

УДК 681.3(075.8)
ББК 32.973-02я73
Г91

Грушвицкий Р. И., Мурсаев А. Х., Угрюмов Е. П.
Г91 Проектирование систем на микросхемах программируемой логики. —
СПб.: БХВ-Петербург, 2002. — 608 с.: ил.
ISBN 5-94157-002-3

В книге систематически изложены основные архитектурные и схемотехнические особенности современных больших интегральных схем с программируемой структурой (ИСПС) всех ведущих производителей, в том числе особенности реконфигурируемых систем на кристалле. Рассмотрены методология, маршруты проектирования и специфика основных этапов создания проектов на ИСПС с ориентацией на использование систем автоматизированного проектирования. Представлены наиболее распространенные языки описания цифровых устройств и их место в процессе проектирования.

*Для студентов, аспирантов, преподавателей профильных специальностей,
а также инженеров-разработчиков вычислительных, управляющих,
радиотехнических и телекоммуникационных систем*

УДК 681.3(075.8)
ББК 32.973-02я73

Группа подготовки издания:

Главный редактор	Екатерина Кондукова
Зав. редакцией	Анна Кузьмина
Редактор	Григорий Добин
Компьютерная верстка	Ольги Сергиенко
Корректор	Зинаида Дмитриева
Дизайн обложки	Игоря Цырульников
Зав. производством	Николай Тверских

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 19.06.02.

Формат 70×100^{1/8}. Печать офсетная. Усл. печ. л. 49.

Тираж 3000 экз. Заказ № 273

"БХВ-Петербург", 198005, Санкт-Петербург, Измайловский пр., 29.

Гигиеническое заключение на продукцию, товар № 77.99.02.953.Д.001537.03.02
от 13.03.2002 г. выдано Департаментом ГСЭН Минздрава России.

Отпечатано с готовых диапозитивов
в Академической типографии "Наука" РАН
199034, Санкт-Петербург, 9 линия, 12.

ISBN 5-94157-002-3

© Грушвицкий Р. И., Мурсаев А. Х., Угрюмов Е. П., 2002
© Оформление, издательство "БХВ-Петербург", 2002

Содержание

Предисловие.....	9
Введение.....	13
Глава 1. Элементная база электронных устройств и систем. Архитектура и схемотехника интегральных схем с программируемой структурой.....	23
1.1. Общие сведения. Предшественники. Классификация.....	23
1.1.1. Вводные замечания. Историческая справка.....	23
1.1.2. Базовые матричные кристаллы (вентильные матрицы).....	28
1.1.3. Классификация логических микросхем программируемой логики.....	31
1.1.4. Общие (системные) свойства микросхем программируемой логики.....	41
1.2. CPLD — сложные программируемые логические устройства.....	42
1.2.1. Структура CPLD.....	42
1.2.2. Программируемая матрица соединений.....	44
1.2.3. Функциональные блоки CPLD.....	46
1.2.4. Блоки ввода/вывода CPLD.....	50
1.3. FPGA — программируемые пользователем вентильные матрицы.....	52
1.3.1. Основные сведения.....	52
1.3.2. Функциональные блоки FPGA.....	54
1.3.3. Блоки ввода/вывода FPGA.....	56
1.3.4. Системы межсоединений.....	58
1.4. ПЛИС с комбинированной архитектурой.....	61
1.4.1. Структура микросхем семейства FLEX.....	61
1.4.2. Логические элементы.....	63
1.4.3. Встроенные блоки памяти.....	65
1.5. Основные сведения о СБИС типа "система на кристалле".....	67
1.6. СБИС типа "система на кристалле" с однородной структурой (с полностью синтезируемыми блоками).....	70
1.6.1. Микросхемы семейств APEX20K/KE, APEXII.....	70
1.6.2. Микросхемы семейств Virtex, Virtex E, Virtex II.....	74

