отображение прогресса расщепления

- Список имен расщепленных библиотек отображается на следующей странице мастера (рис. 12.9). Слева направо в диалоговом окне **Review** располагаются колонки:
 - **Output Directory** — выходной каталог;
 - **Splitted Library Name** — имя расщепленной библиотеки: библиотеке присваивается имя, повторяющее имя символа исходной библиотеки, при этом символы "точка", "восклицательный знак", "апостроф", открывающая и закрывающая квадратные скобки, если они присутствуют в исходных именах, заменяются на знак подчеркивания "_".
 - **Component Name** — имя компонента;
 - **Source Library** — полный путь к исходным библиотекам.
- Кнопкой **Next** активизировать заключительную страницу мастера, на которой выполняется обработка подготовленных данных — окончательно формируются расщепленные библиотеки. В диалоговом окне, аналогичном рис. 12.8, отображается линейка прогресса и имена файлов, конвертируемых в каждый текущий момент.
- Щелчком на кнопке **Finish** закрыть работу мастера расщепления.

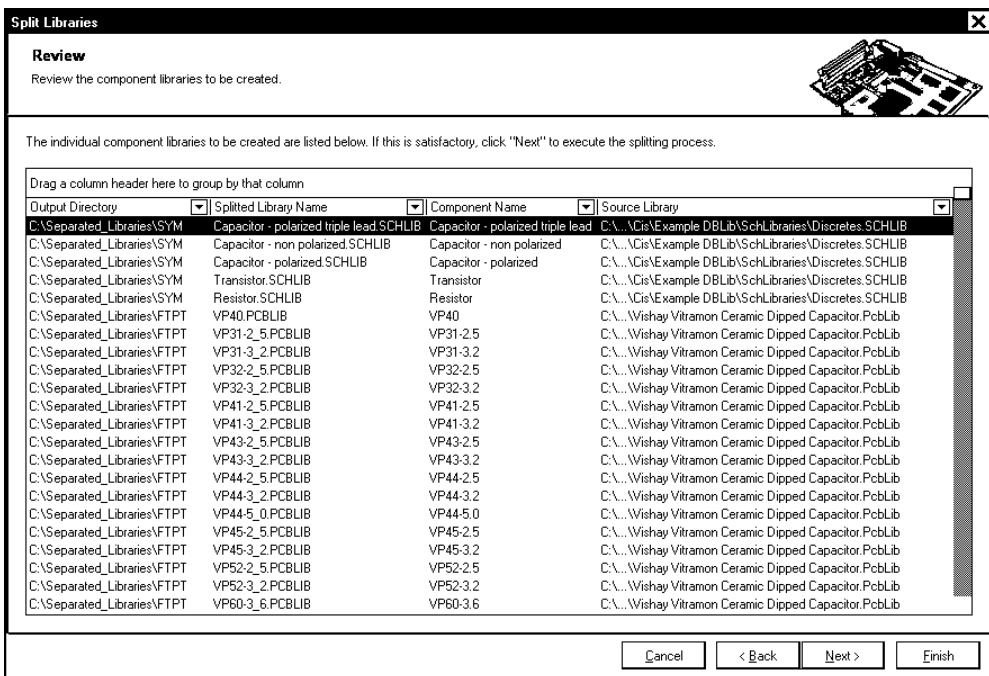


Рис. 12.9. Мастер расщепления интегрированных библиотек: отображение результатов расщепления

12.3.3. Образование библиотеки SVN Database Library

Порядок действий таков.

1. Для образования библиотечного файла-оболочки *.SVNDBLib выполнить команду главного меню **File | New | Library | SVN Database Library**. Откроется оболочка библиотечной структуры с именем файла по умолчанию SVNDatabase_Libs1.SVNDBLib, полностью аналогичная оболочкам **Data-base Link file** и **Database Library file** (см. рис. 11.3 и 11.10).
2. Аналогично структурам *.DBLink и *.DBLib (см. рис. 11.1 и 11.10) установить связь структуры *.SVNDBLib с внешней базой данных.
3. Командой главного меню **Tools | Options** или командой **Options** из контекстного меню, вызываемого щелчком правой кнопки мыши, активизировать диалог установления связи библиотечной структуры SVNDBLib с

депозиториум системы контроля версий. Откроется диалоговое окно (рис. 12.10), на вкладке **SVN Repository** которого следует назначить:

- **Location** — размещение депозитория в файловой системе компьютера или на сетевом ресурсе;
- **Models Location** — путь к каталогам символьных библиотек (Symbols) и библиотек посадочных мест (Footprints) в депозитории системы контроля версий.

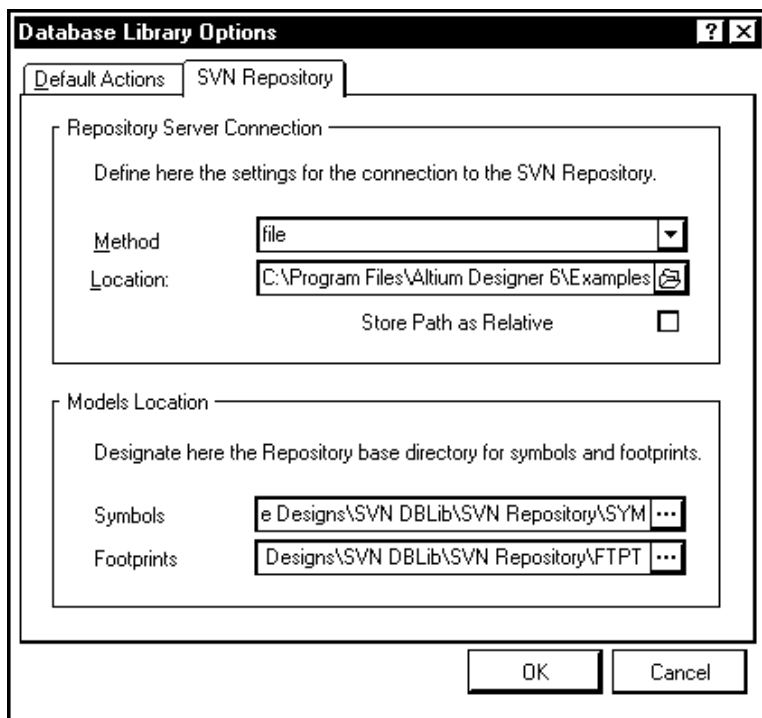


Рис. 12.10. Окно настройки связей библиотечной структуры SVNDBLib с депозиторием системы контроля версий

Следует отметить, что эти библиотеки должны размещаться в корневом каталоге или подкаталогах депозитория. При этом путь к ним нужно указать в полях внешней базы данных и эти поля (**Database Field Names**) связать на вкладке **Field Mappings** в оболочке библиотечной структуры SVNDBLib с параметрами проекта [Library Path] и [Footprint Path] в колонке **Design Parameters**. Можно задать абсолютный (полный) или относительный (относи-

тельно корневого каталога депозитория) путь, имя файла или не указывать путь вовсе. В зависимости от способа указания пути изменяются порядок и область поиска компонента в библиотеках. Если пути в записях базы данных не обозначены, это делает всю структуру поиска даже более устойчивой: при изменении расположения библиотек символов и посадочных мест в депозитории нет необходимости исправлять ссылки в полях базы данных.

12.3.4. Редактирование таблиц базы данных из оболочки SVNDBLib

В отличие от структур DBLink и DBLib, оболочка SVNDBLib позволяет вносить изменения в записи базы данных, добавлять новые записи, не открывая для этого файл внешней базы данных, открывать и редактировать библиотечные файлы.

- Щелчком правой кнопки мыши в выбранной строке активной таблицы открыть контекстное меню (рис. 12.11) и указать в нем одну из команд:
 - **Add New Component** — добавить новый компонент (новую строку) в таблицу;
 - **Delete Component** <Part Number> — удалить из таблицы выбранную строку;
 - **Edit Component** <Part Number> — отредактировать выбранный компонент (записи в строке таблицы);
 - **Open Symbol** <Library Ref> — открыть файл символьного библиотечного элемента;
 - **Open Footprint** <Footprint Ref> — открыть файл библиотеки посадочного места;
 - **Options** — активизировать диалог настройки связи с депозиторием системы контроля версий (см. рис. 12.10).

При запуске команды редактирования компонента активизируется диалоговое окно (рис. 12.12) со списком полей таблицы базы данных в колонке **Field Name** и записей фигурирующих в этих полях в колонке **Field Value**.

- Для редактирования выбрать строку в этом списке, внести в поле значений **Field Value** новую запись и щелчком на кнопке **OK** завершить операцию.

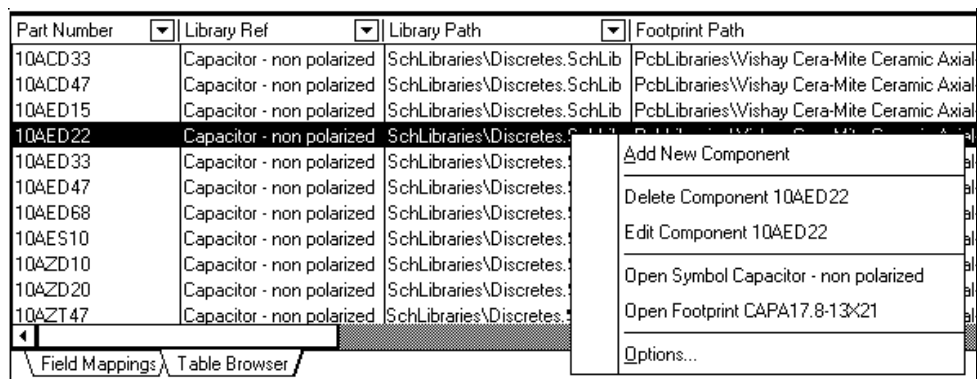


Рис. 12.11. Редактирование таблиц базы данных из оболочки SVNDBLib

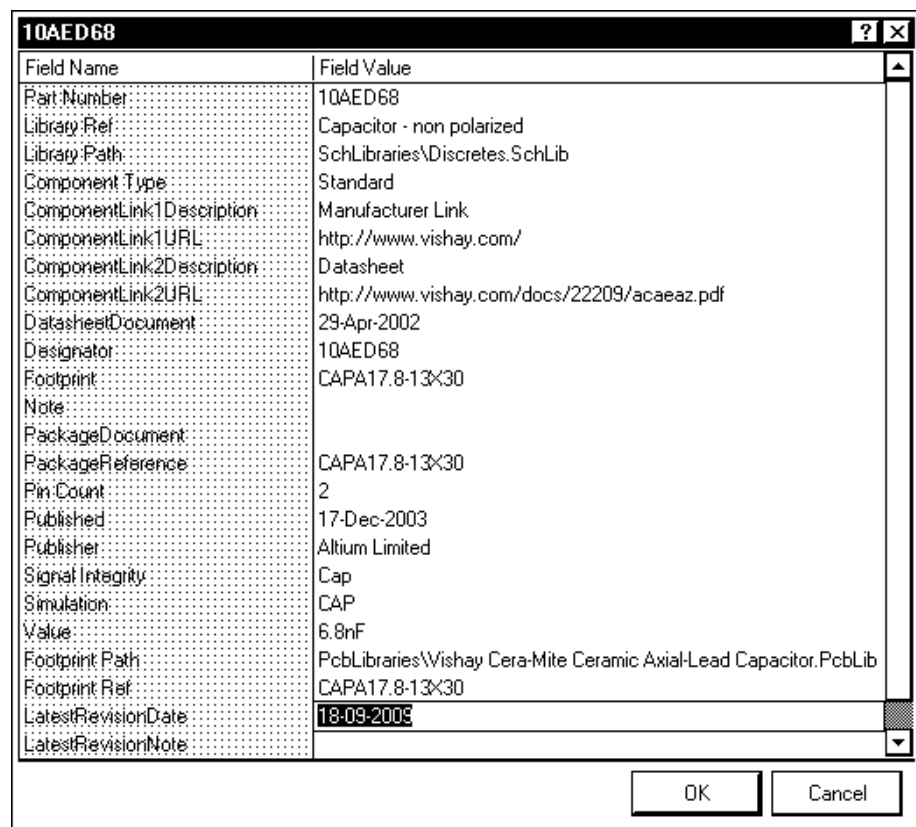


Рис. 12.12. Окно редактирования параметров компонента в базе данных

При выполнении команды **Add Component** (Добавить компонент) активируется то же самое диалоговое окно, но с пустым полем **Field Value**. После заполнения записей в этом поле новая строка включается в таблицу библиотеки **SVNDBLib**.

3. Редактирование вызываемых из библиотеки схемных символов и моделей посадочных мест выполняется во временной рабочей папке, определенной при настройке ветви **SVN Libraries** в диалоге **DXP Preferences** (рис. 12.13).

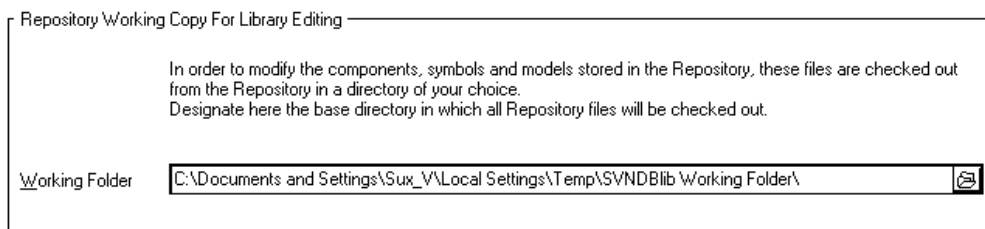


Рис. 12.13. Определение рабочей папки системы контроля версий

Интерфейсом между библиотечными файлами, вызванными в рабочий каталог, и депозиторием системы управления версиями служит панель рабочего пространства **Storage Manager**. Операции вызова документов из депозитория, контроля и сравнения версий и возврата в депозиторий, рассмотренные в *разд. 12.2*, полностью относятся и к библиотечным документам.

ЗАМЕЧАНИЕ

Следует, правда, отметить, что в то время, как вызов библиотечных документов для редактирования может выполняться не только из панели **Storage Manager**, но также из оболочки **SVNDBLib**, из панели **Libraries**, командой **File | Open**, а также с помощью техники **Drag-and-Drop** из структуры файлов **Windows Explorer** непосредственно в **Altium Designer**, возврат отредактированных документов в депозиторий возможен только из панели **Storage Manager**.

4. Перед тем как возвращать отредактированные документы в депозиторий необходимо командой **Refresh Status** из контекстного меню панели **Storage Manager**, активизируемого щелчком правой кнопки мыши, обновить статус связей отредактированной версии в рабочем каталоге и версии,

хранящейся в депозитории. Статус связей документов отображается условными значками и текстовой строкой в поле панели **Storage Manager**:

- **No modification** — документ возвращен в депозиторий, а версия, находящаяся в рабочем каталоге исполнителя, с ним полностью совпадает (значок в виде "птички" на зеленом кружке);
- **Out of date** — документ возвращен в депозиторий, а версия, находящаяся в рабочем каталоге исполнителя, с ним не совпадает (вопросительный знак на желтом кружке) — следует выполнить операцию восстановления **Update**;
- **Modified** — документ изменен и находится в рабочем каталоге исполнителя (восклицательный знак на красном кружке) — следует вернуть отредактированную версию в депозиторий;
- **Conflict** — конфликт; документ одновременно модифицирован другим исполнителем (восклицательный знак на желтом треугольнике);
- **Missing** — документ хранится в депозитории, но отсутствует в рабочем каталоге исполнителя (красный крестик).

Этими же метками отмечаются файлы, находящиеся под контролем системы управления версиями, в панели **Projects**.

5. После выполнения необходимых обновлений связей документов отредактированный документ нужно сохранить и возвратить в депозиторий командой **Commit**. Статус приобретает значение **No modification**.

12.3.5. Использование библиотек разработчиком проектов

В крупных фирмах, где ведение библиотек компонентной базы выполняет специализированное подразделение, разработчик проектов получает в свое распоряжение готовые библиотеки, в том числе структуры *.DBLib или *.SVNDBLib, связанные с базой данных предприятия. Для работы с этими библиотечными структурами необходимо включить их в состав доступных библиотек и использовать панель **Libraries** как окно просмотра связанной внешней базы данных и поиска в ней необходимых компонентов.

Технологическая особенность библиотечной структуры *.SVNDBLib состоит в том, что библиотеки схемных символов и моделей посадочных мест компонентов, хранящиеся в депозитории системы контроля версий, помещаются на

компьютере разработчика проекта во временный рабочий кэш-каталог. Требуемые символы схемных библиотек и образы посадочных мест компонентов извлекаются в проект из этой кэш-памяти.

При настройке **DXP Preferences** на странице **SVN Libraries** (рис. 12.14) выводится комментарий, поясняющий порядок использования и обновления библиотек.

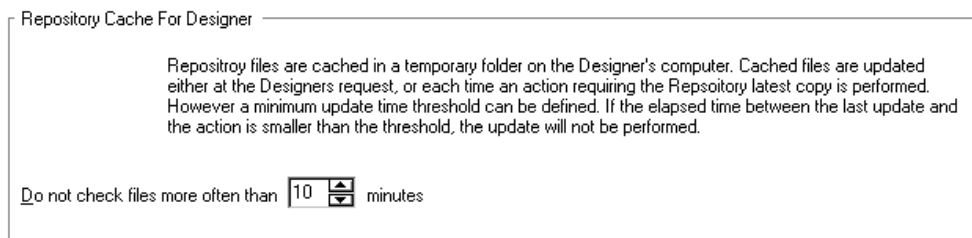


Рис. 12.14. Определение пользовательского кэш-каталога для временного размещения компонентов SVNDBLib-структуры

Библиотеки обновляются автоматически при выполнении следующих действий:

- ☐ при включении компонента SVNDBLib-библиотеки в проект из панели **Libraries**;
- ☐ при передаче схемного документа проекта на печатную плату;
- ☐ при выполнении команды **Update From Libraries** в схемном документе проекта;
- ☐ при выполнении команды **Update From PCB Libraries** в PCB-документе печатной платы.

Кроме того, в том же поле диалога настройки (рис. 12.14) можно назначить минимальный интервал времени, в течение которого автоматическое обновление запрещается.

Кроме автоматического возможно также принудительное обновление библиотек вручную, командой **Refresh** (Обновить) из контекстного меню панели **Libraries**.

Такое же принудительное обновление происходит, если исключить SVNDBLib-библиотеку из состава доступных библиотек панели **Libraries**, а затем включить ее обратно.

* * *

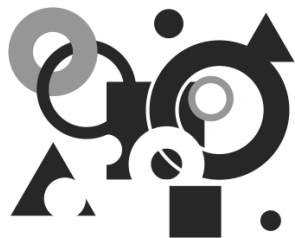
Рассмотренные функции Altium Designer позволяют эффективно применять внешние базы данных для составления схемных документов и выполнения проекта функционального узла РЭС. Однако для полноценного использования этих возможностей в отечественных организациях, разрабатывающих проектную документацию по ЕСКД, нужно предусмотреть ряд организационно-технических мероприятий.

- ❑ Текстовые документы, которые формируются с помощью заложенных в импортные САПР функций генерирования отчетов (Reports), нельзя без существенной переработки использовать как текстовые конструкторские документы, соответствующие стандартам ЕСКД. Известны отечественные разработки программ-генераторов перечней элементов, спецификаций и других текстовых конструкторских документов по ЕСКД, извлекающих необходимые данные из файлов-отчетов (так называемых "списков материалов" — Bill Of Materials, BOM) и выстраивающие эти данные в документ, соответствующий требованиям ЕСКД¹. Следовательно, фирма, разрабатывающая проекты узлов РЭС, должна приобрести такие программы и установить их на своем оборудовании.
- ❑ Чтобы данные, необходимые для формирования документов по ЕСКД, могли попасть в BOM-файл, они должны присутствовать в составе параметрической информации компонентов. Если компоненты проекта поступают из внешней базы данных через DBLib-библиотеку, поля базы данных должны содержать сведения о компонентах, позволяющие составить записи в перечень элементов к схеме электрической принципиальной и в спецификацию сборочного узла по правилам, установленным для этих документов, а также в соответствии с примерами, приводимыми в ведомственных справочных листах.
- ❑ При связи с базой данных посредством формирования DBLib-библиотеки с каждой из этих внешних баз данных нужно ассоциировать библиотеки схемных символов, моделей посадочного места, трехмерных моделей компонентов в формате Altium Designer. Ссылки на имена моделей и указатели путей к этим библиотекам следует также занести в поля базы данных.

¹ См. материалы И. В. Брагина на интернет-сайте www.verzak.ru.

- ❑ При наличии в проекте сразу нескольких способов связи с базами данных (через Database Link File, структуры DBLib и/или SVNDBLib) могут возникать конфликты, если эти структуры связаны с разными базами данных, а имена компонентов в этих базах совпадают. Последнее обстоятельство требует серьезных усилий по администрированию баз данных и компьютерных сетей предприятия.

Таким образом, структура записей базы данных и ассоциированных с ней библиотек компонентной базы оказывается существенным образом завязана на формат и структуру данных Altium Designer и их физическое размещение в файловой системе конкретного компьютера, в компьютерной сети конструкторского подразделения или на других сетевых ресурсах.



