

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*

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 Смирновой</i>
Корректор	<i>Наталья 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 70×100^{1/16}. 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

ISBN 978-5-9775-0542-0

© Суходольский В. Ю., 2010
© Оформление, издательство "БХВ-Петербург", 2010

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

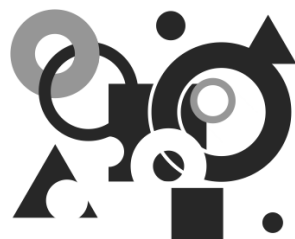
ГЛАВА 9. СХЕМОТЕХНИЧЕСКОЕ МОДЕЛИРОВАНИЕ241

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 депозитория 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

ГЛАВА 1



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.

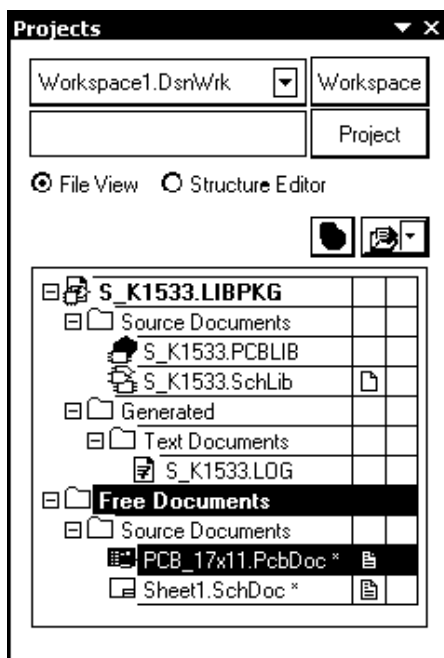


Рис. 1.1. Дерево документов панели **Projects**

3. 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**.
4. 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:

1. 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).

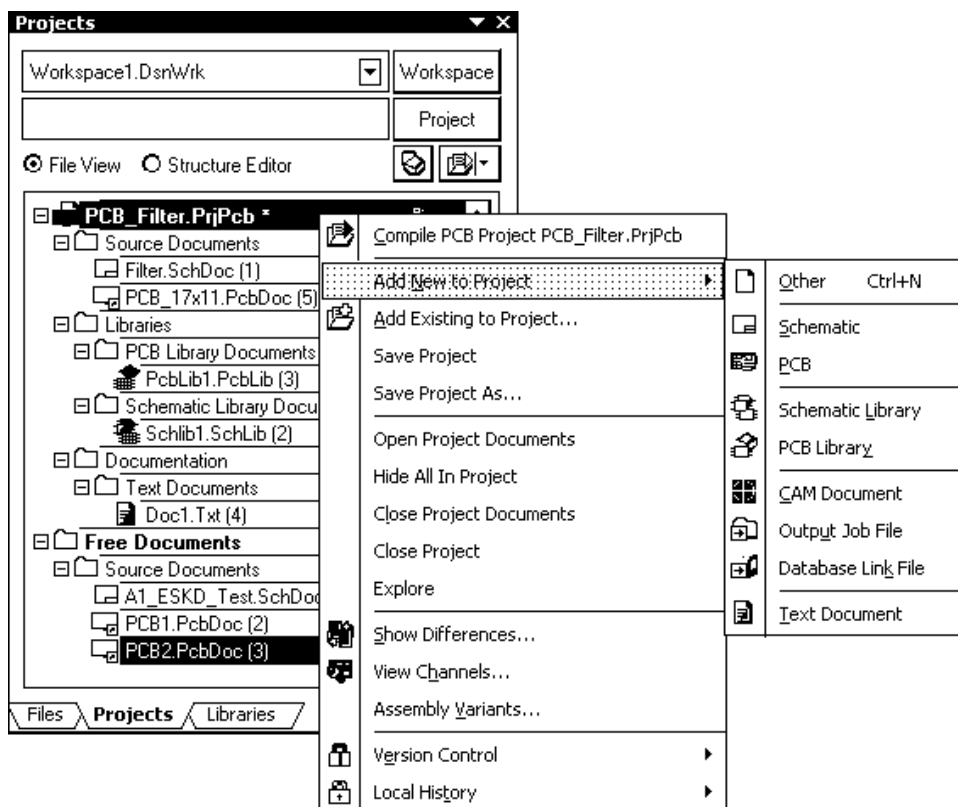


Рис. 1.2. Меню команд управления документами проекта

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 связи 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.

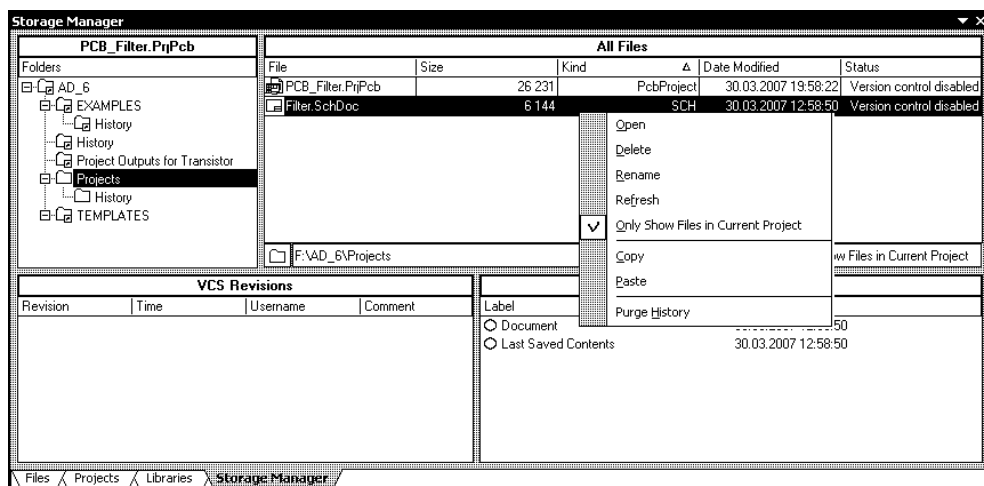


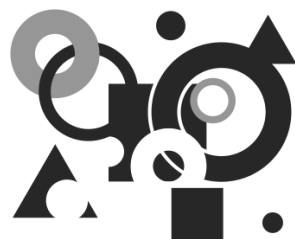
Рис. 1.3. Рабочая панель **Storage Manager**

Отметим также, что дерево документов в плавающей панели **Projects** показывает 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.

ГЛАВА 2



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.