1.7. СБИС типа "система на кристалле" с блочными структурами (с аппаратными ядрами)	79
1.7.1. Вводные замечания	79
1.7.2. SOPC с блочной структурой, не содержащие ядер процессоров	80
1.7.3. SOPC семейства FPSLIC фирмы Atmel	82
1.7.4. SOPC блочного типа фирмы Triscend	88
1.7.5. SOPC блочного типа фирмы Altera	91
1.8. Микросхемы с программируемыми аналоговыми и аналого-цифровыми структурами	92
1.8.1. Общие сведения	92
1.8.2. Практические разработки	95
1.9. О некоторых проблемах разработки и использования БИС/СБИС с программируемой структурой	100
1.9.1. Конвертация проектов	100
1.9.2. Конфигурирование БИС/СБИС с программируемой структурой	103
1.9.3. Засекречивание проектов	106
1.10. Способы оценки параметров ПЛИС. Обзор новой продукции ведущих фирм-производителей ПЛИС	109
1.10.1. Об оценке логической сложности (уровня интеграции) ПЛИС	111
1.10.2. Об оценке быстродействия ПЛИС	117
1.10.3. Сведения о современной продукции крупнейших фирм-производителей микросхем с программируемой структурой	119
Сведения, не включающие описаний блочных "систем на кристалле" с процессорными ядрами	120
Сведения о блочных "системах на кристалле", содержащих процессорные ядра	134
Глава 2. Основы технологии проектирования ИСПС и структуры систем автоматизированного проектирования	139
2.1. Общие сведения о процессе проектирования	141
2.1.1. Факторы, влияющие на методику проектирования электронных устройств	144
2.1.2. Области применения СпИС различных типов	144
2.1.3. Место БИС с программируемой структурой в процессе создания современной аппаратуры	146
2.2. Основы организации проектной процедуры для ИСПС	148
2.2.1. Проектирование цифровых фрагментов на заказных ИС и стандартных дискретных компонентах	157
Реализация в базе дискретных элементов типа МИС и СИС	159
Реализация цифровых фрагментов на ПЛИС	161
Реализация цифровых фрагментов в форме однокристалльного микроконтроллера	162
Реализация цифровых фрагментов в форме полужаказных кристаллов	162
2.2.2. Проектирование микропроцессорных фрагментов систем	173
Этап выбора типа МП	177
Этап выбора периферии	179
Этап разработки программного обеспечения	180
Средства поддержки создания программного обеспечения	182
Управление в реальном времени	183

Этапы кодирования и отладки программного обеспечения	186
Отладка аппаратуры	187
Методы комплексной аппаратно-программной отладки	188
2.2.3. Проектирование систем, связанных с обработкой аналоговых сигналов	195
Проектирование аналоговых фрагментов	196
Проектирование фрагментов со смешанным представлением сигналов	200
Проектирование аналоговых и аналого-цифровых SOPC	204
2.3. Структура и организация САПР	206
2.3.1. Связь процедуры проектирования и САПР БИС программируемой логики	206
2.3.2. Связь проектной проблемы с выбором САПР	207
2.4. Основные этапы проектирования БИС программируемой логики	213
2.4.1. Этап 1. Выбор элементной базы и САПР	213
2.4.2. Этап 2. Спецификация проекта	214
2.4.3. Этап 3. Разработка общей структуры проекта	215
2.4.4. Этап 4. Содержательное описание проекта и его частей	216
Иерархия языков проектирования дискретных устройств	217
Описание структуры операционного блока	221
Описание поведения элементов операционного блока	222
Описание работы устройства управления (УУ)	223
2.4.5. Этап 5. Компиляция проекта	224
2.4.6. Этап 6. Верификация проекта	226
2.4.7. Этап 7. Определение временных характеристик разработанного устройства	227
2.4.8. Этап 8. Организация натуральных экспериментов	227
2.4.9. Этап 9. Подготовка к производственному выпуску	228
2.5. Проблемы и методы проектирования SOPC	229
2.5.1. Специфические технико-технологические особенности реализации систем типа SOPC	231
2.5.2. Идеи и методы сопряженного проектирования	232
Методология проектирования	237
Систематическое повторное использование	238
Современные проектные средства	239
Эффективность стандартов	240
2.5.3. Стиль сопряженного проектирования и язык описания проекта	240
Методы, устраняющие связь сложности систем и скорости моделирования	243
2.5.4. Пакеты и САПР, поддерживающие проектирование SOPC	246
Фирма Mentor Graphics	249
Фирма Cadence	250
Проектный поток в SystemC	252
2.5.5. Понятие платформенно-базированных проектов SOPC	258
2.6. Некоторые аспекты технологии производства систем с ПЛИС	261
2.6.1. JTAG-интерфейс	264
Предпосылки возникновения	264
JTAG-интерфейс и метод граничного сканирования	266
Транспортный механизм JTAG-интерфейса	269
Команды граничного сканирования	273