Экспорт результатов проекта

Разработанный интегральный графический образ функционального узла на печатной плате нуждается в дополнительной обработке, чтобы сформировать комплект конструкторских документов на электронном или бумажном носителе, иллюстраций для научно-технических отчетов и статей, а также файлы управляющей информации для автоматизированного производственного оборудования — данные для механической обработки контура платы, сверления отверстий, формирования послойных фотошаблонов, роботизированной установки компонентов.

Altium Designer располагает обширным набором функций преобразования формата данных графического редактора печатной платы. Команды управления всеми этими функциями сгруппированы в главном меню **File**.

13.1. Экспорт в AutoCAD

Порядок действий таков.

1. По команде главного меню **File | Save As** активизировать стандартный диалог сохранения данных с измененным именем и форматом (типом) файла. Из 12 имеющихся форматов нас в данный момент интересует формат **DWG/DXF**, дающий возможность открывать сохраненный файл и обрабатывать данные в среде конструкторской САПР AutoCAD. Средства управления видами AutoCAD позволяют легко получить из интегрального образа печатной платы два рабочих конструкторских документа — чертеж печатной платы по ГОСТ 2.417-91 и сборочный чертеж функционального

узла по ГОСТ 2.109-73. Щелчком на кнопке **ОК** закрыть фазу назначения имени и типа экспортного файла. Открывается окно настройки параметров экспортируемого файла **Export to AutoCAD** (рис. 13.1).

2. В поле **Options** окна **Export to AutoCAD** указать:

- **AutoCAD Version** — версию AutoCAD: возможен экспорт в формат ранних версий, из которых самая старшая R14 (1999 г.), что не мешает принимать данные экспорта в современные версии, вплоть до 2008;
- **Format** — формат данных: возможен выбор двоичного формата DWG или текстового обменного формата DXF;
- **Units** — выбор дюймовой (Imperial) или метрической (Metric) системы единиц измерения.

3. В поле **Holes** (Отверстия):

- активизировать опцию **Export Pads and Via Holes** — экспорт графических изображений монтажных отверстий КП и межслойных переходных отверстий (ПО);
- выбрать способ формирования отверстий — на специально отведенных слоях (**On dedicated hole layers**) или на тех же слоях, что и металлизация КП и ПО (**On pad and via layers**). В дальнейшем увидим, что в первом случае для отверстий образуются отдельные слои **Pad-hole** и **Viahole**, и отверстия показываются на чертеже черными кружками, а во втором случае отверстия имеют естественный вид.

4. В поле **Components** указать способ представления компонентов: в виде блоков AutoCAD (**Export as blocks**) или графических примитивов (**Export as primitives**).

5. В поле **Tracks and Arcs** (Дорожки и дуги) указать способ построения сегментов печатных проводников: с закругленными (**Export with rounded ends**) или прямоугольными (**Export with square ends**) концами сегментов.

6. В поле **Primitives To Export With Zero Line Widths** указать, какие примитивы должны изображаться линиями нулевой ширины: не изображать никаких (**None**), все (**All**) или выбранные (**Selected**).

7. Щелчком на кнопке **ОК** завершить настройку. Программа генерирует экспортный файл, с которым в дальнейшем предстоит работать в среде AutoCAD (см. главу 14).

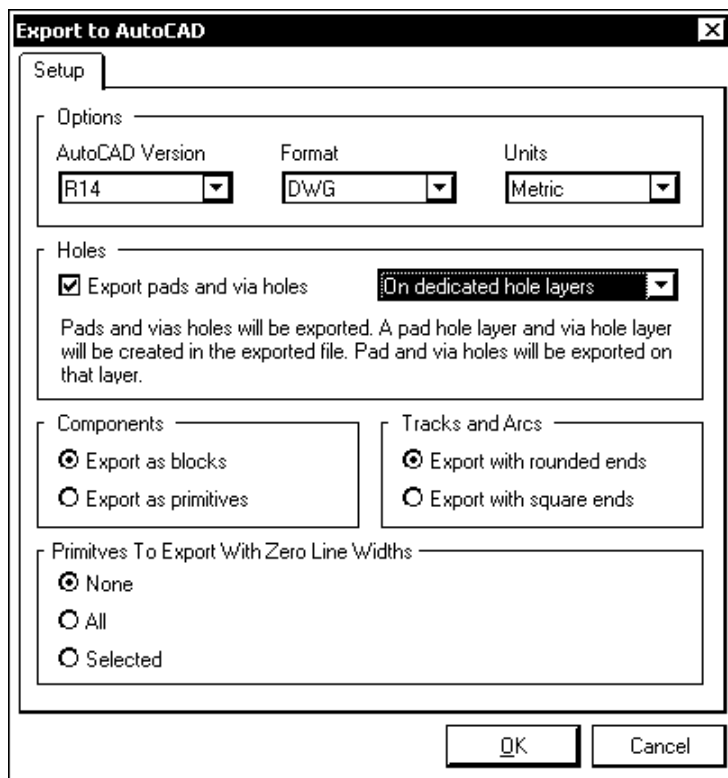


Рис. 13.1. Диалог настройки параметров экспорта в AutoCAD

13.2. Экспорт в PDF-формат Adobe Acrobat

Может быть полезен для составления иллюстраций к статьям, учебным пособиям, техническим отчетам и т. п. При экспорте кроме графической информации передаются элементы структуры исходного документа.

1. Активизировать команду меню **File | Smart PDF**. Открывается процедура мастера экспорта (Export Wizard) **Altium Designer Smart PDF**.
2. На шаге **Choose Export Target** указать, какие объекты подлежат экспорту — текущий проект со всеми входящими в него документами, свободные документы или текущий открытый документ, а также принять пред-

лагаемый программой или назначить другой путь для сохранения результата.

- На шаге **PCB Printout Settings** (рис. 13.2) настроить видимость слоев и объектов на выбранных слоях в результирующем PDF-документе. Слой указывают курсором, после чего двойным щелчком левой кнопки мыши вызывают диалог настройки видимости объектов на слое.

Характер изображения — **Full** (Полное изображение), **Draft** (Набросок — изображение контуров тонкой линией) или **Off** (Выключено) — задают для выбранных типов объекта в полях, отведенных для каждого типа, либо кнопками **Full**, **Draft** или **Off** — для всех объектов в пределах каждой из трех категорий — **Free Primitives** (Свободные примитивы), **Component Primitives** (Примитивы компонентов) и **Other** (Прочие).

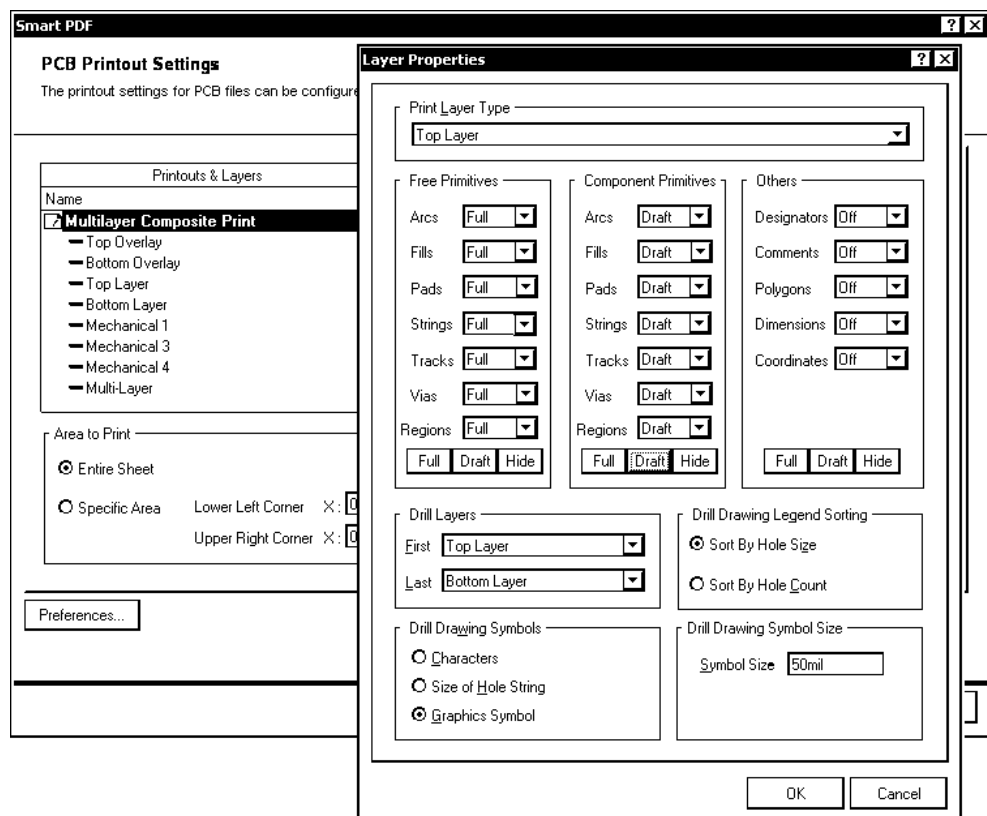


Рис. 13.2. Диалог настройки экспорта в формате Adobe PDF

4. Щелчком на кнопке **Preferences** открыть диалог настройки цветовой палитры или тона "серой шкалы" для монохромного представления результатов экспорта.
5. На заключительных шагах выполняется еще ряд настроек, которые на начальной стадии освоения пакета можно опустить. Процедура завершается щелчком на кнопке **Finish**, после чего открывается программа Acrobat Reader, в окне которой изображается экспортированный образ печатной платы.

В Acrobat Reader доступны средства навигации **Bookmarks** (Вкладки) и **Thumbnails** (Страницы), позволяющие просмотреть подробности в полученном графическом образе. В поле **Bookmarks** располагается дерево структуры графического образа, в котором могут быть развернуты ветви **Components** и **Nets**. Компоненты обозначаются позиционными обозначениями, а цепи — именами, назначенными в проекте Altium Designer. Ветвь **Nets**, в свою очередь, разворачивается в список узлов, принадлежащих выбранной цепи.

При указании курсором обозначения компонента или узла цепи выбранный объект показывается в главном окне Acrobat Reader в увеличенном масштабе.

13.3. Послойные распечатки проекта

Подкоманды главного меню **File | Fabrication Outputs | ...** (Файл | Выходные данные для производства) предназначены для формирования распечаток послойных видов проекта печатной платы (рис. 13.3):

- ❑ **Composite Drill Guide** — в окне предварительного просмотра **Preview Composite Drill Drawing of [<имя проекта>]** изображается карта сверления на всех парных слоях, сведенная в одну картинку, на которой показаны отверстия и обозначены крестиками их центры;
- ❑ **Drill Drawings** — в окне предварительного просмотра **Preview Drill Drawing/Guides of [<имя проекта>]** отображаются карты сверления на отдельных слоях: на верхнем слое пары показаны отверстия, а на нижнем слое — обозначения их центров;
- ❑ **Mask Set** — набор масок: в окне предварительного просмотра **Preview Solder/Paste Mask Prints of [<имя проекта>]** отображаются защитная (Solder) маска и маска-трафарет для нанесения припойной пасты (Paste) на верхнем и нижнем наружных сигнальных слоях печатной платы;

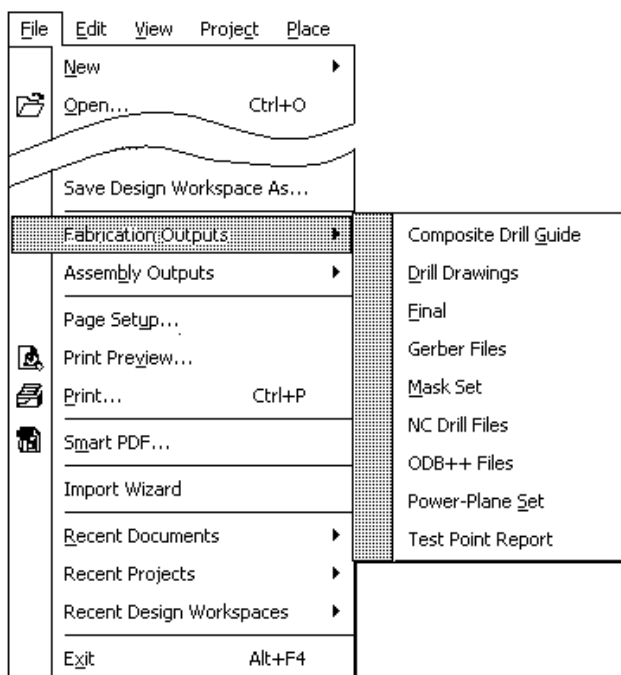


Рис. 13.3. Команды экспорта данных Altium Designer

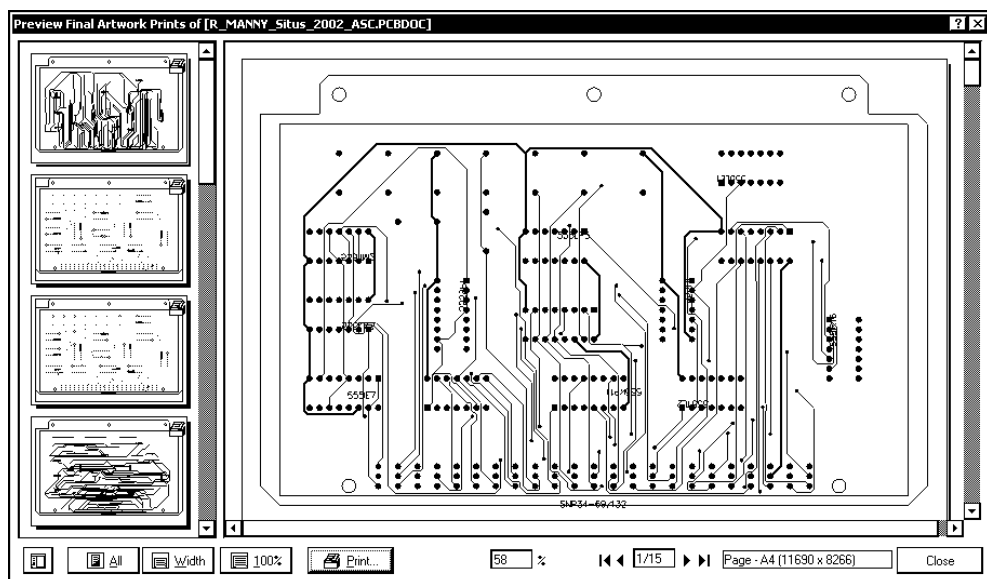


Рис. 13.4. Полная подборка послойных видов проекта

- ❑ **Power-Plane Set** — карта межслойных ПО, пронизывающих внутренние слои металлизации типа Plane;
- ❑ **Final** — полная подборка послойных видов проекта **Preview Final Artwork Prints of [<имя проекта>]** (рис. 13.4).

При активизации какой-либо из перечисленных команд формируется набор предварительных видов распечатки соответствующей группы слоев.

Панель навигации в левой половине окна позволяет выбрать для просмотра нужный слой.

Кнопка **Print** запускает печать всех подготовленных образов. При этом действуют настройки печати, назначенные по умолчанию. При необходимости эти установки можно изменить.

Порядок действий таков.

1. Щелчком правой кнопки мыши на имени проекта в плавающей панели **Projects** вызвать контекстное меню и активизировать в нем команду **Project Options**. Открывается диалоговое окно **Options for PCB Project <имя>.PrjPcb** (рис. 13.5).
2. В группе опций настройки **Fabrication Outputs** указать необходимый набор распечаток, например **Final Artwork Prints**; установить флажок на пересечении строки с колонкой таблицы **Default Print**.
3. Кнопкой **Configure** открыть диалог настройки конфигурации распечаток (рис. 13.6). Тот же результат достигается, если в окне предварительного просмотра (см. рис. 13.4) щелчком правой кнопки мыши вызвать контекстное меню и указать в нем команду **Configuration**.
4. В диалоговом окне (см. рис. 13.6) располагается список всех листов распечатки, активизированных исполняемой командой группы **Fabrication Outputs**.

Для каждого листа указывается состав слоев проекта, включенных в распечатку. Например, в распечатку верхнего слоя печатной платы Top Layer кроме собственно проводников верхнего сигнального слоя с тем же именем по умолчанию включаются контактные площадки компонентов, располагающиеся на слое Multi-Layer, и контур печатной платы на слое Mechanical 3.

5. Указать лист распечатки и установить флажки против объектов, включаемых в распечатку, в частности, отверстий (**Holes**).

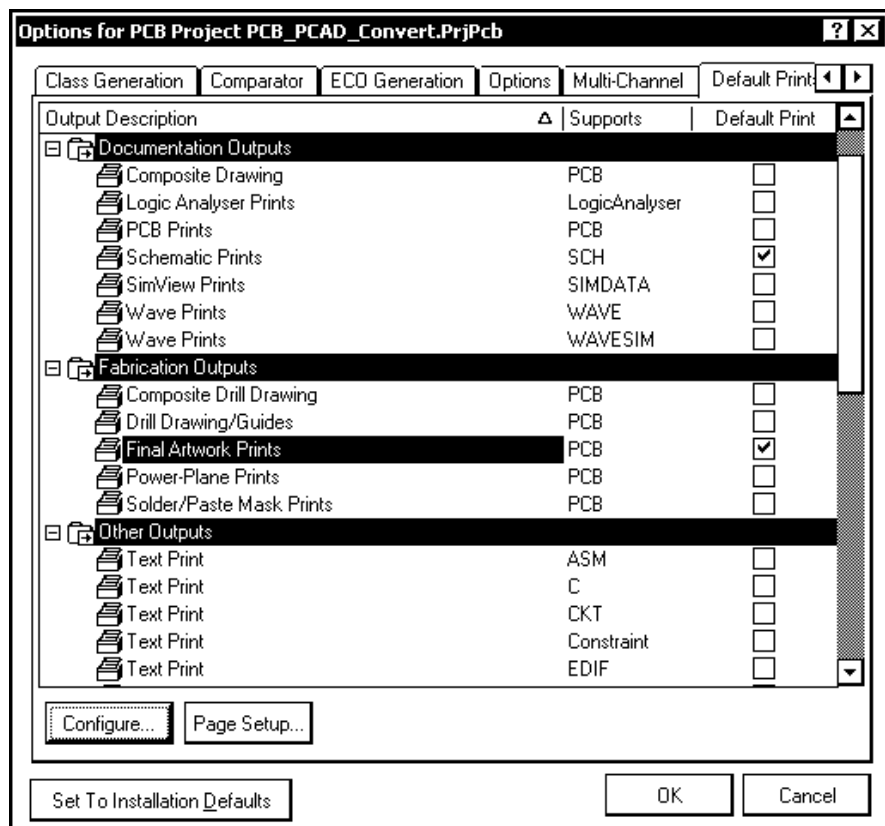


Рис. 13.5. Определение набора послойных распечаток проекта

- Двойной щелчок левой кнопки мыши активизирует окно, аналогичное изображенному на переднем плане рис. 13.2. Кнопки **Full**, **Draft** и **Hide** включают или отключают видимость указанного слоя в данном листе распечатки.
- Щелчок правой кнопки мыши в любом поле окна (рис. 13.6) открывает контекстное меню, в котором дублируются команды групп **File | Assembly Outputs** (Выходные данные для сборки) и **File | Fabrication Outputs**. При выборе этих команд текущая настройка распечаток теряется, и все управление конфигурацией распечаток начинается сначала. Кроме того, контекстное меню содержит команды:
 - Insert Printout** — добавить распечатку;
 - Insert Layer** — добавить слой в выбранную распечатку;

- **Delete** — удалить выбранную распечатку или выбранный слой в выбранной распечатке.

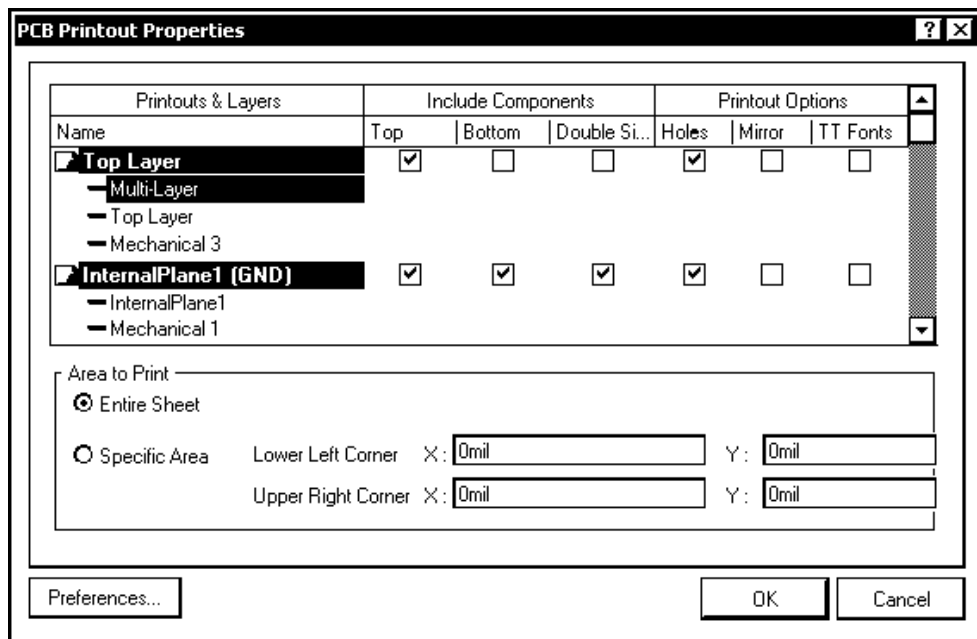


Рис. 13.6. Настройка конфигурации послойных распечаток

13.4. Формирование распечатки видов сборки узла

Команда главного меню **File | Assembly Outputs** (рис. 13.3) имеет две подкоманды: **Assembly Drawings** (Чертежи сборки) и **Generate Pick and Place Files** (Генерировать файлы точек подхвата и установки).

1. По команде **Assembly Drawings** в окне предварительного просмотра **Preview Assembly Drawings of [<имя_проекта>]** формируются и отображаются виды сборки функционального узла сверху и снизу. Конфигурация распечатки полученных видов настраивается так же, как это описывалось в предыдущем разделе.

- По команде **Generates Pick and Place Files** программа формирует файлы координат точек захвата компонентов для роботизированной сборки функционального узла.

13.5. Экспорт данных для сверления

Порядок действий таков.

- Активизировать команду главного меню **File | Fabrication Outputs | NC Drill Files** (см. рис. 13.3). Открывается диалог настройки формата экспортируемых данных (рис. 13.7).
- В поле **Units** диалогового окна (рис. 13.7) установить метрическую систему единиц измерения — **Millimeters**.
- В поле **Format** установить точность передачи координатной информации (число десятичных знаков до и после запятой). Этот формат нужно согласовать с изготовителем печатных плат, поскольку он определяется точностью и форматом входных данных контроллера того оборудования, которым располагает изготовитель.
- Из остальных настроек можно активизировать опцию раздельной генерации файлов для сверления металлизированных и неметаллизированных отверстий — **Generate separate NC Drill files for plated & non-plated holes**.

ЗАМЕЧАНИЕ

Отметим, что выходные данные для неметаллизированных отверстий генерируются только в том случае, если в проекте печатной платы эти отверстия имеют статус контактных площадок. Отверстий, которые просто начерчены на плате, программа "не видит".

- Щелчок мышью на кнопке **ОК** запускает генерацию выходных файлов.
- Программа генерирует несколько файлов.

Текстовый файл <Имя_проекта>.LDP, содержащий информацию о парах слоев сверления, установленных в текущем проекте (листинг 13.1).

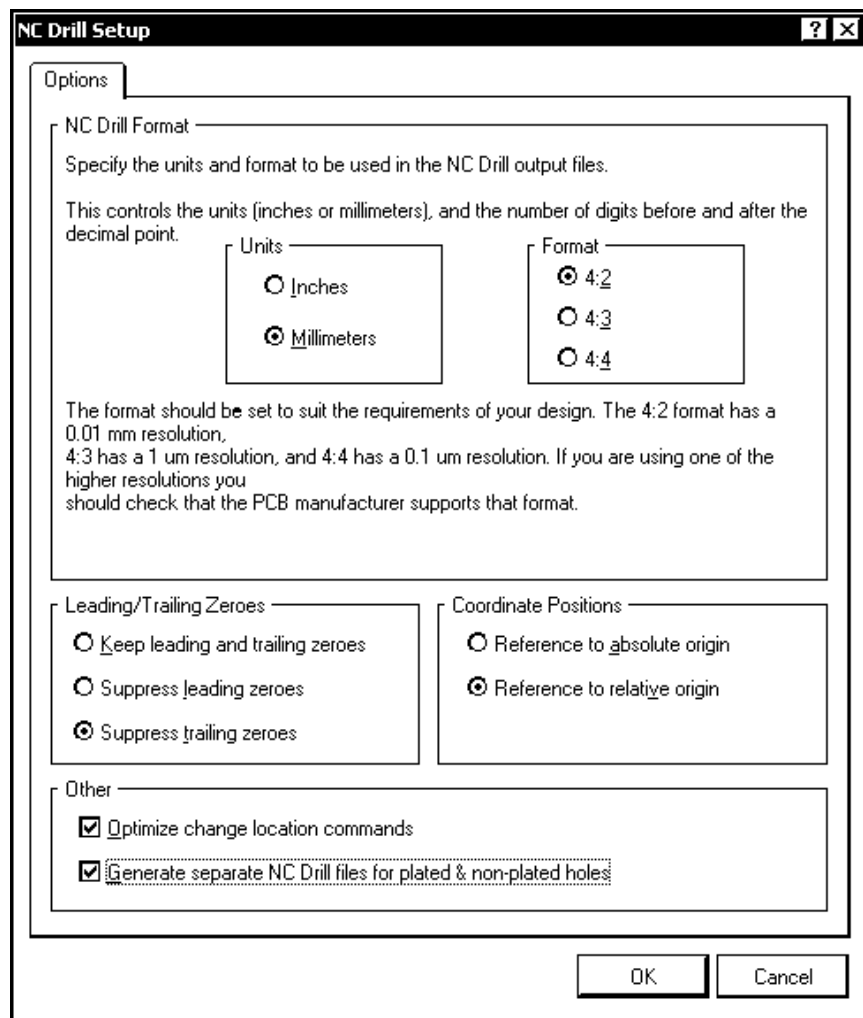


Рис. 13.7. Настройка формата данных
для сверления

Листинг 13.1

```
Layer Pairs Export File for PCB:  
F:\student\KURS2006\Project_PCB.PRJPCB\Project.PCBOC  
  
LayersSetName=Top_Bot_Plated_Thru_Holes|DrillFile=Project-  
plated.txt|LayerPairs=gt1,gb1
```

Текстовый файл отчета <Имя_проекта>.DRR, содержащий информацию о диаметрах и числе отверстий, условные обозначения инструментов (сверл), путь, пройденный рабочим органом станка (листинг 13.2).

Листинг 13.2

```
-----
---
NCDrill File Report For: R_MANNY_Situs_2002_ASC.PCBDOC    06.01.2008
17:18:12
-----
---
```

```
Layer Pair : Top Layer to Bottom Layer
ASCII Plated RoundHoles File : R_MANNY_Situs_2002_ASC-Plated.TXT
EIA File   : R_MANNY_Situs_2002_ASC.DRL
```

Tool	Hole Size	Hole Type	Hole Count	Plated	Tool Travel

T1 Inch)	0.46mm (18.11mil)	Round	47		685.09 mm (26.97
T2 Inch)	0.8mm (31.496mil)	Round	279		1360.19 mm (53.55

Totals Inch)			326		2045.28 mm (80.52

```
Total Processing Time (hh:mm:ss) : 00:00:04
```

Текстовый файл командной информации <Имя_проекта>.TXT в формате сверлильного станка с ЧПУ Excellon, являющемся фактически мировым стандартом в производстве печатных плат. Листинг 13.3 содержит фрагмент этого файла.

Листинг 13.3

```
M48
;Layer_Color=9474304
```

```
;FILE_FORMAT=4:2
```

```
METRIC,LZ
```

```
;TYPE=PLATED
```

```
T1F00S00C0.46
```

```
T2F00S00C0.80
```

```
%
```

```
T01
```

```
X008621Y003907
```

```
X009729Y00394
```

```
X010358Y00362
```

```
X011199Y003249
```

```
X011842Y00362
```

```
X015238Y003107
```

```
X01617Y005687
```

```
Y00375
```

```
X017Y006
```

```
Y00725
```

```
Y0075
```

```
X01775
```

```
Y00725
```

```
Y007
```

```
Y00675
```

```
.....
```

```
.....
```

```
X01425
```

```
X0135Y0085
```

```
Y00825
```

```
Y008
```

```
Y0085
```

```
X016Y009
```

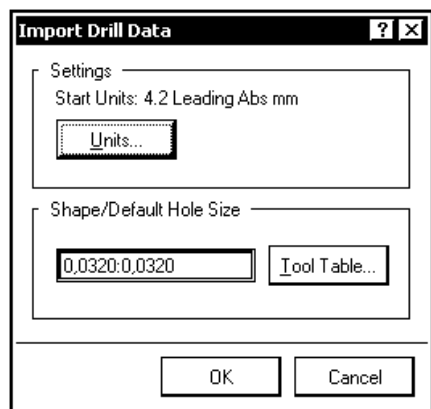
```
Y00975
```

```
M30
```

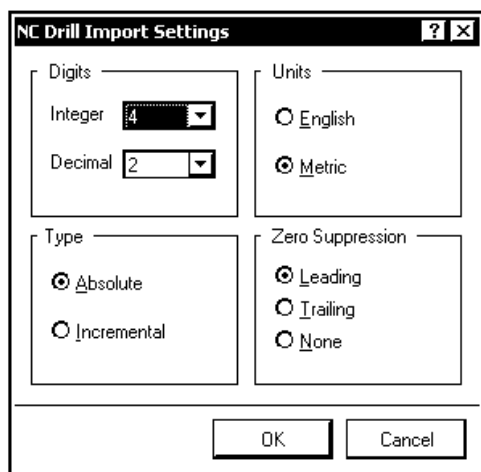
Двоичный файл координатной информации <Имя_проекта>.DRL. Данные из этого файла автоматически импортируются в среду САМ-приложения Altium CAMtastic, в которой выполняются визуализация карты сверления, контроль

и доработка данных перед окончательной передачей их производителю печатных плат.

CAMtastic — специализированная CAM-программа фирмы Innovative CAD Software (CAM — Computer Aided Manufacturing — компьютерная подготовка производства). Программа интегрирована в Altium Designer как динамически-загружаемое (DLL) приложение. Открывается диалоговое окно настройки формата импортируемых данных **Import Drill Data** (рис. 13.8, а), в котором следует щелчком на кнопке **Units** активизировать диалог **NC Drill Import Settings** (рис. 13.8, б) и подтвердить настройки, выполненные ранее, при экспорте информации для сверления из Altium Designer (см. рис. 13.7).



а



б

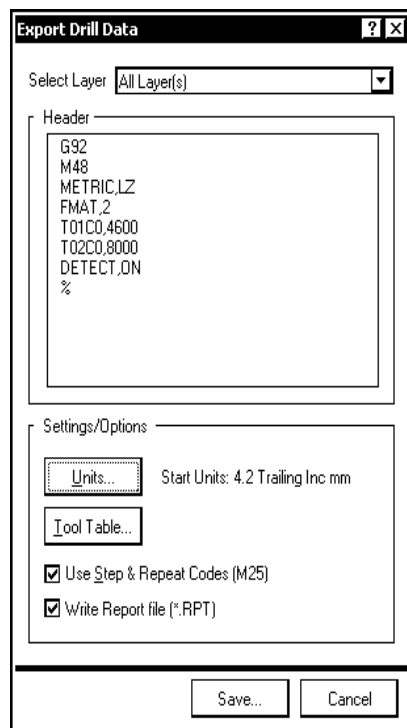
Рис. 13.8. Настройка параметров импорта данных для сверления в CAMtastic:

а — определение системы единиц измерения и таблицы инструментов;

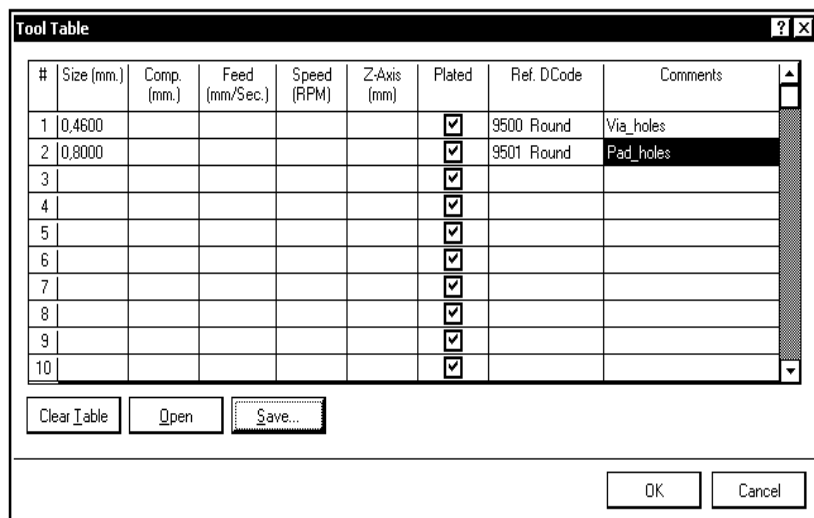
б — определение формата координатной информации

Для передачи данных CAMtastic на производственное оборудование командой главного меню программы **File | Export | Drill** (Файл | Экспорт | Сверление) следует запустить процедуру экспорта. В открывающемся диалоговом окне (рис. 13.9, а) предоставляется возможность настройки формата выходных данных.

- ❑ В поле **Header** приведен заголовок текстового файла управляющей информации для сверлильного станка ЧПУ.



a



б

Рис. 13.9. Настройка параметров экспорта данных сверления из CAMtastic в выходные файлы: а — просмотр заголовка выходного файла; б — редактирование таблицы инструментов

- ❑ Кнопкой **Units** можно повторно запустить процедуру настройки формата выходных данных — привести координатную информацию в соответствие с точностью позиционирования рабочего органа станка, имеющегося в распоряжении изготовителя. Дублирование операций настройки формата данных объясняется тем, что программа CAMtastic может использоваться автономно для проектирования деталей механизмов, в том числе и выполнения "механической" составляющей проекта функциональных узлов на печатных платах. В этом случае координатная информация порождается самой программой CAMtastic, и при ее экспорте необходима настройка формата данных.
- ❑ Кнопкой **Tool Table** можно вызвать и отредактировать таблицу инструментов (рис. 13.9, б). При импорте в CAMtastic из проекта печатной платы передаются значения диаметров отверстий КП и ПО. Нужно, однако, помнить, что в конструкторской документации указывается диаметр готового металлизированного отверстия, поскольку именно это значение должно быть достаточным для монтажа штыревых выводов компонентов и только его можно проконтролировать на готовой печатной плате. Поэтому диаметр отверстия, подлежащего металлизации, а значит и диаметр сверла должен превышать значение, импортированное из проекта печатной платы, на величину, равную двойной толщине слоя металлизации (устанавливается технологом и составляет обычно около 0,1 мм).

Кроме диаметра инструмента в таблице можно назначить скорость вращения сверла (**Speed**), подачу (**Feed**) и глубину сверления (**Z-axis**).

13.6. Экспорт в формате фотоплоттера Gerber

Система команд и формат представления координатной информации, разработанные в свое время для фотоплоттеров фирмы Gerber, в настоящее время стали фактически мировым стандартом в производстве фотшаблонов проводящего рисунка печатных плат. На практике используются две разновидности фотоплоттеров — с набором диафрагм, формирующих световое пятно (апертуру), производящее рисунок на фотопленке, и с программно синтезируемыми апертурами, форма и размеры которых импортируются из проекта печатной платы. Форматы данных для фотоплоттеров этих двух разновидностей называются, соответственно, RS274D и RS274X.

Управляющая информация для фотоплоттера состоит из команд выбора апертуры для выполнения очередного фрагмента печатного рисунка, перемещения рабочего органа фотоплоттера, включения и выключения засветки протяженных участков печатного проводника, включения вспышки света в местах расположения контактных площадок.

Основное различие массивов управляющей информации для фотоплоттеров с фиксированными и с синтезируемыми апертурами состоит в необходимости в первом случае кроме непосредственно файлов с координатной информацией формировать файл с таблицей апертур. При передаче файлов управляющей информации производителю данные из этой таблицы используются для выбора соответствующих диафрагм на имеющемся в его распоряжении фотопостроителе. Таблица апертур может генерироваться из проекта печатной платы, либо подбирается из имеющихся библиотек.

Для фотопостроителя с синтезируемыми апертурами при экспорте проекта печатной платы генерируются и включаются в файл выходных данных макрокоманды управления синтезом апертур. Формат RS274X представляет собой развитие формата RS274D: содержит команды заливки полигонов металлизации и работает с негативными изображениями слоев сплошной металлизации Plane-слоев печатной платы.

Полное описание форматов RS274D и RS274X приведено в [7]. Здесь ограничимся рассмотрением процедуры формирования управляющей информации в Gerber-формате.

1. Активизировать команду главного меню **File | Fabrication Outputs | Gerber Files** (см. рис. 13.3). Открывается диалоговое окно настройки экспорта **Gerber Setup** (рис. 13.10) с пятью панелями-вкладками.
2. На вкладке **General** установить метрическую систему единиц измерения (**Millimeters**) и выбрать формат представления чисел в выходной управляющей информации. Число десятичных разрядов до разделителя (запятой или десятичной точки) определяет максимально возможные линейные размеры обрабатываемого объекта, а число знаков после разделителя — необходимую точность позиционирования рабочего органа оборудования (фотоплоттера).
3. На вкладке **Layers** (Слои), устанавливая флажки, составить список слоев, для которых необходимо выполнить послойные фотошаблоны (рис. 13.11). В колонке **Mirror** можно указать слои, для которых нужны зеркально отображенные фотошаблоны.

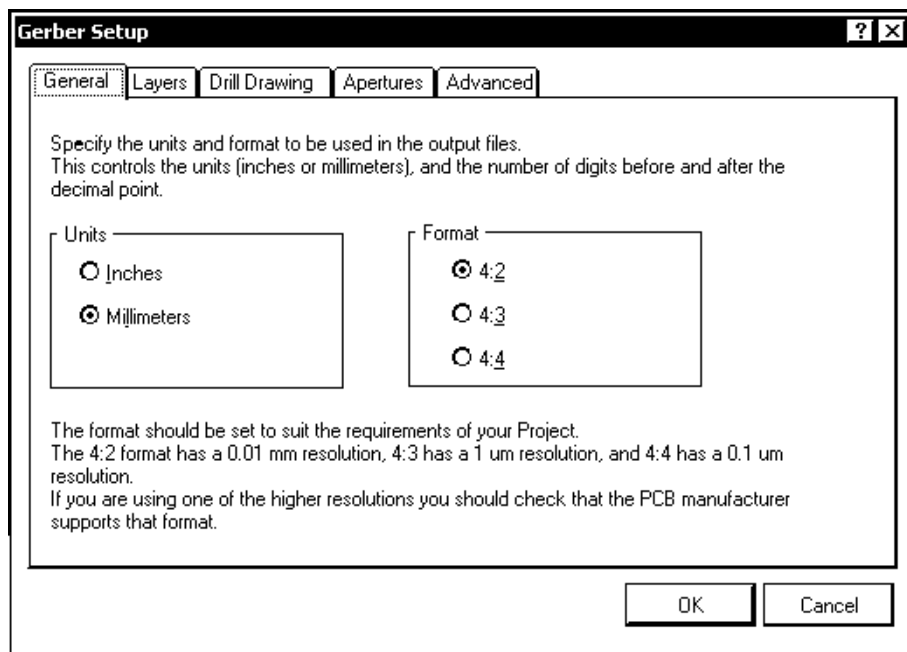


Рис. 13.10. Настройка параметров экспорта в формат фотоплоттера Gerber: панель формата координатных чисел

СОВЕТ

На практике встречается рекомендация разработчику печатной платы не формировать зеркально отображенные фотошаблоны, а предоставить производителю самому выполнить зеркальное отображение. Это облегчает контроль выполнения проекта в момент передачи его производителю.

Кроме того, в поле **Mechanical Layers to Add to All Plots** указать механические слои, которые должны быть изображены на всех фотошаблонах, например, контуры печатной платы, реперные знаки для совмещения слоев при сборке платы и т. п.

- На вкладке **Apertures** (рис. 13.12) активизировать (или нет) опцию **Embedded Apertures (RS274X)**. Если опция выбрана, это означает, что выходные данные будут формироваться в формате RS274X, т. е. команды синтеза апертур будут генерироваться и встраиваться в файл выходных данных. В противном случае с помощью кнопок управления, расположенных в правой половине диалогового окна, следует сформировать или загрузить извне готовую таблицу апертур. Для формирования таблицы апер-

тур из параметров печатного рисунка проектируемой платы следует воспользоваться кнопкой **Create List From PCB**.

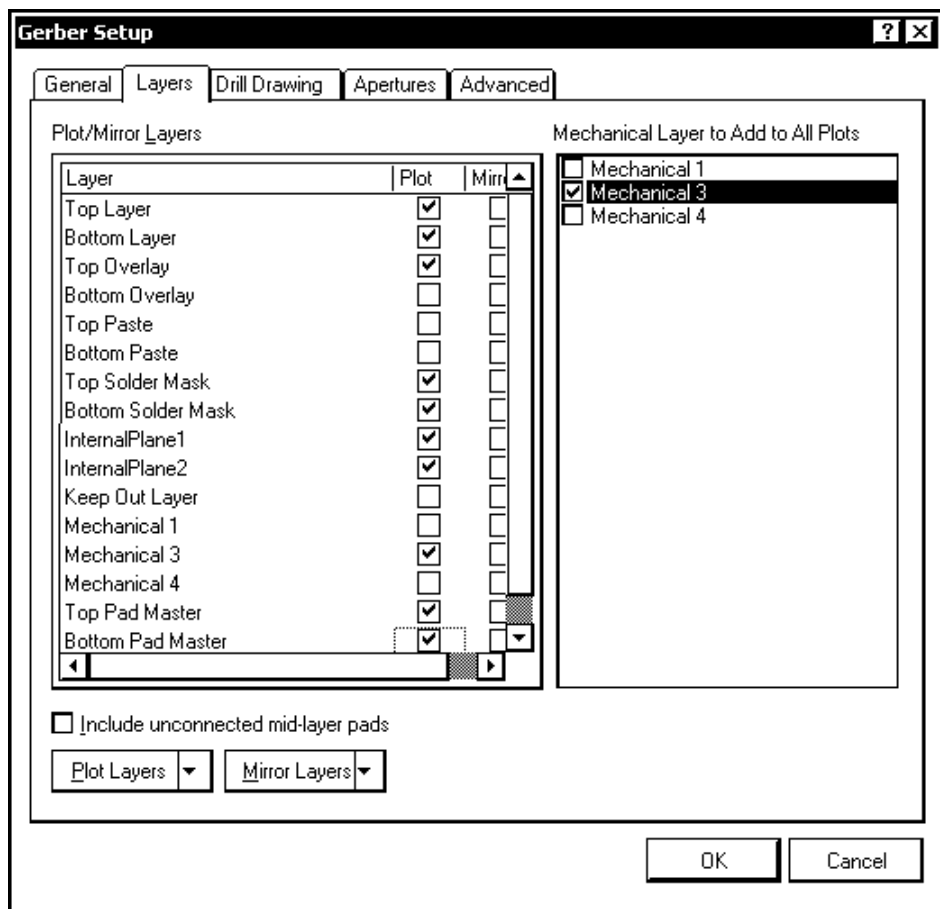


Рис. 13.11. Настройка параметров экспорта в формат фотоплоттера Gerber: панель выбора слоев для фотошаблона

- На вкладке **Drill Drawing** назначить (если нужно) формирование слоев с изображением отверстий (Drill Drawings) и обозначениями центров отверстий (Drill Guide).
- На вкладке **Advanced** установить в поле **Batch Mode** опцию **Separate file per layer** — генерировать отдельный файл данных для каждого слоя (фотошаблона).