Устройство управления JTAG-интерфейса	275
Стандарты, связанные с JTAG-интерфейсом	277
2.6.2. Проблемы и методология создания пригодной для тестирования аппаратуры	278
Работы, выполняемые проектировщиком при создании проекта	280
Работы, выполняемые при тестировании готовой продукции	281
2.6.3. Обзор средств поддержки JTAG-интерфейса ведущими фирмами	285
Аппаратная поддержка JTAG-интерфейса	286
Программная поддержка тестирования	288
Комплексная организация поддержки тестирования	289
Другие подходы	291
2.6.4. Системные функции на основе JTAG-интерфейса	292
Конфигурирование БИС ПЛИС	292
Развитие средств конфигурирования схем ИСПС	293
Встраивание отладочных приборов и инструментов в ПЛИС	293
Глава 3. Языки описания дискретных устройств	297
3.1. Общие положения	297
3.1.1. HDL-программа как модель проектируемого устройства	297
Типы данных	298
Структура и поведение	299
Стили описания проектов	303
3.1.2. Принципы интерпретации поведения дискретных устройств средствами моделирования	306
Моделирование и реальное время	306
Алфавит моделирования	310
3.1.3. Соглашение о правилах записи программ	312
3.2. Основы языка VHDL	314
3.2.1. Язык VHDL как программная система	314
3.2.2. Структура проекта. ENTITY и архитектурные тела	318
3.2.3. Типы данных	324
Предопределенные типы данных	326
Скалярные типы, вводимые пользователем	328
Физические типы	330
Массивы и записи	330
Подтипы	333
3.2.4. Сигналы и переменные. Оператор <i>PROCESS</i>	334
3.2.5. Атрибуты в языке VHDL	339
3.2.6. Последовательные операторы	343
Операторы присваивания	344
Оператор условия и оператор выбора	347
Оператор ожидания	352
Операторы повторения	353
Оператор проверки	358
3.2.7. Параллельные операторы	359
Оператор блока	362
3.2.8. Описание в VHDL типовых дискретных устройств	364
Комбинационные логические схемы	364
Описание триггеров и регистровых схем	372

Простые арифметические узлы	379
Описание цифровых автоматов	385
3.2.9. Подпрограммы	394
3.2.10. Разрешаемые сигналы и шины	399
3.2.11. Структурное представление проекта	404
3.2.12. Настройка и конфигурирование компонентов	409
3.2.13. Пакеты в VHDL. Концепция видимости описаний	414
Концепция видимости объектов в VHDL	417
3.3. Элементы языка Verilog HDL	419
3.3.1. Предварительные замечания	419
3.3.2. Типы данных	423
Цепи и регистры	423
Правила записи векторных констант	426
Память	427
Целые и действительные типы данных. Время	428
Строки	429
Параметры	430
3.3.3. Операции и выражения	430
Арифметические операции	432
Операции отношения и сравнения	432
Операции сдвига	433
Логические и поразрядные операции	433
Операции свертки	433
Условная операция	434
3.3.4. Операторы <i>initial</i> и <i>always</i>	434
3.3.5. Операция присваивания, операторы присваивания	436
Непрерывное присваивание	437
Последовательные присваивания	442
3.3.6. Операторы принятия решений	444
3.3.7. Операторы повторения	447
3.3.8. Инициализация процедурных операторов	449
3.3.9. Блоки	453
3.3.10. Подпрограммы	455
3.3.11. Структурное описание проекта	458
3.3.12. Примитивы	462
Предопределенные примитивы	462
Примитивы, определяемые пользователем	463
3.4. Язык AHDL	467
3.4.1. Общая характеристика языка и структура программы	467
3.4.2. Типы данных и выражения. Оператор присваивания	470
3.4.3. Оператор выбора и оператор условия	477
3.4.4. Таблицы в AHDL	479
3.4.5. Оператор повторения	480
3.4.6. Описание регистровых схем	481
3.4.7. Монтажная логика и буферные примитивы	485
3.4.8. Цифровые автоматы	490
3.4.9. Иерархическое проектирование в AHDL	495
3.5. Что дальше?	506