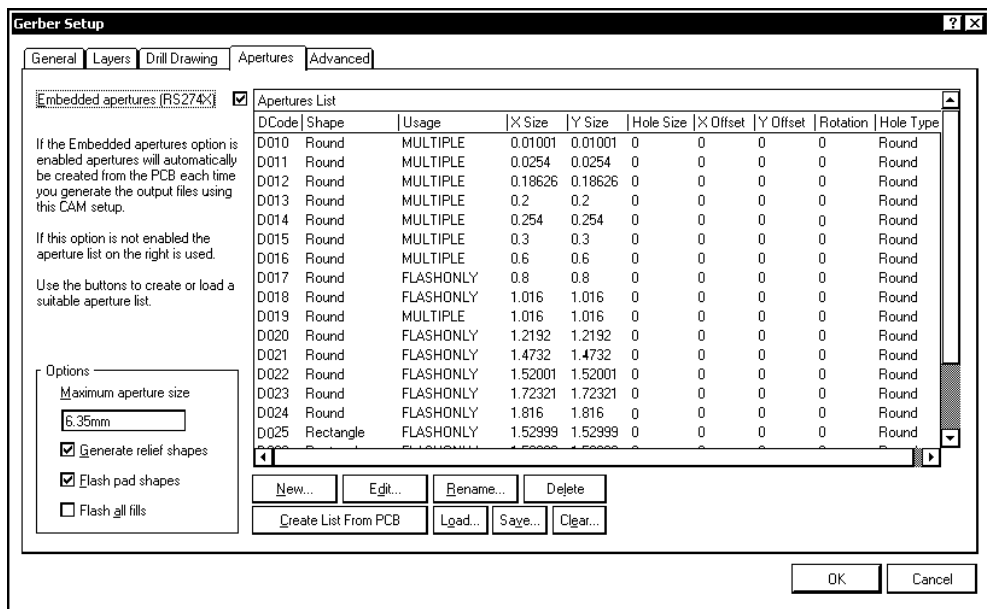


Рис. 13.12. Настройка параметров экспорта в формат фотоплоттера Gerber: панель редактирования апертур

- Щелчком кнопки **ОК** завершить настройки и запустить генерацию выходных данных.
- Программа генерирует несколько выходных файлов. Полный их список выводится в файл Status Report.txt (листинг 13.4).

Листинг 13.4

```
Output: Gerber Files
Type   : Gerber
From   : Project [Free Documents]

Generated File[Project.GTL]
Generated File[Project.GBL]
Generated File[Project.GTS]
Generated File[Project.GBS]
Generated File[Project.GP1]
Generated File[Project.GP2]
Generated File[Project.GM3]
```

```

Generated File[Project.GPT]
Generated File[Project.GPB]
Generated File[Project.GG1]
Generated File[Project.RUL]
Generated File[Project.EXTREP]
Generated File[Project.REP]

```

Files Generated : 13

Documents Printed : 0

Finished Output Generation At 21:44:01 On 07.01.2008

Видим, что имена всех файлов-фотошаблонов совпадают с именем проекта печатной платы, а в расширениях имен зашифровано имя слоя. Расшифровка содержится в файле *.EXTREP (от Extension Report — отчет о расширениях) (листинг 13.5).

Листинг 13.5

```

-----
Gerber File Extension Report For: Project.GBR    07.01.2008   21:44:00
-----
-----

```

Layer Extension	Layer Description

.GTL	Top Layer
.GBL	Bottom Layer
.GTS	Top Solder
.GBS	Bottom Solder
.GP1	InternalPlane1
.GP2	InternalPlane2
.GM3	Mechanical 3
.GPT	Top Pad Master
.GPB	Bottom Pad Master
.GG1	Drill Guide

.GTL	Top Layer
.GBL	Bottom Layer
.GTS	Top Solder
.GBS	Bottom Solder
.GP1	InternalPlane1
.GP2	InternalPlane2
.GM3	Mechanical 3
.GPT	Top Pad Master
.GPB	Bottom Pad Master
.GG1	Drill Guide

После этого автоматически открывается программа CAMtastic. Программа выполняет визуализацию сформированных фотошаблонов, а также данных сверления и фрезерной обработки контуров платы, и дает возможность их дополнительной коррекции. Не будем подробно описывать программу CAMtastic, а ограничимся функциями коррекции и экспорта данных.

В главном окне программы представлено графическое изображение всех фотошаблонов "на просвет" так же, как в графическом образе печатной платы.

В плавающей панели **CAMtastic** (рис. 13.13) выводится список слоев фотошаблонов с указанием цвета и активности каждого слоя, а также ряд функций управления. Установка или снятие флажков включает или отключает видимость слоев на экране.

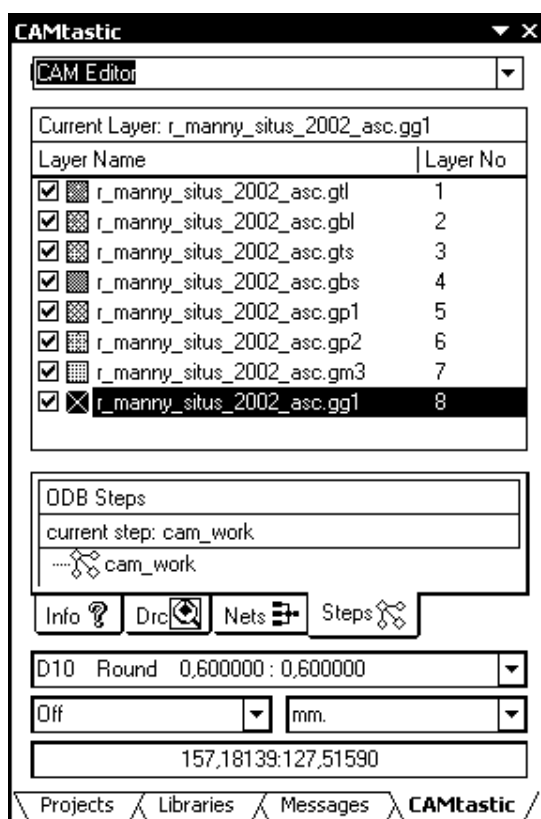


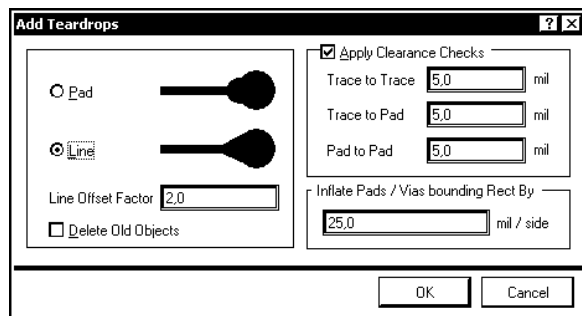
Рис. 13.13. Сводка информации о фотошаблонах в панели CAMtastic

В нижней части панели расположены поля с информацией о координатах курсора, выборе системы единиц измерения, назначения или отключения захвата курсором объектов на фотошаблоне.

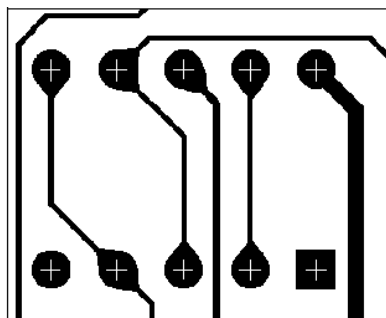
Из функций редактирования фотошаблонов рассмотрим одну — сглаживание переходов от контактной площадки к печатному проводнику — выполнение так называемых "слезок" (Teardrops).

Последовательность действий такова.

1. Активизировать команду **Tools | Teardrops**. Выделить рамкой селекции группу контактных площадок.
2. Щелчком правой кнопки мыши вызвать диалоговое окно настройки **Add Teardrops** (рис. 13.14, а).
3. Выбрать форму "слезки" **Line** (сглаживание линией) или **Pad** (размещение на стыке линии и КП кружка по типу контактной площадки) и завершить формирование щелчком на кнопке **ОК**. Контактные площадки вытягиваются в сторону проводника (рис. 13.14, б) и приобретают вид капли.



а



б

Рис. 13.14. Формирование "слезок" на переходе от КП к проводнику:
а — диалог настройки; б — результат формирования "слезок"

Чтобы окончательно сформировать набор фотошаблонов для производства, нужно выполнить операцию экспорта:

1. Командой главного меню **File | Export | Gerber** открыть диалоговое окно настройки параметров экспорта (рис. 13.15).
2. В поле **Format** левой кнопкой выбрать один из трех доступных форматов выходных данных: формат **RS-274X**, **RS-274** или **Fire9000** (при щелчках на кнопке название формата меняется циклически).

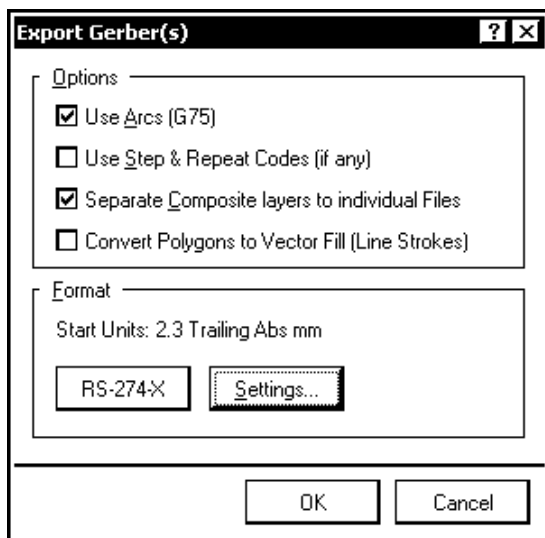


Рис. 13.15. Настройка параметров экспорта из CAMtastic в файлы управляющей информации для фотоплоттера

3. Кнопка **Settings** вызывает диалог настройки формата координатных чисел, аналогично рис. 13.8, б.

Щелчком на кнопке **OK** завершить настройку формата выходных данных. Открывается окно **Write Gerber(s)** (Записать Gerber-файлы), (рис. 13.16). В центральном поле окна располагается список всех Gerber-файлов, подготовленных к экспорту. Редактирование записей в этом списке невозможно.

- ☐ В колонке флажков слева от списка установить флажки для тех файлов, которые необходимо записать.
- ☐ В нижней части окна (рис. 13.16) расположено поле, в котором указывается путь для экспорта. Путь можно ввести непосредственно с клавиатуры либо выбрать в дереве файловой структуры компьютера, разворачиваемом по щелчку мышью на кнопке, расположенной справа.

Завершить экспорт щелчком мышью на кнопке **OK**. Программа записывает экспортные файлы по указанному адресу. Выходные данные представляют собой текстовые строки с командами управления и координатами элементов печатного проводящего рисунка, защитных масок, шелкографии, пастовых трафаретов и т. д.

Фрагмент выходного Gerber-файла приведен в листинге 13.6.

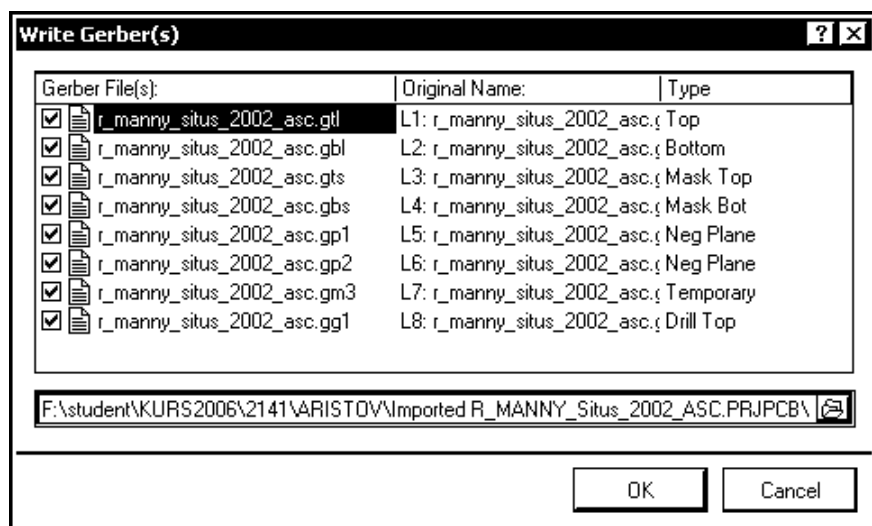


Рис. 13.16. Сводка фотошаблонов для вывода на фотоплоттер

Листинг 13.6

```
%FSTAX33Y33*%
%MOMM*%
%SFA1B1*%

%IPPOS*%
%ADD10C,0.599999*%
%ADD11C,0.250000*%
%ADD12C,0.299999*%
%ADD13C,0.030000*%
%ADD14C,1.519997*%
%ADD15R,1.529997X1.529997*%
%ADD16R,1.529997X1.529997*%
%ADD17C,1.019998*%
G54D10*
X054999Y052499D02*
X059999D01*
X069999Y084999D02*
X071389Y086389D01*
```



```

X102499Y069999D02*
X104999D01*
X107499Y089999D02*
X112499Y084999D01*
X147499Y052499D02*
X151469D01*
.....
.....

X159249Y066639D03*
X168179Y090099D03*
X114149Y045059D03*
X122149Y068299D03*
X152379Y031069D03*
X153059Y052709D03*
X144199Y046049D03*
G54D13*
X040537Y096962D03*
G54D11*
X043134Y097499D02*
X043125Y098125D01*
X042499Y098134*
X043134Y097499D02*
X043214Y098214D01*
X042499Y098134*
M02*

```

13.7. Экспорт в формат ODB++

При экспорте в формат ODB++ строится стандартная файловая структура, организованная в дерево каталогов и подкаталогов, в которых структурируется и сохраняется вся информация об элементах проекта печатной платы. Такую структуру часто называют базой данных проекта. Она позволяет переносить проект из одних САПР в другие без потери информации. Все файлы

структуры ODB++ представляют собой текстовые файлы в стандартной кодировке ASCII¹.

1. Командой меню **File | Fabrication Outputs | ODB++ Files** открыть диалог настройки **ODB++ Setup** (рис. 13.17).

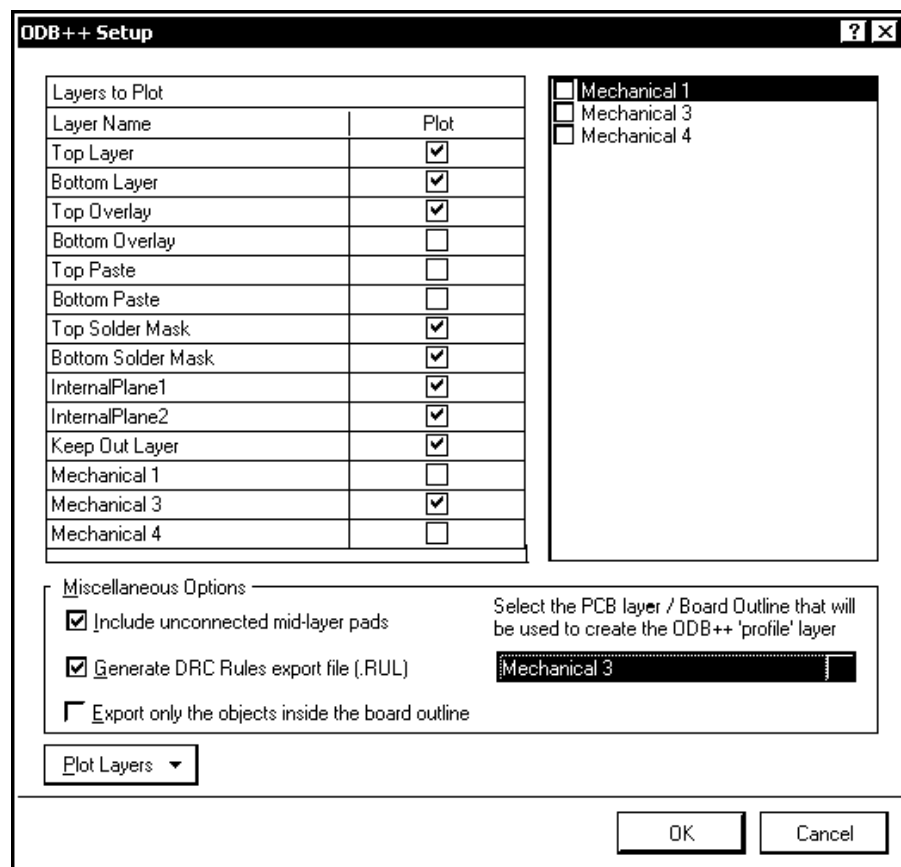


Рис. 13.17. Настройка экспорта в формат ODB++

2. В поле **Layers to Plot** представлен список всех слоев печатной платы текущего открытого проекта. Установкой флажков в колонке **Plot** (Чертить) составить список слоев, подлежащих экспорту. Вместо этого можно кноп-

¹ Документ ODB++ Version B.04. Doc. 0202.0801, Aug. 2001. Valor Computerized Systems, Ltd. Yavne, Israel. Интернет-ресурс <http://www.valor.com>.

кой **Plot Layers** активизировать варианты автоматического назначения слоев:

- **All On** — включить в список все слои;
- **All Off** — исключить все слои из списка;
- **Used On** — включить в список только используемые слои.

3. В поле **Miscellaneous Options** (Дополнительные настройки) установить опции:

- **Include unconnected mid-layer pads** — включить неподсоединенные КП на внутренних слоях;
- **Export DRC Rules export file (.RUL)** — экспортировать файл правил DRC-контроля (расширение имени .RUL);
- **Select the PCB layer / Board Outline that will be used to create the ODB++ 'profile' layer** — выберите слой платы или внешнего контура, который будет использован для создания слоя "профиль" в структуре данных ODB++.

4. Щелчком на кнопке **ОК** завершить настройку и запустить преобразование данных.

Данные в формате ODB++ структурированы в дерево каталогов: **Fonts**, **Matrix**, **Misc**, **Steps**, **Symbols** и **User** (рис. 13.18).

- ❑ Каталог **Fonts** содержит единственный файл Standard с текстовым описанием графики символов шрифта надписей в проекте.
- ❑ Каталог **Matrix** содержит определение физического порядка слоев, их типов, характера межслойных переходов — сквозные (Thru), слепые (Blind), захороненные (Buried).
- ❑ Каталог **Steps** образует дерево подкаталогов, из которых отметим PCB, с подкаталогом Layers, который, в свою очередь, разворачивается в подкаталоги для каждого слоя экспортируемого проекта. Кроме слоев, определенных в проекте печатной платы, в Altium Designer, в ODB-структуре выделяются в отдельные слои данные о расположении компонентов на верхней и нижней сторонах платы (слои **comp_+_top** и **comp_+_bot**) и данные для сверления (слой **drill**).
- ❑ В подкаталог Netlists\Cadnet экспортируется текстовое описание электрических цепей проекта (Netlist).

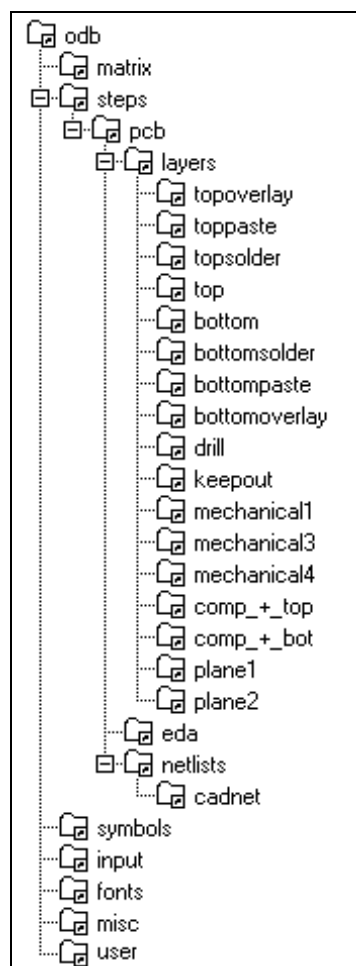


Рис. 13.18. Дерево структуры документов в формате ODB++

- ❑ В каталог **User** экспортируется файл правил DRC-контроля, если его экспорт был задан в диалоге настройки (см. рис. 13.17).

Сгенерированные выходные данные включаются в логическую структуру текущего открытого проекта и отображаются в плавающей панели **Projects** в ветви **Generated (ODB Output)** дерева текущего проекта (рис. 13.19).

Физический путь в файловой структуре компьютера, по которому размещаются данные экспорта, назначают на вкладке **Options** диалога настройки **Options for Project** (см. рис. 13.5). По умолчанию это путь **Projects\Project Out-**

puts for <Имя_проекта>\ODB. При необходимости можно назначить любой другой путь.

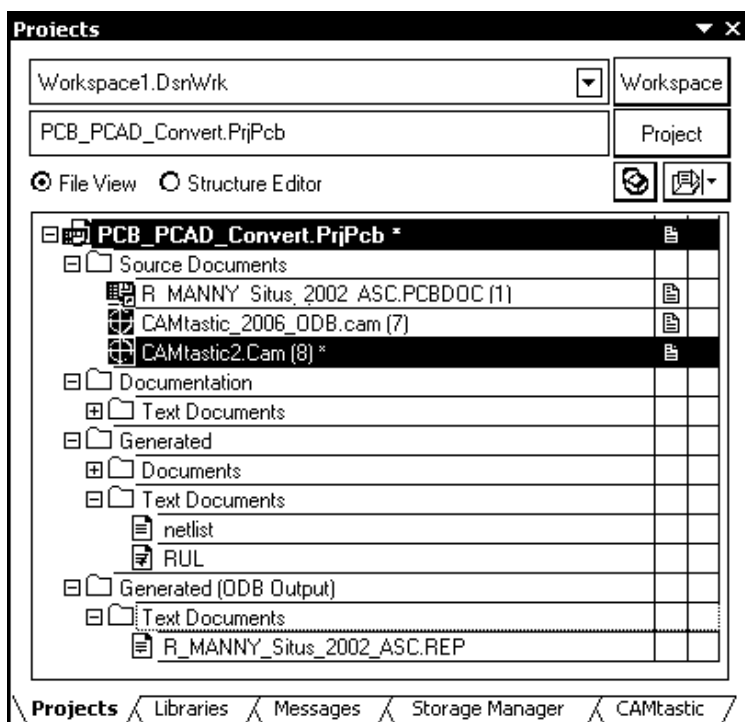
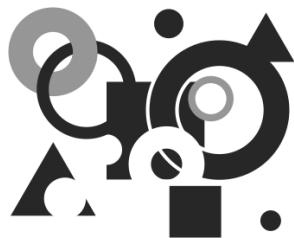


Рис. 13.19. Сводка данных экспорта в панели **Projects**



Разработка графических конструкторских документов средствами AutoCAD

AutoCAD — мощное графическое средство, предназначенное для формирования и выпуска машиностроительных чертежей. Для того чтобы читатель мог выполнить необходимую доработку импортированного образа проекта до состояния полноценных конструкторских документов по ЕСКД, необходимо изучить основные функции AutoCAD. Привести описание его функций в полном объеме из-за ограниченного объема книги невозможно. В последнее время издано несколько исчерпывающих руководств и самоучителей, позволяющих изучить функции и приемы работы в среде AutoCAD самых новейших версий (AutoCAD 2000–2008) [8–10], поэтому в этой главе ограничимся рассмотрением только того минимума функций, который позволяет решить нашу узкую задачу: довести импортированные из Altium Designer DWG/DXF-данные до полноценного конструкторского документа по ЕСКД. Рассмотрим только функции AutoCAD 2002, поскольку объем функций, который необходим для решения задачи формирования конструкторских документов на функциональный узел на печатной плате, в более поздних версиях AutoCAD остался без изменений.

14.1. Общая характеристика графических средств AutoCAD 2002

Окно графического редактора AutoCAD 2002 имеет традиционную для приложений MS Windows форму (рис. 14.1).

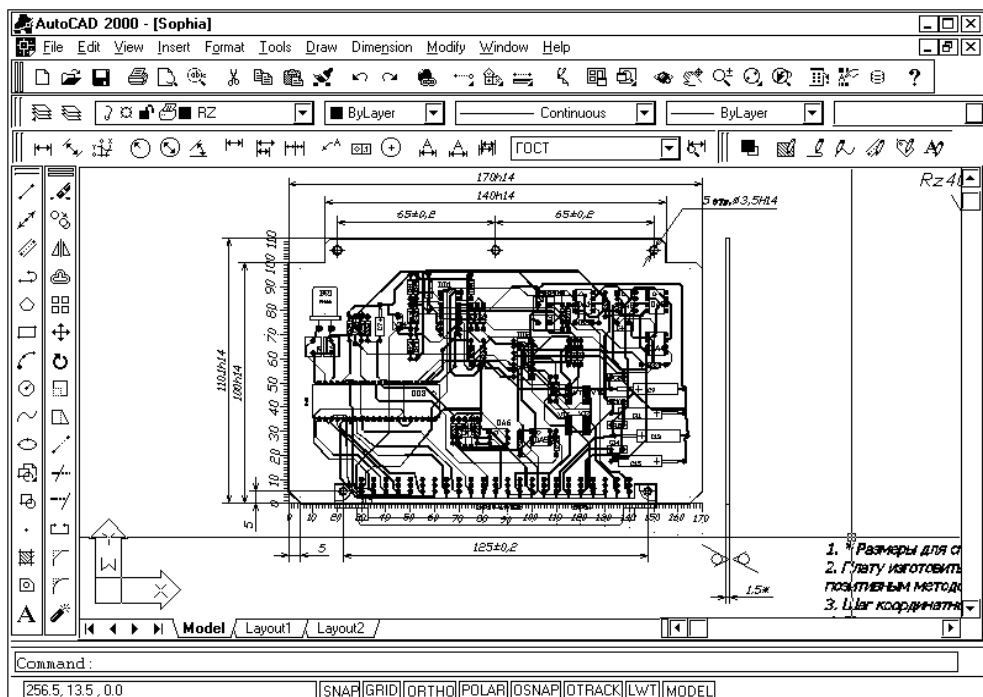


Рис. 14.1. Главное окно AutoCAD

В верхней части главного окна программы располагаются:

- заголовок программы с именем текущего файла проекта;
- главное "выпадающее" (Dropdown) меню программы с командами операционной оболочки (**File**, **Edit** и др.) и командами настройки AutoCAD и управления процессом выполнения проекта;
- "стандартная" линейка инструментов (Standard Toolbar), в которой расположены экранные кнопки, дублирующие ряд наиболее употребительных команд работы с файлами (открытие, сохранение, копирование и вставка копий из буфера обмена) и некоторых часто используемых команд AutoCAD;
- линейка управления слоями графического редактора, цветом, типом и толщиной линий.

Центральную часть экрана занимает графическое окно, обрамленное так называемыми "плавающими" меню, на кнопки-пиктограммы которых выведены все основные команды AutoCAD. Плавающими эти меню называются по-

тому, что их можно в нужный момент вызывать на экран или убрать с экрана, перемещать по экрану в пределах графического окна или фиксировать на его краях. Всего доступно 24 плавающих меню. Команды в них сгруппированы по принципу однородности выполняемых операций (черчение различных фигур, их редактирование, нанесение размеров и т. п.).

В нижней части экрана располагаются текстовое окно команд и линейка статуса, в которую выводятся координаты курсора и ряд "горячих" кнопок-команд, позволяющих изменять значения системных переменных, определяющих ход выполнения чертежа, не прерывая исполнения текущей команды: переключать с непрерывного на дискретное перемещение курсора мыши (**SNAP ON/OFF**) и менять величину шага, включать/отключать ортогональное перемещение курсора при черчении (**ORTHO ON/OFF**), включать/отключать привязку одних графических объектов к другим (**OSNAP ON/OFF**) и др.

Текстовое окно команд служит для ввода команд с клавиатуры. В процессе выполнения команд в него выводится и протоколируется командный диалог всего сеанса работы.

Такое разнообразие представления команд обеспечивает большую гибкость при работе: начать выполнение команды можно, например, из выпадающего меню или кнопкой-пиктограммой, затем ввести необходимые данные через текстовое окно с клавиатуры, после чего снова работать мышью и т. д.

Графические и другие визуально отображаемые данные в AutoCAD структурированы в слои, аналогично "электронным" САПР P-CAD, Altium Designer и др. Вместе с импортом образа печатной платы AutoCAD воспринимает структуру слоев исходного проекта.

Программу запускают так же, как и все приложения Windows: из стартового меню или указанием на значок-ссылку на рабочем столе компьютера. После запуска следует выполнить базовую настройку программы — назначить систему единиц измерения (команда **Format | Units**), размеры рабочего поля (команда **Format | Drawing Limits**) и стиль исполнения текстовых надписей (команда **Format | Text Style**). Настройки выполняются в диалоговых окнах и не представляют большого труда для работника, знакомого с приложениями Windows.

Поскольку в процессе работы над импортированным из Altium Designer интегральным образом печатной платы практически не приходится заниматься черчением (вся графика поступает готовой с DXF/DWG-файлом проекта), ограничимся рассмотрением функций AutoCAD, необходимых для выполнения рабочих чертежей печатной платы и сборки модуля 1-го уровня (ячейки)

на основе импортированного из Altium Designer интегрального образа трассированной печатной платы:

- ☐ нанесение размеров на чертеже;
- ☐ управление видами проекта в плавающих видовых окнах;
- ☐ присоединение к чертежам "форматов" и заполнение основной надписи чертежа.

Последние два вопроса наиболее скупо освещены в упомянутых изданиях.

14.2. Нанесение размеров и предельных отклонений на чертеже

Перед выполнением данной операции следует настроить стиль простановки размеров. Для этого командой главного меню **Format | Dimension Style** открыть диалоговое окно подсистемы Dimension Style Manager — менеджера размерных стилей (рис. 14.2).

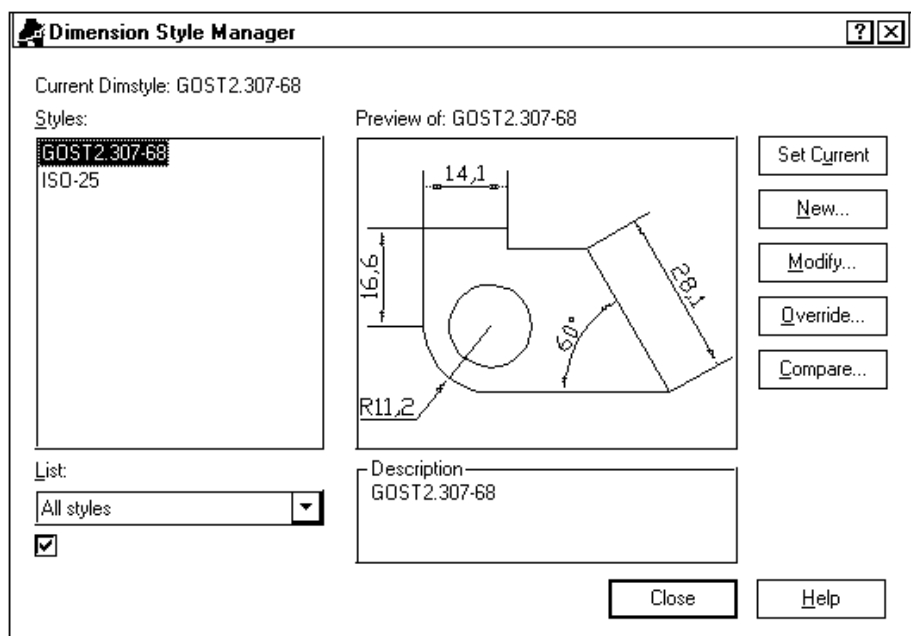


Рис. 14.2. Настройка стилей выполнения размеров

В поле **Styles** представлен список ранее установленных размерных стилей. Если черчение начинается с "чистого" листа, в нем указан единственный стиль **ISO-25**, назначенный по умолчанию.

В поле **Preview** приводится пример простановки размеров.

Чтобы назначить новый стиль, нужно нажать кнопку **New**. Открывается диалоговое окно назначения нового стиля (рис. 14.3).

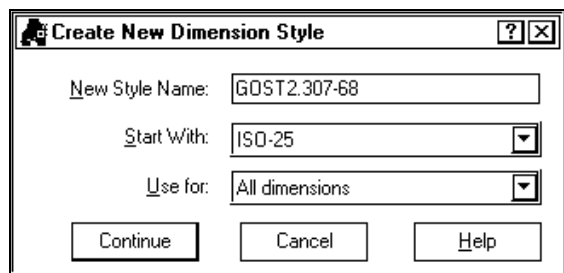


Рис. 14.3. Окно формирования нового размерного стиля

В поле **New Style Name** следует назвать имя нового стиля, например, для размеров по ЕСКД указать имя "GOST2.307-68".

В поле **Use for** выбирают вид размеров, для которых настраивается стиль (линейные, радиальные, угловые и т. д.) или стиль назначают для всех видов размеров (**All dimensions**).

Кнопкой **Continue** активизировать диалог настройки параметров нового стиля. Открывается диалоговое окно (рис. 14.4), содержащее 6 панелей-вкладок, на каждой из которых устанавливаются элементы стиля:

- ❑ **Lines and Arrows** (Линии и стрелки) — способ изображения размерных и выносных линий (поля **Dimension Lines** и **Extension Lines**): цвет (**Color**), толщина линии (**Lineweight**), форма и размеры стрелок (**Arrowheads**), возможность не изображать (**Suppress**) одну или обе выносные линии, один или оба конца размерной линии, способ обозначения и размер перекрестия в центрах окружностей и дуг и т. д.;
- ❑ **Text** — размер шрифта и способ размещения размерного текста относительно выносных и размерных линий;
- ❑ **Fit** — расположение размерного текста, если он не умещается между выносными линиями;

- ❑ **Primary Units** — основные единицы измерения, включая точность обозначения размеров (в нашем случае достаточно одного знака после запятой);
- ❑ **Alternate Units** — альтернативные единицы (дюймы, если для основных единиц принята метрическая система);
- ❑ **Tolerances** — способ обозначения предельных отклонений размеров.

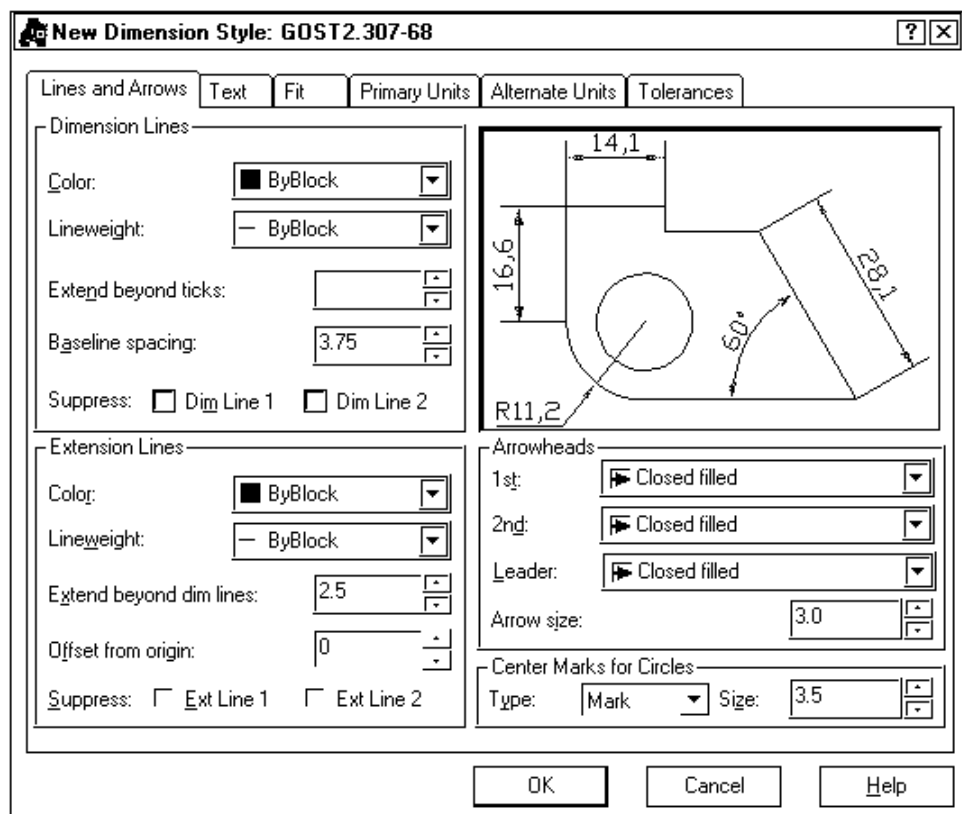


Рис. 14.4. Настройка параметров размерного стиля

Выполнив необходимые настройки, щелчком на кнопке **ОК** закрываем окно, показанное на рис. 14.4, возвращаемся в исходное окно (см рис. 14.2), и кнопкой **Set Current** присваиваем новому стилю статус текущего. Программа возвращается в главное графическое окно с изображением импортированного образа печатной платы (см. рис. 14.1).

При подготовке рабочих чертежей печатной платы и сборки функционально-го узла для нанесения размеров нужно создать два слоя AutoCAD, чтобы на будущем чертеже печатной платы были видны исполнительные и справочные размеры, необходимые для изготовления печатной платы, а на сборочном чертеже узла — габаритные, установочные и присоединительные размеры, относящиеся к сборке.

Перед нанесением размеров необходимо назначить один из этих слоев текущим и активизировать одну из команд-кнопок в линейке плавающего меню **Dimension** либо команду главного меню **Dimension** с соответствующей подкомандой (**Linear**, **Radial**, **Angular** (Угловые размеры) и т. д.).

При нанесении линейного размера последовательно указать курсором (со щелчком левой кнопкой мыши) две точки контура, после чего движением курсора в сторону свободного поля чертежа вынести обозначение размера и зафиксировать его щелчком левой кнопки мыши.

По ходу нанесения размера может возникнуть необходимость изменить форму изображения размера или текст размерной надписи. Так, при обозначении метрической резьбы размерная надпись должна начинаться с буквы "М", а после цифрового обозначения размера может быть указано значение шага резьбы (например, М12×1,25), при обозначении предельных отклонений линейных размеров размерные цифры сопровождаются условным обозначением полей допусков (кавалитетом) или числовыми значениями и т. п. Все эти варианты трудно предусмотреть при настройке размерного стиля, поэтому в таких случаях следует пользоваться опциями, предлагаемыми программой в текстовом командном окне. Возможны два способа редактирования.

1. Непосредственное редактирование надписи в текстовом окне команд — активизируется клавишей <Т> (**Text**). В этом случае введенный с клавиатуры текст полностью заменяет предлагаемую программой размерную надпись.
2. Редактирование с помощью текстового редактора AutoCAD — активизируется клавишей <М>. В этом случае открывается диалоговое окно текстового редактора (**Mtext**) (рис. 14.5).

В главном поле окна угловыми скобками условно обозначен текст размерной надписи, предлагаемый программой, в соответствии с настройками текущего размерного стиля.

Обычными способами редактирования текста слева и справа от этого условного обозначения можно внести необходимые добавления (см. рис. 14.5). Для полной замены размерной надписи следует удалить угловые скобки и напи-

сать на их месте необходимый текст. Нажатие кнопки **ОК** завершает редактирование, и у размерной линии на чертеже возникает новая надпись.

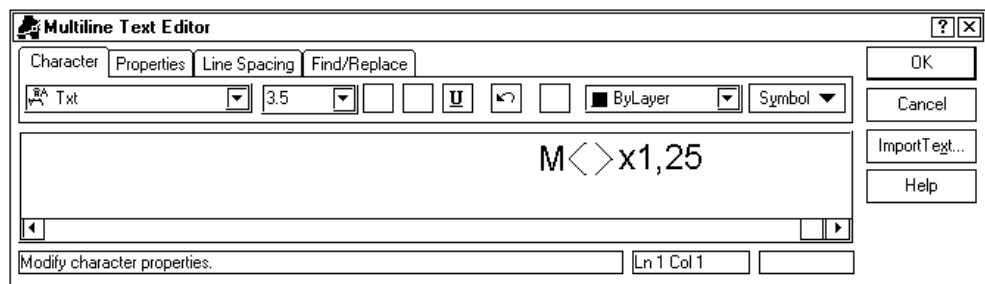


Рис. 14.5. Окно встроенного текстового редактора AutoCAD

14.3. Создание и управление свойствами плавающих видовых окон в AutoCAD 2002

В AutoCAD предусмотрено два способа изображения объектов — изображение в так называемом пространстве модели (Model Space) и в пространстве листа (Paper Space или Layout). В пространстве модели выполняется вся проектная работа. При выполнении планарного чертежа ортогональные проекции объемных конструкций изображаются в единственном видовом окне. При таком способе создания конструкторского документа для его распечатки не придется переходить из пространства модели в пространство листа.

При объемном конструировании различные виды объемной конструкции могут представляться в нескольких видовых окнах. Эти окна могут быть фиксированными, накрывающими весь экран наподобие кафельной плитки (Tiled Viewports), или плавающими (Floating Viewports), произвольного размера, возможно, накрывающими друг друга.