Глава 4. Примеры проектирования устройств с применением ПЛИС	513
4.1. Проектирование операционных устройств.....	513
4.1.1. Операционные устройства с микропрограммным управлением	513
Составление содержательной граф-схемы алгоритма	
и разработка структуры операционного блока	514
Разработка модели поведения цифрового автомата	520
Разработка текстового описания устройства	521
4.1.2. Операционные устройства конвейерного типа	524
4.2. Реализация модулей памяти в ПЛИС	529
4.2.1. Память с адресным доступом	530
4.2.2. Память с последовательным доступом	537
4.2.3. Память с ассоциативным доступом	541
4.3. Цифровые фильтры	546
4.4. Пример автоматизированного проектирования	
аппаратно-программной системы	560
4.4.1. Рассмотрение технического задания на разрабатываемое устройство	
и выбор элементной базы	563
Процедура декомпозиции проекта	569
Выбор САПР	569
4.4.2. Разработка аппаратной части БИС SOPC	572
Этап 1. ТЗ на проектирование аппаратной части БИС	572
Этап 2. Разработка общей структуры аппаратной части проекта	573
Этап 3. Проектирование основных компонентов блока <i>Cnt_ADC</i> —	
блока управления записью данных от АЦП	575
Этап 4. Проектирование блока <i>Cnt_Cmp</i>	586
Этап 5. Проектирование общей схемы — сборка проекта	
из отдельных фрагментов	586
4.4.3. Настройка предопределенных ресурсов кристалла	589
Создание заголовочного файла	590
4.4.4. Разработка программного обеспечения	590
Компиляция и создание объектного кода	591
Кодовая симуляция и отладка	591
4.4.5. Монтирование ресурсов SOPC в кристалл и комплексная отладка	
проекта	592
Загрузка проекта	592
Натурная отладка проекта	592
4.4.6. Разработка конструкции устройства	594
Список литературы	597
Предметный указатель	601

Предисловие

Интересы России требуют преодоления сложившегося в последнее десятилетие однобокого развития экономики с акцентом на сырьевые отрасли производства. Неотъемлемой чертой развитых государств является способность к разработке и производству изделий высокой технологии, среди которых одно из ведущих мест занимает электронная аппаратура. Успех электроники в значительной мере определяет технологический прогресс многих отраслей промышленности. В последнее время, после длительного застоя в России намечилось некоторое оживление как исследований, так и практических разработок в области систем обработки информации самого разного назначения, в том числе цифровых устройств.

В связи с ростом интереса к проблемам разработки электронной аппаратуры растет и потребность в соответствующей учебной и производственно-справочной литературе. Предлагаемая книга посвящена проблемам проектирования аппаратуры на современной элементной базе, в первую очередь, на микросхемах с программируемыми структурами. Наиболее развитыми и широко применяемыми в этой области являются цифровые программируемые микросхемы (ПЛИС) и, естественно, основное внимание в книге уделено именно этим компонентам. Но и состояние разработок, и методологию проектирования аналоговых (ПАИС) и цифроаналоговых программируемых микросхем авторы в некоторой мере попытались представить.

В настоящее время опубликованных работ на русском языке по предлагаемой тематике явно недостаточно. Среди книг, вышедших в последние годы, можно указать работы [3, 8, 24, 30, 56, 57].

Книга А. П. Антонова [3] — это удачное пособие для освоения основных проектных процедур при использовании САПР MAX+PLUS II фирмы Altera. В книге В. Б. Стешенко [24] подробно рассмотрены вопросы, связанные с реализацией задач цифровой обработки сигналов на компонентах программируемой логики, но также с преимущественной ориентацией на продукцию фирмы Altera. Имеется краткая информация о процедуре проектирования и языках проектирования дискретных устройств.

Ряд работ посвящены описанию языков проектирования аппаратуры (П. Н. Бибило [8], С. Емец [13], переводные книги [6, 56 и 57]). Не умаляя достоинств этих работ, хочется заметить, что они, в основном, ограничиваются изложением языка как такового, без достаточной связи с возможностями САПР и анализа влияния формы описания на результаты проектирования.

Книги В. В. Соловьева и А. А. Шалыто посвящены теоретическим вопросам синтеза устройств в элементном базисе, характерном для ПЛИС, но, как нам кажется, рассчитаны в большей мере на разработчиков систем проектирования, нежели на проектировщиков цифровых устройств. Вопросы проектирования устройств и систем с использованием распространенных САПР в них практически не отражены.