Из пространства модели можно распечатать вид, находящийся в единственном, активном окне. Переход в пространство листа позволяет окончательно подготовить проект к получению "твердой копии" — выводу на бумагу совмещенного изображения из нескольких плавающих окон. Изображения, созданные в пространстве модели, при переходе в пространство листа становятся недоступными для редактирования. Изображение становится единым объектом вместе со своим видовым окном. Вместе с окном его можно пере-

мещать по экрану командой **Move**, копировать (**Copy**), масштабировать (**Stretch**), удалять (**Erase**) и т. д. Можно создать неограниченное число плавающих видовых окон (Floating Viewports), в которых размещаются именованные виды элементов проекта, но одновременно видимыми на экране (как и на чертеже) в AutoCAD 2002 могут быть не более 64 окон. Границы плавающих видовых окон, в отличие от границ фиксированных окон, можно сделать невидимыми, перемещения курсора происходят по всему экрану, и весь этот вид можно отобразить на бумаге в виде единого чертежа.

В AutoCAD 2000 и более поздних версий в одном проекте одновременно можно назначить несколько конфигураций в пространстве листа. По-английски они называются **Layouts**, что ближе всего по смыслу термину "раскладка". В нижней части главного окна программы помещены "вкладки" (Tabs), позволяющие в любой момент работы обращаться то к пространству модели, то к одной из раскладок пространства листа (см. рис. 14.1). При указании на вкладку курсором, со щелчком левой кнопкой мыши, соответствующий вид выдвигается на первый план на экране. В текстовом окне команд это переключение сопровождается фраза **<Switching to Layout>** или **<Switching to Model>**.

Функции установки конфигурации видовых окон сосредоточены в выпадающем меню **View | Viewports** (Вид | Видовые окна). Конфигурацию устанавливают по контексту: если мы работаем в пространстве модели, можно выбрать фиксированные видовые окна, а в пространстве листа — плавающие окна.

14.3.1. Создание новых плавающих окон

При разработке нового проекта автоматически устанавливается пространство модели, а "в запасе" обозначена вкладка первого, неконфигурированного пространства листа с именем **Layout1**.

При необходимости перехода в пространство листа следует указать курсором мыши (со щелчком левой кнопкой мыши) на вкладку **Layout1**. Открывается окно настройки конфигурации пространства листа **Page Setup** (рис. 14.6).

Центральная зона окна содержит два поля-вкладки для установки конфигурации: **Plot Device** (плоттера) и **Layout Settings** (листа). Поскольку пространство листа используется для подготовки документа к печати, функции конфигурации практически совпадают с функциями настройки печати, за исключением предварительного просмотра распечатки.

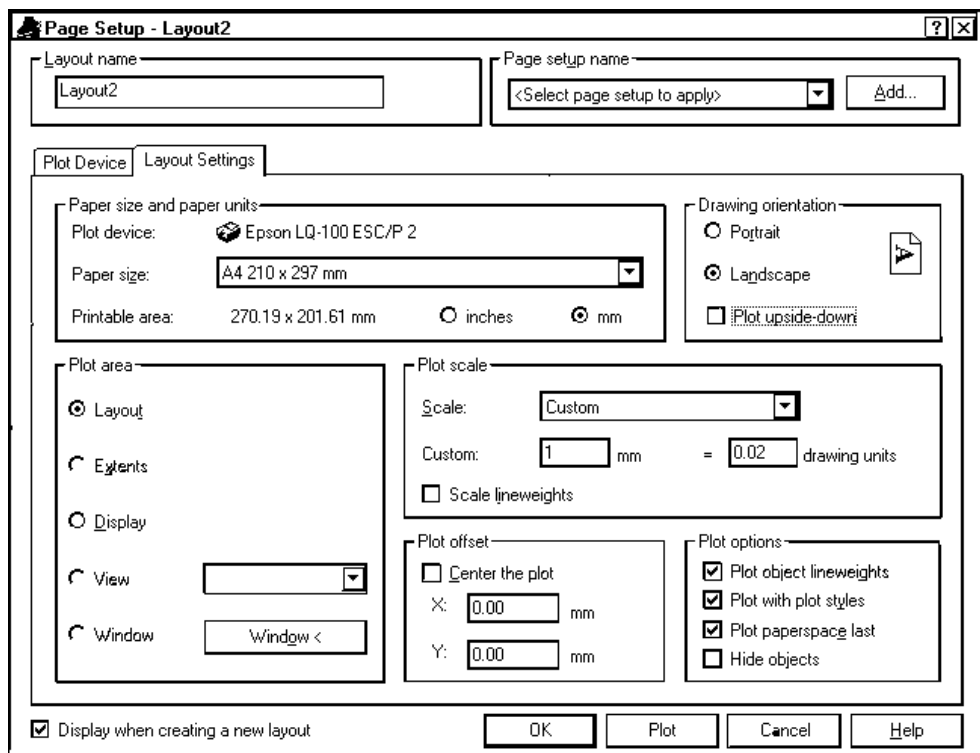


Рис. 14.6. Окно настройки формата пространства листа

На вкладке **Plot Device** можно указать устройство печати, на которое предполагается вывод чертежей. Это может быть принтер, установленный в конфигурации компьютера, и тогда придется выводить чертеж большого формата по частям либо один из широкоформатных плоттеров, с которыми по умолчанию в состоянии работать AutoCAD. Несмотря на то, что выбранный плоттер может отсутствовать в конфигурации компьютера, программа позволяет выполнить настройку чертежа в пространстве листа для этого плоттера, и только при выводе чертежа использовать то реальное устройство, которым располагает пользователь.

В поле **Paper size and paper units** указывают систему единиц измерения и размеры листа. Последние выбирают из списка в поле **Paper size** (Размер бумаги), но не более максимально допустимых "физических" размеров листа для назначенного плоттера или принтера. Для выпуска чертежей по ЕСКД нужно указать один из форматов ISO A0...A4, совпадающих по размерам листа с форматами ЕСКД по ГОСТ 2.301-68.

В версиях AutoCAD выше 2004-й настройка листа **Layout** модернизирована: выполняется в среде менеджера настройки, вызываемого из контекстного меню щелчком правой кнопки мыши на вкладке **Layout**.

По умолчанию в новом пространстве листа установлено единственное видовое окно, укладываемое в размеры назначенного листа, куда автоматически передается вид импортированного образа печатной платы из активного видового окна пространства модели.

Для назначения новых плавающих видовых окон указать в главном меню **View** подкоманду **Viewports**. Из него раскрывается выпадающее подменю, содержащее следующие подкоманды настройки конфигурации видовых окон:

- ☐ **Named Viewports** — именованные видовые окна;
- ☐ **New Viewports** — новые видовые окна;
- ☐ **1 viewport** — одно окно;
- ☐ **2 viewports** — два окна;
- ☐ **3 viewports** — три окна;
- ☐ **4 viewports** — четыре окна;
- ☐ **Polygonal viewports** — полигональные видовые окна;
- ☐ **Object** — объект в качестве границ видового окна.

Указать подкоманду **New Viewports**. Раскрывается окно установки конфигурации новых видовых окон (рис. 14.7).

В поле **New name** можно задать имя нового плавающего окна и при дальнейшем использовании в других чертежах вызывать его из списка установленных конфигураций, указывая подкоманду **Named Viewports**.

В поле предварительного просмотра **Preview** обозначаются контуры конфигурации нужных окон.

В поле **Standard viewports** курсором выбрать одну из строк списка возможных конфигураций. В наших целях следует активизировать опцию **Single** — единственное окно.

В поле **Setup** можно указать, работаем ли мы с двумерным (**2D**) или трехмерным объектом (**3D**). При работе с двумерным объектом во все назначаемые окна передается вид из текущего активного видового окна пространства модели (Current active viewport). Для работы с изображениями трехмерных объектов нужно открыть поле **Change View to** и в разворачивающемся списке

указать одну из шести стандартных ортогональных проекций: **Top**, **Bottom**, **Front**, **Back**, **Left**, **Right** (виды сверху, снизу, спереди, сзади, слева и справа) или четырех изометрических проекций объекта: **SW Isometric**, **SE Isometric**, **NE Isometric**, **NW Isometric** (вид с четырех сторон света: с юго-запада, юго-востока, северо-востока и северо-запада).

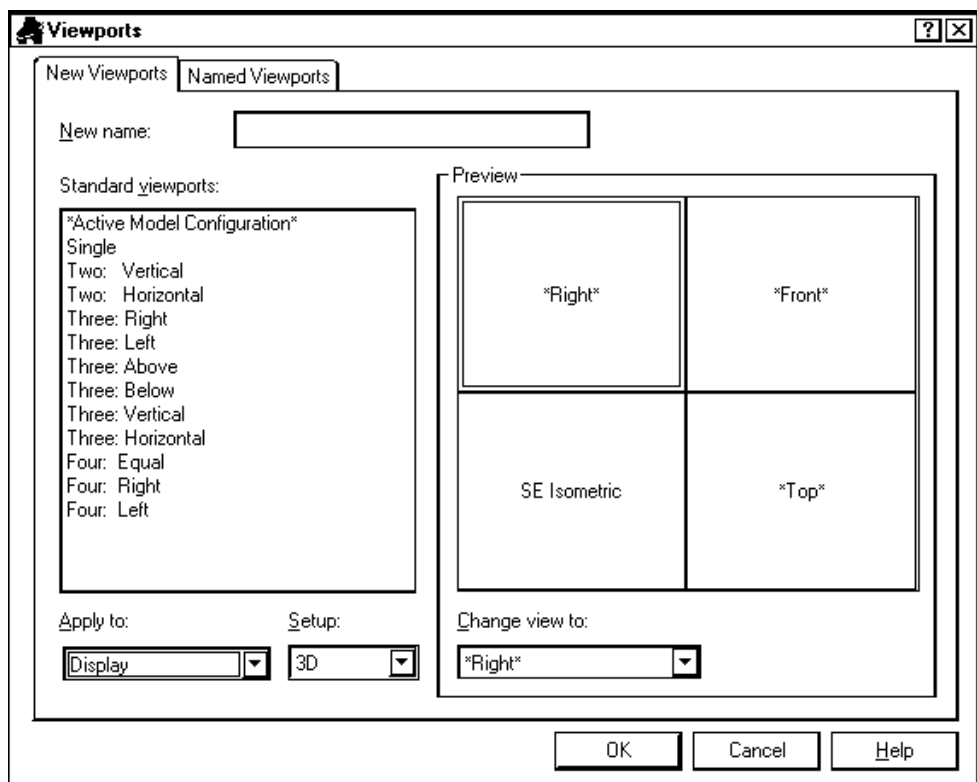


Рис. 14.7. Настройка плавающих видовых окон пространства листа

Так, в частности, при работе с импортированным из Altium Designer образом печатной платы, хотя он и планарный, главный вид печатной платы в трехмерном ее представлении является видом сверху, а для изображения вида со стороны пайки следует указать в поле **Setup** для нового плавающего окна пространства листа опцию **3D** и назначить вид снизу — **Bottom**.

Настройка конфигурации завершается щелчком на кнопке **OK**, после чего AutoCAD возвращается в главное окно, а в окне команд выводятся два предложения: **Specify first corner or [Fit]** (обозначить первый угол нового окна на

экране или автоматически вписать его в имеющийся экран) и **Specify opposite corner** (Обозначить противоположащий угол). В новом видовом окне строится зеркальное отображение полного образа трассированной печатной платы. На обоих видах пока еще сохраняется видимость объектов на всех слоях проекта "на просвет", как в исходном образе. Масштаб изображения в новом плавающем окне программа назначает так, чтобы в пределах окна размещались все объекты, сформированные в пространстве модели.

14.3.2. Масштабирование изображений в плавающих окнах

Размеры изображения объектов в пространстве листа масштабируются в так называемые единицы чертежа, **Drawing units**, отличающиеся от единиц измерения в пространстве модели. Масштаб изображения устанавливается при настройке конфигурации пространства листа (см. рис. 14.6).

При создании нового плавающего видового окна по умолчанию программа масштабирует изображение из пространства модели так, чтобы оно полностью поместилось в размеры назначенного листа распечатки. Изменение границ плавающего окна не меняет масштаба изображения объектов, расположенных в нем. Для изменения масштаба изображения в плавающем окне следует выбрать окно курсором, вызвать щелчком правой кнопки мыши "всплывающее" контекстное меню и активизировать в нем функцию управления свойствами объектов **Properties**.

Открывается окно менеджера свойств объектов (рис. 14.8). Свойства объекта могут быть перечислены по алфавиту или структурированы по категориям. Набор свойств, доступных для редактирования, зависит от вида выбранного объекта. Всего доступны 35 контекстов. Для изменения масштаба нужно выбрать в группе функций **Misc** строку **Standard scale** (при этом разворачивается таблица доступных масштабов) и указать необходимый стандартный масштаб из списка, либо назначить масштаб в строке пользовательских масштабов **Custom scale**.

В интересующем нас случае как в исходном окне пространства листа (с главным видом печатной платы), так и в новом окне (с видом снизу) устанавливаем единый стандартный масштаб. После этого надлежит так расположить изображения обоих видовых окон, чтобы установилась проекционная связь видов платы. Для этого следует активизировать команду перемещения **Move**, выбрать границу плавающего видового окна, а в качестве исходной точки перемещения указать какую-либо характерную точку на виде печатной платы

(один из углов контура платы, верхний или нижний край и т. п.) и, перемещая изображение, совместить линию курсора с некоторой характерной точкой или линией контура на виде платы в другом, неподвижном окне.

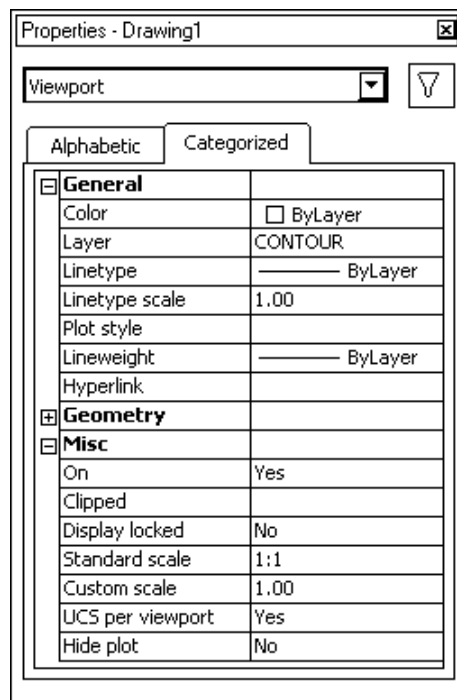


Рис. 14.8. Окно настройки свойств объектов AutoCAD

14.3.3. Управление видимостью объектов в плавающих окнах пространства листа

При исполнении конструкторских документов приходится управлять видимостью различных частей изображения в разных видах, скрывать одни его части, накрываемые в сборке другими частями и т. п.

В AutoCAD заложены обширные возможности управления видимостью. Нас интересуют средства управления видимостью в плавающих окнах пространства листа:

- ☐ затенение изображения (Screening) в выбранных видовых окнах;

- ☐ исключение скрытых линий контуров при распечатке чертежа;
- ☐ отключение видимости плавающих окон;
- ☐ управление видимостью слоев изображения в выбранных плавающих окнах;
- ☐ клиппирование (вырезание части) плавающих окон.

Применительно к выпуску конструкторской документации на функциональные узлы РЭС на печатных платах нас интересуют только две из перечисленных функций:

- ☐ для отключения видимости целого плавающего окна аналогично предыдущему случаю в группе **Misc** выбрать функцию **ON** и установить для нее значение **No** (нет). Значение **Yes** (да) делает окно видимым. При отключении видимости плавающего окна его границы остаются видимыми;
- ☐ управление видимостью слоев изображения в выбранных плавающих окнах применяется тогда, когда, например, нужно привести на чертеже изображение двух видов такого объекта, как печатная плата.

Сначала в пространстве листа выбрать плавающее окно, в котором требуется отключить видимость каких-либо объектов, и "активизировать" его двойным щелчком левой кнопкой мыши. Управляющей в данном случае является команда **MSPACE** (дублируется в текстовом окне команд).

"Активность" выбранного окна проявляется как переход его в пространство модели. Это особое состояние, в котором в активном окне возможно редактирование графических объектов, причем внесенные изменения передаются в пространство модели всего проекта и отображаются во всех плавающих окнах пространства листа, а видимость некоторых слоев в данном окне можно отключить (слои замораживаются), и элементы изображения, размещенные на этих слоях, остаются невидимыми только в этом видовом окне после того, как оно будет возвращено в прежнее, "пассивное" состояние (команда **PSPACE** или двойной щелчок левой кнопкой мыши за пределами активного окна).

Затем командой главного меню **Format | Layer** или щелчком мыши на пиктограмме в линейке статуса главного окна программы открыть окно управления слоями графического редактора **Layer Properties Manager** (рис. 14.9).

Далее в поле списка слоев нужно выбрать необходимые слои и для каждого из них указать в колонке **Active** опцию **Active VP Freeze** (в более поздних версиях — **Current VP Freeze**) — заморозить (слой) в активном окне. Вы-

бранные слои замораживаются, т. е. становятся невидимыми, недоступными для редактирования, и исключаются из состава регенерируемых и печатаемых на бумаге.

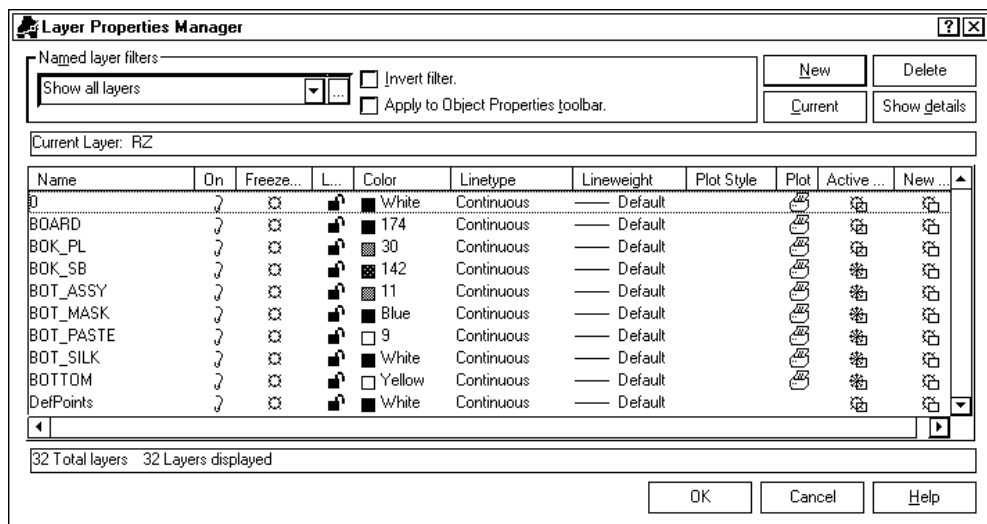


Рис. 14.9. Окно управления видимостью объектов в плавающем видовом окне

В интересующем нас случае работы над чертежом печатной платы так достигается видимость на главном виде проводников, расположенных на стороне установки компонентов (слой Top Layer), контуров платы, исполнительных и справочных размеров, обозначений чистоты обработки, маркировки, координатных рисок с необходимой оцифровкой и т. д., а на виде снизу — контуров платы, проводников со стороны пайки (слой Bottom Layer), координатных рисок с оцифровкой в противоположном направлении.

Границы плавающего окна представляют собой полилинию в текущем активном слое графического редактора. Если мы хотим сделать границы невидимыми на экране, для них следует предварительно создать специальный слой и активизировать его, после чего создавать плавающее окно. Впоследствии можно управлять видимостью слоя границ. Если границы плавающих окон должны быть невидимы только в распечатке, следует вызвать окно управления слоями и в колонке списка слоев с пиктограммой в виде принтера указать опцию отключения печати слоя. После щелчка мышью пиктограмма принтера в данной строке помечается красным значком запрета.

14.4. Подключение форматов к чертежу AutoCAD

Чертежи всегда оформляют на листах стандартных форматов. В ЕСКД это форматы по ГОСТ 2.301-68, с основной надписью первого и последующих листов по ГОСТ 2.104-2006.

Если перед началом черчения ясно, на каком формате будет размещаться чертеж, то при загрузке AutoCAD или при выполнении команды построения нового чертежа **File | New** для подключения форматки следует активизировать в стартовом окне программы (рис. 14.10) кнопку вызова формата **Use a Template**.

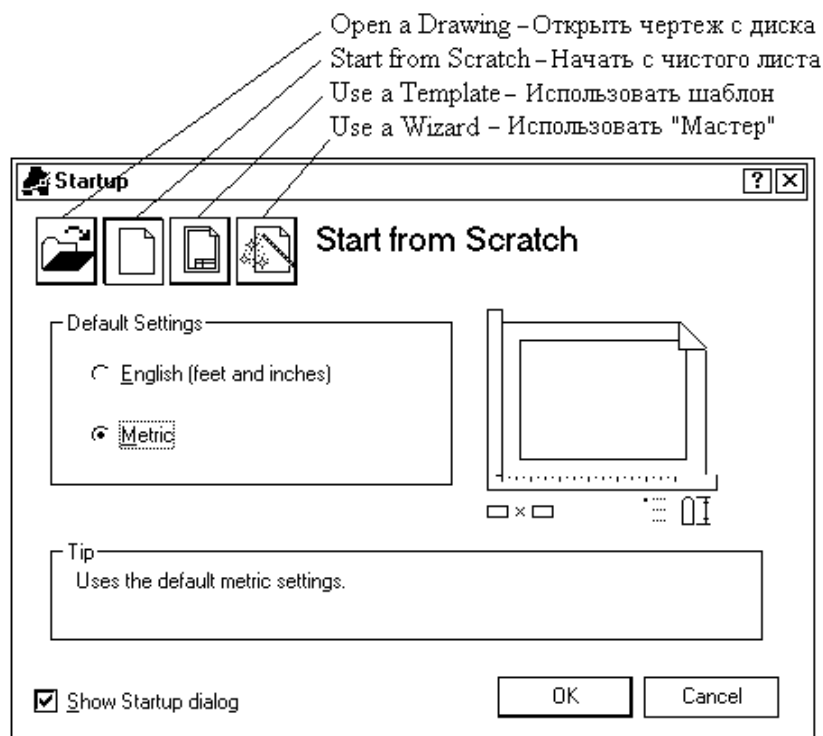


Рис. 14.10. Окно выбора начального шага формирования чертежа

Раскрывается окно **Create New Drawing** (Создать новый чертеж) (рис. 14.11), повторяющее структуру стартового окна. В центральном поле окна **Select a Template** располагается список шаблонов, доступных для загрузки в рабо-

чую среду программы из подкаталога Templates, путь к которому устанавливается по умолчанию при инсталляции AutoCAD. В поставку входят шаблоны, созданные в стандартах ANSI (American National Standards Institute), DIN (Deutsch Industrienormen), ISO (International Standards Organization) и JIS (Japanese Industrial Standard). Бланк форматки ни в одном из этих стандартов не годится для выпуска чертежей по ЕСКД. Кнопка **Browse** позволяет выполнить поиск шаблонов-форматок ЕСКД в пользовательских подкаталогах.

Выбранный шаблон щелчком на кнопке **ОК** загружается в рабочую среду программы. Загрузка происходит в пространство листа. Кроме начертания форматки шаблон включает ряд настроек для будущего чертежа: слои, единицы измерения, стили простановки размеров и др., а также набор подлежащих редактированию атрибутов блока-форматки, образующих записи в основную надпись чертежа.

ЗАМЕЧАНИЕ

Отметим, что приведенные иллюстрации относятся к версии AutoCAD 2000. В более поздних версиях интерфейс постоянно модернизируется, но по существу функции формирования нового документа и вызова шаблона остаются без изменения.

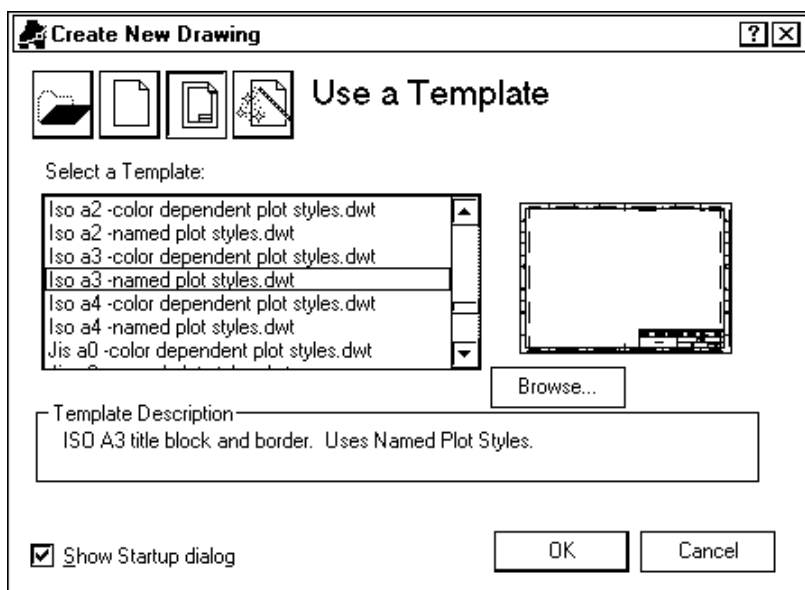


Рис. 14.11. Окно выбора шаблона

Шаблон приносит с собой настройку системы единиц измерения, размеров листа и плавающее видовое окно, границы которого совпадают с внутренней рамкой листа-форматки. При загрузке шаблона это окно автоматически активизируется, и в нем непосредственно можно начать черчение.

Шаблон-форматка изображается целиком на белом поле, размеры которого превышают размеры стандартного листа по ЕСКД (или другой системе стандартов) на несколько миллиметров на каждую сторону, независимо от настройки конфигурации плоттеров или принтеров. При попытке вывода на печать форматки с размещенным на листе чертежом по умолчанию открывается настройка масштаба **Scaled to Fit** (Масштабировать к размерам физического листа плоттера).

При назначении стандартного масштаба 1:1 на листе, физически доступном принтеру или плоттеру, может быть выполнена часть чертежа. После распечатки эта часть изображается на белом поле, остальная часть чертежа — на сером фоне. Разный цвет фона никак не препятствует черчению на всей площади активного плавающего окна. Он означает только то, что распечатка чертежа в назначенном масштабе на данном устройстве возможна лишь по частям.

Выполненный в активном плавающем окне чертеж автоматически переносится в пространство модели проекта. Редактировать чертеж можно как в активном плавающем окне пространства листа, так и в пространстве модели.

Рассмотренным способом формируется один лист чертежа. Выпуск многолистового чертежа или комплекта из нескольких чертежей на основе объекта, созданного в пространстве модели, требует настройки нескольких листов в пространстве листа и, соответственно, применения нескольких разных шаблонов (первый и последующие листы или листы разного формата).

Действия при подключении нового шаблона:

- ☐ указать курсором на вкладку **Layout1** и щелчком правой кнопки мыши вызвать контекстное Pop-Up-меню;
- ☐ в контекстном меню активизировать команду **From Template** (из каталога шаблонов-форматок);
- ☐ в открывшемся стандартном окне поиска файлов найти имя файла подходящей форматки и открыть файл.

Открывается новый лист пространства листа, в активном видовом окне которого размещается весь ранее выполненный в пространстве модели чертеж. Далее, средствами формирования и управления свойствами плавающих видовых окон чертежу в новом листе придают нужный вид.

При импорте DXF/DWG-файла Altium Designer графические данные приходят в пространство модели. Размещение графики на заранее вызванную форматку при этом невозможно. Поэтому подключать форматку следует после приема импортируемых данных.

Существует еще один способ подключения форматки к готовому DWG-документу:

1. Командой **File | Open** открыть необходимый файл форматки *.dwt.
2. Файл форматки открывается как обычный графический документ, при этом графика форматки и ее атрибуты имеют статус блока AutoCAD.
3. В открытом документе селектировать мышью блок-форматку.
4. Командой меню **Edit | Copy** или комбинацией клавиш <Ctrl>+<C> загрузить отмеченный блок в буфер обмена Windows.
5. Активизировать в главном меню AutoCAD команду меню **Window** и перейти в окно проекта.
6. Командой меню **Edit | Paste** или комбинацией клавиш <Ctrl>+<V> загрузить блок-форматку в текущий лист проекта.

При загрузке программа требует указать координаты точки вставки шаблона в текущий лист. Координаты можно ввести с клавиатуры либо указать мышью. Точка привязки блока-форматки обычно назначается в левом нижнем углу листа, поэтому в качестве точки вставки назначить левый нижний угол пределов чертежа (ввести с клавиатуры координаты 0.00, 0.00).

В процессе этой перегрузки шаблона через буфер обмена теряется принадлежащее шаблону плавающее видовое окно. Никакой беды в этом нет: в пространстве листа импортированная графика располагается в своем видовом окне, и дополнительно могут быть созданы новые окна, чтобы чертеж нес всю необходимую информацию об объекте, сформированном в пространстве модели.

Применение в этой процедуре шаблонов ЕСКД, имеющих хождение у пользователей AutoCAD, может вызвать искажение графики при попытке загрузить в разные листы пространства листа бланки различных форматов. Это связано с тем, что блок-форматка имеет одно и то же блоковое имя "РАМКА" во всех этих шаблонах, при разных размерах бланка. Чтобы избежать этой ошибки, следует переименовать блоки-форматки, присвоив им в каждом шаблоне различающиеся имена.

Последовательность действий такова.

1. Активизировать команду главного меню **Format | Rename**. Открывается диалоговое окно переименования — **Rename** (рис. 14.12).

2. В поле **Named Objects** выбрать в качестве объекта блоки — **Blocks**. В поле **Items** появляется список имен блоков, принадлежащих текущему проекту (в данном случае это единственный блок с именем **РАМКА**).
3. Выделить курсором имя **РАМКА**. Это имя появляется в поле "старого имени" **Old Name**.
4. В поле против кнопки **Rename To** (Дать новое имя) внести с клавиатуры новое имя блока, например, РАМКА_A1 и т. п. Щелчком на кнопке **Rename To** активизировать переименование.

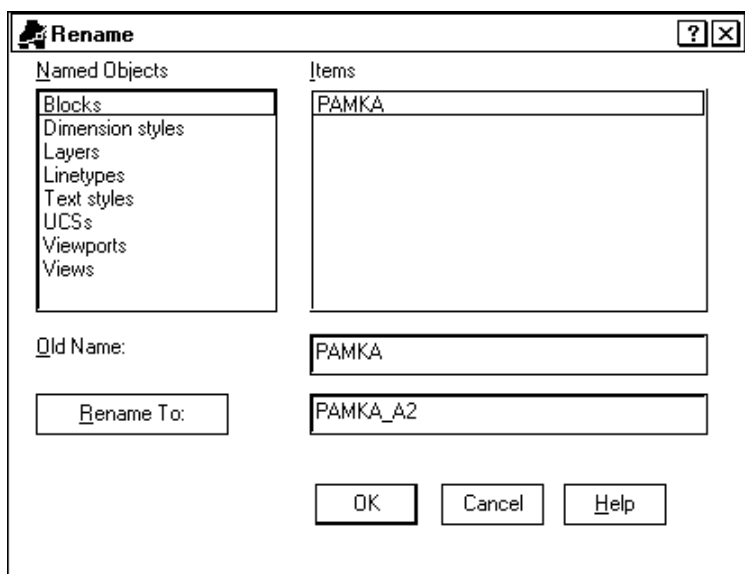


Рис. 14.12. Окно переименования объектов AutoCAD

5. Завершить переименование щелчком на кнопке **OK**.
6. Сохранить файл шаблона под новым именем в дисковой памяти.

14.5. Заполнение реквизитов документа

Записи в основную надпись чертежа (реквизиты документа) представляют собой атрибуты блока-форматки. Для заполнения основной надписи в плавающем меню **Modify II** нужно выполнить команду **Edit Attributes**. Откры-

вается диалоговое окно редактирования атрибутов блока (рис. 14.13), в центральном поле которого расположен список атрибутов, соответствующих реквизитам документа, а в поле **Value** вносятся значения реквизитов.

При оформлении неосновных конструкторских документов, которым присваивается код документа по ГОСТ 2.201-80, в первой графе основной надписи по ГОСТ 2.104-2006 второй строкой под наименованием изделия должно помещаться наименование документа. На сборочном чертеже функционального узла (модуля) это должна быть запись "Сборочный чертеж". В диалоговом окне, показанном на рис. 14.13, поле для этой записи не предусмотрено.

Block: ПАМКА_A3_rop
Tag: TECH CONTR

Select block

Attribute | Text Options | Properties

Tag	Prompt	Value
DEVELOPED...	Разработал	Иванов
DEV_DATE	Дата разработки	25.10.07
CHECKED_BY	Проверил	Петров
CHK_DATE	Дата проверки	28.10.07
TECH_CONTR	Техн. контроль	Сидоров
T_CONTR_D...	Дата техн. контроля	31.10.07
AUX_LEFT	Доп. слева	
AUX_RIGHT	Доп. справа	
AUX_DATE	Доп. дата	
NOM_CONTR	Нормоконтроль	Зоркий
NOM_CONTR...	Дата нормоконтроля	10.11.07
APPROVED_...	Утвердил	Главный
APPROVED_...	Дата утверждения	20.12.07
DOCUMENT	Обозначение доку...	АБВГ.758726.012
NAME	Наименование изд...	Плата печатная
MATERIAL	Материал детали	СФ2-35-1,5 ГОС...
MASS	Масса	75 гр.
SCALE	Масштаб	1:1
DIFF_INDEX	Индекс предприятия	000 Пога и Кон...
PAGE_NO	Номер листа	
PAGES_AM...	Число листов	1
LITER_1	Литера 1	0
LITER_2	Литера 2	
LITER_3	Литера 3	

Value:

Apply OK Cancel Help

Рис. 14.13. Окно заполнения реквизитов документа

Эту трудность легко преодолеть без перепрограммирования окна. Необходимо запись можно внести, например, вместо реквизита "Материал детали" (на сборочном чертеже третья графа основной надписи остается пустой), а по окончании редактирования записей можно селектировать блок-рамку, взять блок-запись из третьей графы за "прищепку" и перенести ее в первую графу основной надписи.

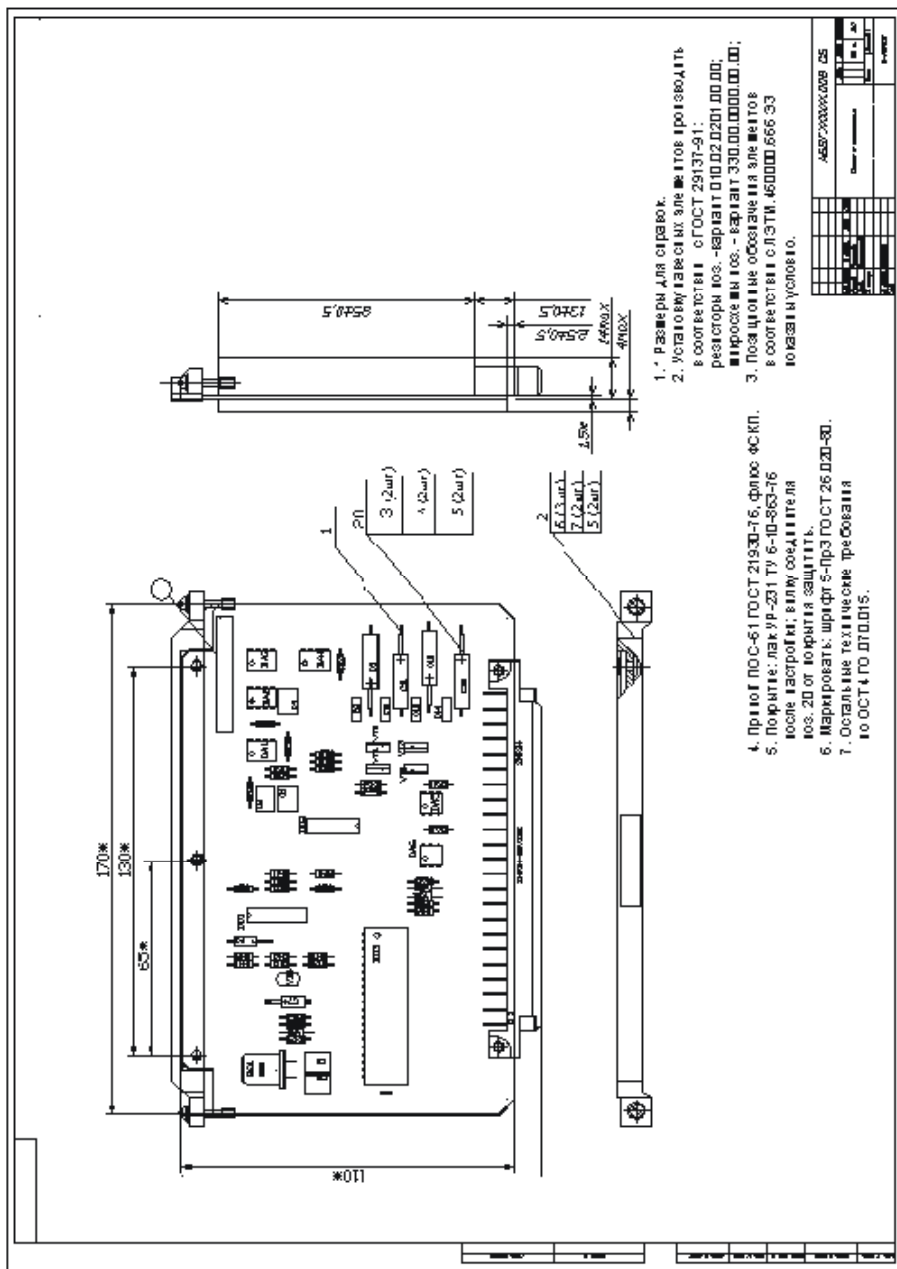
Щелчком на кнопке **ОК** редактирование завершают. Внесенные сведения отображаются в графах основной надписи на чертеже.

14.6. Формирование чертежей функционального узла

Мы рассмотрели основные функции и приемы работы в среде САПР AutoCAD 2000, позволяющие, располагая графическими данными, импортированными из Altium Designer, как и из других "электронных" САПР, выпустить два графических конструкторских документа по ЕСКД: чертеж печатной платы по ГОСТ 2.417-91 и сборочный чертеж функционального узла по ГОСТ 2.109-73.

Приведем теперь последовательность операций, выполнение которых позволяет получить требуемый результат (рис. 14.14 и 14.15).

1. Открыть полученный в результате конверсии DWG-файл проекта командой **File | Open** (для файла в формате DXF указать в окне поиска файлов имя *.dxf).
2. Создать новые слои AutoCAD для размещения на них:
 - обозначения размеров на чертеже печатной платы и сборочном чертеже модуля;
 - обозначения чистоты обработки поверхностей печатной платы;
 - линеек с рисками координатной сетки для видов лицевой и обратной сторон печатной платы;
 - деталей сборки модуля и контуров для условно-упрощенного изображения сборки;
 - текстовых технических требований для чертежа печатной платы и сборочного чертежа модуля;
 - плавающих видовых окон.



3. Вычертить контуры платы и сборки в виде сбоку — с ребра (можно вместо этого вытянуть экструзией контуры платы в трехмерное тело и работать с платой как с трехмерным объектом).
4. Пользуясь функциями команды **Dimension**, нанести в соответствующих слоях линейные размеры для платы и сборки модуля и указать предельные отклонения исполнительных размеров.
5. Командой **Insert Block** подгрузить к чертежу заранее подготовленные объекты-блоки, изображающие:
 - линейки с рисками координатной сетки по ГОСТ 2.417-91;
 - обозначения шероховатости поверхностей по ГОСТ 2.309-73;
 - детали, входящие в состав сборки модуля.

ЗАМЕЧАНИЕ

Отметим, что при проектировании функциональных узлов РЭС на основе базовых несущих конструкций (БНК) пп. 2...5 можно выполнить один раз для целой группы проектов, чтобы избавить конструктора от черчения контуров платы, простановки размеров и т. д. в каждом проекте. При этом подготовленный заранее шаблон-БНК с нанесенными размерами, координатными линейками и др. знаками открывается как документ AutoCAD, после чего операциями селекции, копирования в буфер обмена Windows с указанием базовой точки (**Edit | Copy with Base Point**; базовую точку указать в одном из углов контура платы или в центре крепежного отверстия) и вставки из буфера обмена в пространство модели (командой **Edit | Paste** или **<Ctrl>+<V>**) шаблон-БНК накладывается на импортированный вид печатной платы.

6. Составить текстовые технические требования на поле чертежа по ГОСТ 2.316-68. Тексты технических требований для чертежа печатной платы и для сборочного чертежа модуля пишутся каждый в своем слое и размещаются на свободном пространстве графического листа.
7. Перейти в пространство листа, в котором открыть и сконфигурировать два листа: **Layout1** и **Layout2**. На первом листе будем формировать чертеж печатной платы по ГОСТ 2.417-91, а на втором — сборочный чертеж модуля первого уровня разукрупнения по ГОСТ 26632-85.
8. Подключить к каждому листу бланк форматки по ГОСТ 2.301-68, с основной надписью по ГОСТ 2.104-2006 из пользовательского каталога **Templates**.

9. В первом листе **Layout1** открыть новые плавающие видовые окна (Viewports). Установить в новых окнах изображение вида снизу (Bottom) и слева (Left) и назначить во всех видовых окнах единый стандартный масштаб по ГОСТ 2.302-68.
10. Активизировать по очереди каждое из видовых окон и "заморозить" в каждом окне видимость слоев, "лишних" для соответствующего вида платы (функция **Freeze in Active Viewport** в AutoCAD 2000 или **Current VP Freeze** в более поздних версиях). В видовых окнах листа изображаются в проекционной связи три вида печатной платы: главный вид (сторона установки компонентов), вид сбоку и вид со стороны монтажа, текстовые технические требования и другие данные.
11. Перейти на лист **Layout2** и выполнить действия пп. 9 и 10, управляя видимостью слоев проекта так, чтобы остались видимыми элементы графики, образующие сборочный чертеж функционального узла.
12. Составить реквизиты чертежа печатной платы и сборочного чертежа модуля. Конечный результат этой работы показан на рис. 14.14 и 14.15.
13. Сохранить результаты на жестком диске компьютера в формате AutoCAD. Все виды сохраняются в едином файле (файл *.DWG).
14. Вычертить чертежи на плоттере или напечатать на принтере. Каждый лист распечатывается по команде **File | Plot**.