Авторы предлагаемой книги старались показать проблемы проектирования в широком плане, не сосредотачиваясь на результатах работы какой-либо отдельной фирмы, но и не жертвуя конкретным характером материалов. Авторы стремились нарисовать интегральную картину создания проекта с использованием ПЛИС, начиная от выбора способа реализации отдельных частей проекта (для этого, в частности, служит обзорный материал по современным программируемым микросхемам) и определения стратегии проектирования до завершающего представления проекта в форме, понятной системе автоматизированного проектирования, его верификацией и подготовкой к реализации.

Авторы являются сотрудниками кафедры вычислительной техники Санкт-Петербургского электротехнического университета (ЛЭТИ). Предлагаемый материал широко использовался в учебном процессе в курсах лекций по дисциплинам "Элементы и устройства ЭВМ", "Языки автоматизации проектирования средств вычислительной техники", "Автоматизация проектирования ЭВМ" для студентов, обучающихся по специальности 220100 "Электронные вычислительные машины, комплексы, системы и сети", а также при проведении курсов повышения квалификации работников научных и промышленных предприятий. Университет располагает методическими материалами и комплексом средств для выполнения лабораторных занятий и практических работ.

При участии авторов реализован ряд конкретных проектов с использованием ПЛИС из областей цифровой обработки сигналов, интерфейсных модулей компьютеров, кодирования информации.

Материалы книги распределены между авторами следующим образом: введение написано авторами совместно. Глава 1 написана Е. П. Угрюмовым, глава 2 — Р. И. Грушвицким, глава 3 — А. Х. Мурсаевым, глава 4 — А. Х. Мурсаевым и Р. И. Грушвицким. Такое разделение содержания между авторами лишь приблизительно отражает вклад каждого из них в создание книги, т. к.

при этом имели место взаимный обмен материалами и коллективное обсуждение; возникали ситуации, когда над отдельными разделами авторы работали совместно.

Книга может быть полезна студентам, обучающимся в областях вычислительной техники, автоматики, измерительных систем, радиотехники и телекоммуникаций, а также работникам научных учреждений и промышленных предприятий, специализирующихся в этих областях.

Авторы будут благодарны читателям за замечания по содержанию и характеру изложения книги.

Введение

Что такое "микросхемы с программируемой структурой?" Почему они появились? Как с ними обращаться и где применять? Это можно понять полнее, рассматривая возникновение микросхем с программируемой структурой на фоне предшествующих этапов развития вычислительной техники. Вычислительные машины второй половины XX века ведут свой отсчет от универсального компьютера фон Неймана, в котором процесс решения задачи организован как последовательное во времени выполнение простых операций. Каждая операция выполняется по своей команде, совокупность команд образует программу, разработанную для решения данной задачи. Состав аппаратных средств фиксирован (процессор, память, устройства ввода/вывода данных), и изменение решаемой задачи отражается лишь на числе и составе команд программы. Такой процесс решения соответствует *программной интерпретации алгоритмов*. Хотя в современных вычислительных архитектурах реализованы принципы одновременной (параллельной) реализации нескольких команд, в том числе конвейерные и суперскалярные принципы обработки, параллельная реализация нескольких программ в многопроцессорных системах и т. п., в целом в подобных системах сохраняется принцип последовательного во времени исполнения алгоритма.

Программный метод интерпретации алгоритмов не является единственным способом решения задач. Можно применять *структурную* или, как еще говорят, *аппаратную интерпретацию алгоритма*. В этом случае также выполняется множество отдельных простых операций. Но для выполнения отдельных действий применяются свои функциональные блоки, соединяемые в цепь определенной структуры. Функциональные характеристики блоков и характер соединений между ними соответствуют алгоритму решения задачи. Иными словами, для получения решения создается структура, отображающая интерпретируемый алгоритм. Алгоритм реализуется за счет продвижения данных и их преобразования по путям обработки (асинхронно или при тактировании) от входов к выходам схемы, с которых снимается результат. Команды для выполнения отдельных операций отсутствуют. Это обеспечи-

вает распределение решения задачи не только во времени, а и в пространстве. Естественным образом достигается *высокая степень параллелизма*. В этом случае, состав аппаратных средств тесно связан с решаемой задачей. Усложнение задачи ведет к увеличению количества используемого оборудования. *Изменение задачи требует изменения состава компонентов и способов их соединения.*

Роль и значение компьютеров в современной жизни общеизвестны. *Аппаратные методы решения задач также важны и в ряде случаев незаменимы.* Многие приложения, такие как моделирование динамических объектов, управление движением (особенно управление движением совокупности объектов), обработка сигналов и изображений в реальном времени, реалистичная компьютерная графика и подобные, связаны с многократным выполнением сравнительно простых операций, причем требуют выполнения до нескольких миллиардов операций в секунду.

Последовательное выполнение большого числа элементарных шагов при компьютерном решении задачи занимает относительно большое время. Кроме того, независимость аппаратных средств от сложности решаемой задачи имеет и оборотную сторону — даже для простейших задач нужны блоки, образующие компьютер в целом. Таким образом, простые задачи и задачи, которые должны решаться в реальном масштабе времени, могут привести к необходимости применения аппаратных вариантов решения. Понятие "реального времени" означает, что *результат должен быть получен за ограниченный интервал времени*, иначе он становится бесполезным.

Остановимся подробнее на некоторых путях реализации задач аппаратными средствами. Для создания аппаратных средств решения задач нужен набор различных элементов, узлов и устройств. Такой набор можно реализовать на микросхемах разного уровня интеграции: малого (МИС), среднего (СИС), большого (БИС) и сверхбольшого (СБИС). *Применение БИС/СБИС и исключение из схемы МИС и СИС существенно улучшает параметры аппаратуры* (стоимость, быстродействие, надежность, экономичность по потребляемой мощности и др.). В то же время на пути реализации схем в виде БИС/СБИС могут возникнуть большие экономические трудности, поскольку разработка БИС/СБИС чрезвычайно дорога и требует больших затрат времени (стоимость проектирования достигает сотен миллионов долларов, а время разработки — многих месяцев). Такие затраты могут оправдаться только при больших тиражах выпуска микросхем. Тогда высокая стоимость их проектирования раскладывается на большое число изготовленных и проданных экземпляров (порядка сотен тысяч или более), чего при проектировании специализированных вычислительных средств обычно не бывает.

Противоречие между желательностью и возможностями применения БИС/СБИС в специализированных устройствах и системах вызвало к жизни *концепцию программирования структур.*

Благодаря программированию структур для электронной промышленности осуществлялась *универсализация БИС/СБИС*, сделавшая их пригодными для широкого круга потребителей. Это было достигнуто изготовлением промышленным способом некоторых заготовок, которые далее преобразуются потребителем в нужные ему устройства. Заготовки можно производить массовым тиражом, что позволяет выполнять их в виде БИС/СБИС. Естественно, необходима простота доведения заготовок до законченных микросхем, т. е. простота программирования структур БИС/СБИС по спецификациям заказчика. В этом случае *применение БИС/СБИС даже для малотиражных изделий становится возможным и эффективным.*

Таким образом, программируемость БИС/СБИС, иными словами, создание программируемых логических интегральных схем, называемых ПЛИС, позволила использовать их там, где ранее приходилось применять МИС/СИС. Более позднее достижение — *репрограммируемость* микросхем, обеспечившая возможность многократной смены их настройки. Это позволяет изменять функционирование кристалла, т. е. на одном и том же кристалле получать устройства различного назначения, стирая старую конфигурацию его структуры и записывая новую. Для некоторых разновидностей СБИС программируемой логики возможна реконфигурация в оперативном режиме, т. е. без извлечения микросхемы из работающей системы и с высокой скоростью. *Оперативное программирование микросхем — путь для новых принципов построения аппаратуры.*

Из изложенного видно, что программирование — концепция, сыгравшая большую роль в развитии вычислительной техники. В этой концепции можно выделить два аспекта: программирование задач для процессоров с последовательной интерпретацией алгоритма и программирование структур для процессоров (используем здесь тот же термин) с параллельной интерпретацией алгоритма.