Список литературы

1. Потапов Ю. В. Protel DXP. — М.: Горячая линия – Телеком, 2006.
2. Разевиг В. Д. Проектирование печатных плат в P-CAD 2001. — М.: СОЛОН-Р, 2001.
3. Прангишвили И. В., Абрамова Н. А., Бабичева Е. В., Игнатушенко В. В. Микроэлектроника и однородные структуры для построения логических и вычислительных устройств. — М.: Наука, 1967.
4. Стешенко В. Б. ПЛИС фирмы ALTERA: проектирование устройств обработки сигналов. — М.: Додека, 2000.
5. Разевиг В. Д. Применение программ P-CAD и PSpice для схемотехнического моделирования на ПЭВМ: В 4-х вып. — М.: Радио и связь, 1992.
6. Разевиг В. Д. Система сквозного проектирования электронных устройств DesignLab 8.0. — М.: Солон, 1999.
7. RS-274X Format User's Guide. Barco Gerber Systems Corp. South Windsor, CT, USA, 1998.
8. Романычева Э. Т., Соколова Т. Ю. Компьютерная технология инженерной графики в среде AutoCAD 2000. Учебное пособие. — М.: ДМК Пресс, 2001.
9. Уваров А. С. AutoCAD 2002 для конструкторов. — М.: ДМК Пресс, 2002.
10. Полещук Н. Н., Савельева В. А. Самоучитель AutoCAD 2004. — СПб.: БХВ-Петербург, 2003.

Глоссарий

API (Application Programming Interface) — интерфейс программирования в среде пакета прикладных программ (приложения).

ASIC (Application-Specific Integrated Circuit — Специализированная интегральная схема для решения конкретной задачи) — "заказная" интегральная микросхема высокой степени интеграции, выполняющая строго ограниченные функции, характерные только для конкретного устройства.

AutoCAD — широко распространенная система графического моделирования деталей машин и механизмов, строительных конструкций и т. п.

CAD (Computer-Aided Design — Проектирование с помощью компьютера) — русскоязычный эквивалент — термин САПР (Система автоматизированного проектирования).

CAM (Computer-Aided Manufacturing — Изготовление с помощью компьютера) — программно-аппаратные комплексы автоматизированного производственного оборудования.

CALS (Continuous Acquisition and Life cycle Support — Непрерывная поддержка обретения (разработки и изготовления) и жизненного цикла) — современный подход к проектированию и производству высокотехнологичной и наукоемкой продукции, заключающийся в использовании компьютерной техники и информационных технологий на всех стадиях жизненного цикла изделия.

CPLD (Complex Programmable Logic Device — Сложный программируемый логический прибор) — разновидность программируемых интегральных микросхем, состоящих из блоков логических вентилей, объединенных программируемой коммутационной матрицей. Современные CPLD, как правило, электрически перепрограммируемые и сохраняют логическую структуру после отключения питания. В отечественной схемотехнике подобные приборы, независимо от их сложности и технологии выполнения коммутации, обозначают термином ПЛИС (Программируемая логическая интегральная схема).

Devices View (дословно Вид приборов) — диалоговая оболочка, в которой выполняется управление основными фазами реализации проекта ПЛИС: выбор микросхемы ПЛИС для реализации проекта, трансляция исходного опи-

сания логики ПЛИС в конструкции языка обмена данными EDIF, передача EDIF-описания логики в САПР производителя ПЛИС (см. **Vendor Tools**), получение обратно результата размещения логики на кристалле выбранной ПЛИС и программирование ПЛИС на отладочной панели (см. **Nano Board**) или в аппаратуре пользователя.

Directory (Директория) — каталог в структуре записей в памяти компьютера. В локализованных ОС употребляется термин "Папка".

DLL (Dynamic Link Library — Динамически связываемая библиотека) — средство операционной системы, позволяющее хранить подпрограммы, выполняющие определенные функции большого программного комплекса, в виде файлов с расширением dll. В частности в Altium Designer все функциональные модули пакета хранятся в памяти компьютера как dll-файлы и вызываются из интегрирующей DXP-оболочки пакета.

Drag-and-Drop (дословно Тащи и бросай) — используемый в ОС Microsoft Windows прием перемещения и/или копирования массивов данных (файлов) путем указания курсором мыши на имя файла или пиктограмму-указатель на этот файл, перемещения этого имени или указателя при нажатой левой кнопке мыши из одного диалогового окна в другое и оставление его на новом месте после отпускания кнопки мыши.

Drag Mode (Режим перетаскивания) — функция редактирования графических объектов, характерная в Altium Designer тем, что при перемещении объекта по полю чертежа на экране он не отрывается от других объектов, электрически связанных с ним, а вытягивает эти связи за собой.

DRC (Design Rules Check — Контроль выполнения правил проектирования) — операция верификации проекта, выявляющая нарушения установленных правил проектирования печатной платы: уменьшение зазоров между элементами печатного рисунка ниже допустимого значения, разрывы, короткие замыкания цепей и т. п.

Drop-Down Menu ("Выпадающее" меню) — список подкоманд, разворачивающийся на экране монитора при указании курсором команды главного меню программы.

DXF (Data eXchange Format) — текстовый формат, в который и из которого конвертируются проектные данные САПР электронных функциональных узлов при обмене с "машиностроительными" конструкторскими САПР, в частности с AutoCAD.

DXP (Design eXPlorer — дословно Обозреватель проекта) — интегрирующая оболочка, организующая порядок использования графических редакто-

ров и других встроенных программных модулей Altium Designer и технологию обработки данных.

ECO (Engineering Changes Order — дословно **Приказ на инженерные изменения**) — заложенная в Altium Designer диалоговая процедура внесения изменений в документы текущего открытого проекта.

EDIF (Electronics Design Interchange Format) — язык описания, применяемый для обмена проектными данными между САПР электронных функциональных узлов.

EPLD (Erasable Programmable Logic Device — **Стираемый программируемый логический прибор**) — разновидность программируемых интегральных микросхем, состоящих из блоков логических вентилях, объединенных коммутационной матрицей, в которых действующая программа при перепрограммировании стирается ультрафиолетовым излучением и/или электрическим сигналом.

FPGA (Field-Programmable Gate Array — дословно **Матрица логических вентилях, программируемая полем**) — разновидность программируемых интегральных микросхем, состоящих из блоков логических вентилях, объединенных коммутационной матрицей, программируемой с помощью МДП-транзисторов с изолированным затвором, выполненных на пересечениях линий внутренних магистралей и сохраняющих заряд, сообщенный при программировании.

Grips (Прищепки) — специальные значки, чаще всего в виде маленьких квадратиков, которыми программа помечает узловые точки контура графических объектов при их выборе на экране монитора для редактирования. Перемещая "прищепки" при нажатой левой кнопке мыши, можно изменять форму контура объекта.

HDL (Hardware Description language — **Язык описания аппаратуры**) — группа специализированных языков программирования высокого уровня, предназначенных для описания логики функционирования электронных устройств взамен или в дополнение к привычному радиоинженеру схемному описанию. Известны HDL-языки, разработанные фирмами — производителями ПЛИС для проектирования устройств на "своих" микросхемах (например, язык AHDL фирмы Altera). Другие HDL-языки являются универсальными и могут применяться для описания логики ПЛИС безотносительно к особенностям микросхем конкретного производителя (VHDL, VerilogHDL). Специализированные САПР выполняют трансляцию HDL-описания проектируемого устройства и формируют команды программирования внутренней коммутации ПЛИС под задачу пользователя.

IBIS-модели (I/O Buffer Information Specification — дословно **Спецификация информации о входных/выходных буферах**) — математические модели входных/выходных буферов интегральных микросхем, контактов электрических соединителей и других ЭРК, служащие для моделирования паразитных эффектов в печатном монтаже программой Signal Integrity.

JTAG (Joint Test Action Group) — специализированный интерфейс для внутрисхемного программирования и отладки логики ПЛИС (стандарт IEEE 1149.1). Из-за широкой функциональности JTAG стал повсеместно использоваться для отладки и программирования логических устройств. Работа средств обеспечения интерфейса JTAG подчиняется сигналам автомата управления, встроенного в микросхему ПЛИС.

Layout (дословно **Раскладка**). В некоторых САПР термин применяется в наименовании подсистем проектирования печатной платы (PCB Layout); в AutoCAD данным термином обозначается "пространство листа" программы, в котором выполняется подготовка чертежа к печати на бумажный носитель.

Model Space — *см. Пространство модели.*

Move Mode (Режим перемещения) — функция редактирования графических объектов, характерная в Altium Designer тем, что при перемещении объекта по полю чертежа на экране его электрические связи разрываются.

NanoBoard — поставляемая по отдельному заказу аппаратная отладочная панель, на которой производится отладка логики и программирование ПЛИС, выбранной для реализации проекта логического устройства. Панель NanoBoard связывается с компьютером по линиям JTAG-интерфейса.

Paper Space — *см. Пространство листа.*

P-CAD (Personal Computer-Aided Design) — одна из широко распространенных импортных САПР радиоэлектронных функциональных узлов на печатных платах. После 2006 г. дальнейшее развитие и поставка системы на рынок прекращены.

PCB (Printed Cards Board) — аббревиатура, служащая для обозначения печатной платы, а также графического редактора печатных плат во многих САПР электронных устройств (в Altium Designer это DLL-подпрограмма AdvPCB.dll). С точки зрения правил русского языка дословный перевод термина представляется избыточным, поскольку Printed Cards дословно переводится как "печатные карты", а Board — как "пластина". По-русски вполне достаточен перевод "печатная плата".

PDF (Portable Data Format) — портативный (сжатый) формат данных, применяемый для формирования и хранения текстовых документов и рисунков в программах Adobe Acrobat.

PDF (P-CAD Data Interchange Format) — текстовый формат обмена данными в ранних версиях САПР электронных устройств P-CAD 3.0 – 8.7, используемый также при обмене данными этих ранних версий с версиями P-CAD 2000 – 2006.

Pop-up Menu (Всплывающее контекстное меню) — список команд, разворачивающийся по щелчку правой кнопки мыши и зависящий от контекста, т. е. от того, на какой из объектов или в какую область экрана монитора указывает курсор мыши.

Schematic — схемный документ проекта электронного устройства, а также обозначение в САПР электронных устройств программы графического редактора электрической принципиальной схемы.

Signal Integrity (Целостность сигнала) — подсистема моделирования паразитных эффектов (отражений и взаимных наводок) в проводниках печатного монтажа.

SMD (Surface Mount Device — Компонент, монтируемый на поверхность) — тип ЭРК, называемый в отечественной практике "планарным" или компонентом в корпусе типа 4 по ГОСТ 17467-88. Ленточные выводы, таких компонентов перед монтажом подвергают формовке, чтобы добиться прилегания к плоскости печатной платы, после чего выводы припаивают к контактным площадкам.

SMT (Surface Mount Technology — Технология поверхностного монтажа) — аббревиатура SMT в англоязычной лексике Altium Designer распространяется также и на поверхностно-монтируемые компоненты.

SPICE (Simulation Program with Integrated Circuits Emphasis — дословно Программа моделирования, акцентированная на интегральные схемы) — данной аббревиатурой обозначается широко распространенная (практически мировой стандарт) технология схемотехнического моделирования. Подсистемы моделирования, основанного на SPICE-технологии, включены в большинство САПР электронных функциональных узлов: OrCAD, P-CAD, MicroCAP, Protel, Altium Designer и др.

SPICE-модели — математические модели электронных компонентов, предназначенные для схемотехнического моделирования на основе SPICE-технологии.

Vendor Tools (дословно — Инструменты поставщика) — в интегрированной среде проектирования Altium Designer — специализированные САПР (разработанные фирмами — поставщиками ПЛИС), выполняющие трансляцию схемного или/и HDL-описания логики функционирования электронного

устройства, реализуемого на ПЛИС, в систему логических примитивов выбранной ПЛИС и схему коммутации внутренних логических связей, формирующие программу коммутации внутренней матрицы ПЛИС, подключения входных и выходных логических сигналов к внешним выводам микросхемы и возвращающие результат в Altium Designer для завершения проекта функционального узла.

* * *

ЕСКД (Единая система конструкторской документации) — комплекс государственных стандартов Российской Федерации, устанавливающих взаимосвязанные правила и положения по порядку разработки, оформления и обращения конструкторской документации.

Контекстное меню — меню программы, состав команд которого зависит от контекста, т. е. от того, какая из подпрограмм или подсистем большого пакета прикладных программ работает в текущий момент. В Altium Designer главное меню DXF-оболочки пакета является контекстным. Контекстными являются также "всплывающие" меню, вызываемые в графическом окне программы по щелчку правой кнопки мыши.

Перечень элементов — список элементов, образующих принципиальную схему устройства (независимо от функционального назначения устройства). Правила формирования перечня элементов закреплены в ГОСТ 2.701-80 и ГОСТ 2.702-75.

ПЛИС (Программируемая логическая интегральная схема) — установленное в отечественной схемотехнике определение для интегральных микросхем, функции которых пользователь программирует под конкретную прикладную задачу обработки информации. Термин употребляется вне зависимости от технологии выполнения внутренней коммутации — однократного "прожига", электрического перепрограммирования, стирания с помощью ультрафиолетового излучения и т. п.

ПЛИМ (Программируемая логическая матрица) — обозначение программируемых интегральных микросхем, исторически предшествующее появлению термина ПЛИС.

Пространство модели (Model Space) — режим работы AutoCAD, в котором выполняется графическое моделирование объектов.

Пространство листа (Paper Space или Layout) — режим работы AutoCAD, в котором выполняется формирование чертежа видов объекта, выполненного

в процессе графического моделирования в пространстве модели, для печати на "твердый" (бумажный) носитель.

Радиоэлектронное средство (РЭС) — в соответствии с ГОСТ 26632-85 этим термином обозначают изделие и его составные части, в основу функционирования которых положены принципы радиотехники и электроники.

Радиоэлектронный функциональный узел — в соответствии с ГОСТ 26632-85 — радиоэлектронное средство, представляющее собой функционально законченную сборочную единицу, выполненную на несущей конструкции, реализующее функцию преобразования сигнала и не имеющее самостоятельного эксплуатационного применения.

Реквизиты документа — записи в основной надписи конструкторского документа, содержащие обозначение документа, наименование изделия, фамилии и подписи лиц, уполномоченных для работы с документом, другие сведения (всего до 34 записей).

Спецификация — в соответствии с ГОСТ 2.102-68 "Виды и комплектность конструкторских документов" — основной конструкторский документ на сборочную единицу, комплекс или комплект, представляющий собой список составных частей изделия. Правила составления спецификации установлены ГОСТ 2.106-96.

Форматка — бланк, на котором выполняются конструкторские документы. Формат листа для конструкторских документов устанавливает ГОСТ 2.301-68 "Форматы"; состав, расположение и содержание записей, определяющих принадлежность документа — ГОСТ 2.104-2006 "Основные надписи".

ЭРК (Электрорадиокомпонент) — покупное изделие или изделие собственного производства, являющееся составной частью радиоэлектронного функционального узла и выполняющее элементарную (резистор, конденсатор) или более сложную функцию преобразования электрического сигнала (аналоговая или цифровая интегральная микросхема).

ЭРЭ (Электрорадиоэлемент) — элемент, совокупность которых вместе с объединяющими их условными обозначениями электрических связей образует электрическую принципиальную схему радиоэлектронного функционального узла. Ввиду отсутствия стандартизированной терминологии различие между ЭРЭ и ЭРК размыто: так, по ЕСКД дополнительно к принципиальной схеме выпускается документ, называемый "Перечень элементов", хотя в нем на самом деле перечисляются именно ЭРК с полным обозначением типа, параметров, вида исполнения, документа на применение.

Предметный указатель

A

AC Small Signal Analysis 258
API 9
ASIC 179
AutoCAD 2002 433

B

Bed-of-nails testing 223
Bill Of Materials 401
Board Implementation 6
Bus Joiner 188

C

CAMtastic 416
Cluster Placer 126
Concurrent Versions System 350, 381
Core Project 9
CPLD 180
Crosstalk 326

D

Designator 98
Devices View 210, 221
Digital SimCode 298
DXP 5

E

ECO-процедура 109, 361
EDIF 218
Embedded Intelligence Implementation 6
Embeded Project 9
EPLD 180

F

Fast Fourier Transform *См.* БПФ
Flash PLD 180
Foundation 5
FPGA 180
FPGA Project 8
FPGA to PCB Project Wizard 227

G

Graphical Bus 92
Grips 94

H

Hot Spot 85

I

Instrument Rack 224

J

JTAG 188

L

Library Splitter Wizard 390

Logical Bus 92

M

Microsoft Source Code Control Interface

381

Model Space 440

MSCS 241, 298

N

NanoBoard 211, 225

Node Bridges 319

P

Paper Space 440

PCB Layout 6

PCB Project 7

Push'n'shove 139

Q

Query Language 369

R

Reference Point 37

Reflections 326

Rise Time 277

RS274D 419

RS274X 419

S

Schematic Symbol 243

Signal Integrity 325

Situs 162

Slam'n'jam 139

SMD 40

SPICE 241

Statistical Placer 126

Subversion 381

T, V, W

Teardrops 425

VHDL 218

Wired Function 188

А

Анализ:

- взаимных наводок 345
- импульсной характеристики 277
- малосигнальный 258
- параметрический 262
- передаточной функции 282
- переходных процессов 273
- по постоянному току 281
- случайных разбросов параметров компонентов 285
- шумовых характеристик 294

Б

БНК 458

БПФ 279

Д

Депозиторий 382

Дерево классов 111

Директива:

- моделирования 257

Дифференциальные пары 152

- правила трассировки 159

- рекомендации по выполнению 154

Документ:

- активизация 12
- возврат в депозиторий 399
- заполнение основной надписи 453
- "неприсоединенный" 384
- основная надпись 63
- присоединение к системе контроля версий 385
- реквизиты 61
- свободный 10
- связанный 10
- сравнение версий 389

З

Запретная зона 76

Запрос:

- сложный 359

И

Идентификатор:

- цепи 87

Интегрированная библиотека

- включение в рабочую среду 54
- формирование 48

К

Компонент:

- многосекционный 29
- описание 15
- поверхностно-мантируемый 40
- поиск по совокупности признаков 358
- поиск в базе данных 369
- поиск в библиотеке 55
- разнесенный способ формирования схемного символа 29
- редактирование 16
- редактирование свойств 32
- редактирование свойств выводов 27
- создание УГО 19
- формирование 17
- формирование набора сведений 101
- функциональное назначение выводов 26
- черчение контуров корпуса 39

Компоненты:

- авторазмещение 126
- интерактивное размещение 124
- ручное размещение 122

Конфликт:

- при разводке 138

Курсор:

Small 85

Tiny 85

Л

Логический символ 15

М

Миграция библиотек 374

Мода:

Live 211

Not Live 211

Модель 16

"поведенческая" 242

IBIS 326

Signal Integrity 45

SPICE 44

SPICE-префиксы 249

компонента 242

операционного усилителя 244

присоединение к схемному

компоненту 41

цифровой интегральной

микросхемы 299

Н

Настройка:

анализа нулей и полюсов

передаточной функции 283

анализа переходных процессов 274

параметров контактных площадок 40

параметров модели 311

параметров электрических цепей 114

таблиц базы данных 365

формата данных для сверления 412

экспорта в формат PDF 405

экспорта в формат фотоплоттера

Gerber 419

О

Объект:

графический 80

группирование в классы 111

электрический 80

П

ПЛИС 180

выбор микросхемы 203

конфигурация схемного символа 231

образование нового проекта 184

программирование 217

формирование логической

структуры 194

Позиционное обозначение 98

Правило:

авторазмещения 128

бинарное 120

настройка 113

проектирования 75

унарное 119

Проводник печатный:

варианты выполнения 136

Проект 7

активизация 13

включение документов 10

встроенный 9

история 349

компиляция 107

логического ядра 9

печатной платы 7

ПЛИС 8

процедура создания 9

связь FPGA и PCB.

См. FPGA to PCB Project Wizard

Проектирование:

сквозное 227

Р

Расщепление библиотек. См. Library
Splitter Wizard

С

Сетка:

формирование 67

Скрипт-проект 9

Скрытый вывод

формирование 31

"Слезка" 425

Слой:

автоматическое назначение 430

определение структуры 68

специальный 73

структура по умолчанию 35

типы 72

Суффиксы масштаба 252

Т

Технологический стандарт:

интегральной микросхемы 330

ТПМ 15

формирование новой библиотеки 33

Трассировка:

"тонкая" 148

проходы 166

ручная 132

стандартная 136

стратегии автотрассировки 163

функции автотрассировки 168

У

УГО 16

метки внутри корпуса 26

указатели 27

УГО биполярного транзистора:

построение 19, 20

присоединение выводов 22

Ф

Файл:

BOM. См. Bill Of Materials

Database Library file 353, 361

Database Link file 353

выходной Gerber 426

задания на моделирование 256

командной информации для станка

с ЧПУ 414

модели 243

описания модели 246

описания на языке SimCode 302

связи с базой данных 353

структуры ODB++ 428

формата DRR 414

формата EXTREP 423

формата NSX 319

формата SCB 305

ФНЧ:

Баттерворта 282

Чебышева 273

Форматка:

подключение 77

подключение к DWG-документу 452

формирование бланка 58

Х

"Худший случай" 288

Я

Язык:

описания моделей цифровых

компонентов 298

построения запросов 359