Между двумя аспектами концепции программирования отнюдь не проходит какая-то разделяющая их граница. Они могут сочетаться, открывая при этом новые горизонты в области создания высокоэффективных средств обработки данных. Более того, уже в начале XXI века можно ожидать появления гибридных вариантов таких средств (так называемых реконфигурируемых компьютеров и динамически реконфигурируемых вентильных матриц). В этих архитектурах оба аспекта программирования в какой-то мере сочетаются.

Уже в первых классических ЭВМ с последовательной интерпретацией алгоритмов встречается, хотя и в малой степени, и программирование структур, т. к. выполнение различных операций в разных машинных тактах означает и некоторую перекоммутацию цепей внутри ЭВМ. В прогнозируемых архитектурах *реконфигурируемых компьютеров* вместо несложных устройств, реализующих элементарные операции, соответствующие простым командам, предполагается использовать так называемые *обрабатывающие поля* заданной

размерности, конфигурируемые для выполнения фрагментов алгоритма оптимальным образом с сокращением времени решения и необходимых для него аппаратных ресурсов. Фрагменты алгоритма выполняются последовательно во времени, соответственно им "перестраиваются" и обрабатывающие поля. Теоретический анализ показывает, что в подобных архитектурах быстроедействие ЭВМ может быть увеличено на порядки.

Со стороны разработчиков ПЛИС ставится вопрос построения FPGA-процессоров (FPGA, Field Programmable Gate Array, т. е. вентильная матрица, программируемая пользователем). В таких процессорах применяется динамически репрограммируемая FPGA с быстрой сменой настроек. Алгоритм работы процессора загружается в FPGA аналогично загрузке в память ЭВМ выполняемой программы. После решения части задачи полученные промежуточные данные сохраняются в специально вводимых в схему элементах памяти (триггерах), для решения следующей части (фрагмента алгоритма) эти данные используются в качестве исходных. Таким образом, с помощью нескольких последовательных смен настроек реализуются функции устройства в целом на аппаратных средствах, сложность которых соответствует лишь части устройства (одному контексту, т. е. схеме, для выполнения функций лишь одного яруса разбиения устройства).

Наиболее широкое применение ПЛИС находят при построении *периферийных модулей компьютеров*, в том числе:

- ☐ интерфейсных средств;
- ☐ средств сопряжения с аппаратурой управляющих комплексов и систем;
- ☐ средств защиты информации;
- ☐ аппаратуры поддержки телекоммуникаций;
- ☐ аппаратных ускорителей (например, для реалистичной графики, обработки изображений, обработки сигналов).

Кроме того, ПЛИС используют для создания высокопроизводительной аппаратуры специализированного назначения, включая:

- ☐ обработку сигналов в реальном времени;
- ☐ средства управления производственными процессами;
- ☐ автономные интеллектуальные датчики;
- ☐ средства кодирования и декодирования информации в системах связи.

Прокомментируем некоторые из отмеченных направлений.

Ядро вычислительной системы составляют процессоры и память, которые с точки зрения производителей ИС являются стандартной продукцией. Кроме стандартных частей вычислительная система содержит и некоторые индивидуальные схемотехнические части для сопряжения модулей и управления

ими. Такие индивидуальные части системы ранее приходилось строить с помощью ИС малого и среднего уровней интеграции, применение которых ведет к резкому увеличению числа корпусов и внешнего монтажа. Применение ПЛИС — кардинальный путь реализации систем в целом на интегральных схемах высокого уровня интеграции. Одно из последних достижений в этой области — системы на кристалле SOC (System-On-a-Chip), интегрирующие процессорный модуль, программируемый "традиционным" способом, т. е. путем занесения программы в постоянную или оперативную память, и реконфигурируемую вентильную матрицу для реализации требуемых в конкретном проекте специфических прикладных аппаратных средств.

Говоря о системах управления, следует отметить, что они, как правило, содержат большое количество разнообразных датчиков и исполнительных устройств, отличающихся способами представления информации, форматами данных, временными характеристиками потоков данных. Узлы системы часто работают не синхронно с вычислительным ядром. Решение задач сопряжения принципиально требует использования специальных аппаратных средств для буферизации, упаковки и распаковки данных, взаимного оповещения узлов системы о тех или иных событиях. Устройства преобразования формы представления информации (аналого-цифровые и цифроаналоговые преобразователи) также реализуются с использованием структурных подходов и часто включают как цифровые специализированные узлы, так и аналоговые подсистемы. Необходимость включения в системы аналоговых компонентов, перестраиваемых в соответствии с требованиями потребителей, вызвала в последние годы разработку и производство программируемых аналоговых и цифроаналоговых интегральных схем (ПАИС и ПЛАИС).

Разнообразны возможности применения специализированных (но зачастую перестраиваемых) аппаратных средств для защиты информации. Это могут быть простейшие средства контроля доступа к информации на жестких магнитных и лазерных дисках — так называемые индивидуальные электронные ключи доступа, которые персонально программируются для допущенных пользователей и уносятся ими после сеанса работы. В более ответственных случаях применяется шифрация информации перед записью на диск или передачей в сеть. Преимущество аппаратных средств защиты по сравнению с программными наряду с повышением скорости работы состоит в трудности и даже невозможности скопировать алгоритм шифрации.

Реализация коммуникационных протоколов нижних уровней (физического и канального) традиционно решалась с использованием аппаратных средств для формирования битовых потоков и их первичного декодирования, установления соединений абонентов. Применение программируемой логики делает возможным и экономически эффективным выполнение в аппаратной среде интеллектуальной обработки, включая сжатие, буферизацию, пакетирование данных и т. д.

Аппаратные ускорители вычислений также не являются чем-то существенно новым. Видеокарты, блоки для реализации задач компьютерной графики и обработки изображений занимают заметное место на компьютерном рынке. Программируемая логика и, особенно, репрограммируемые схемы обеспечивают новые возможности. Во-первых, покупая плату, содержащую одну или несколько ПЛИС (а такие платы производят многие фирмы), и подключая ее к компьютеру стандартным образом, пользователь получает в свое распоряжение как бы несколько различных устройств. Достаточно загрузить в микросхему одну из возможных конфигураций из числа сохраняемых на жестком диске компьютера, чтобы вызвать к исполнению тот или иной алгоритм функционирования периферийного блока, подобно тому, как в "обычной" ЭВМ запускаются различные программы. Во-вторых, пользователь может самостоятельно модифицировать работу такого блока, обеспечивая реализацию специфических функций, не предусмотренных изготовителями специализированных устройств.

В связи с этим нельзя не отметить использование ПЛИС при разработке полностью заказных БИС. На первой стадии разработки проект воспроизводят с использованием репрограммируемых компонентов, в которые легко вносить изменения. Затем отлаженная конфигурация импортируется в систему проектирования заказных БИС. Такая методика называется быстрой *прототипизацией систем* (Rapid System Prototyping), а платы с набором микросхем ПЛИС и средствами подключения к отладочному компьютеру — *платами для прототипизации* (prototyping boards, development boards).

Приведенные примеры далеко не исчерпывают всех возможностей. Более того, в связи с постоянным уровнем роста степени интеграции программируемых БИС снижением их стоимости, развитием средств проектирования область их применения постоянно расширяется.

Разработка систем с использованием программируемых БИС невозможна без применения средств и систем автоматизированного проектирования (САПР). Трудно даже представить разработку "ручными" способами устройств с уровнем сложности, эквивалентным сотням тысяч вентилях. Рост уровня сложности требует не только использования современных средств проектирования, но и переосмысления всей технологии проведения проектных работ, включая декомпозицию проекта, планирование и управление работами, параллельную и взаимосвязанную разработку его фрагментов и грамотное объединение фрагментов в целостное устройство. Особо значимыми становятся процедуры отладки и верификации проектных решений. Рынок САПР больших интегральных схем, и в частности программируемых ИС, переживает период бурного развития. В конкурентной борьбе участвуют как фирмы-производители БИС, так и ряд фирм, специализирующихся на разработке систем автоматизированного проектирования.

Современной тенденцией развития САПР является интегрирование в одной системе средств для выполнения большинства этапов проектирования, на-

чиная от описания проекта, его моделирования и отладки, вплоть до разработки проектной документации и управления процессом физической реализации. Много внимания уделяется средствам повторного использования проектных решений (Reusing), организации корпоративной проектной деятельности (библиотеки, встроенные системы управления базами данных).

Тем не менее, нередко ситуации, когда по тем или иным причинам части проекта подготавливаются в разных проектных средах. Это может быть связано с привычкой отдельных членов коллектива проектировщиков к различным формам представления проектной информации, желанием приобрести от третьих фирм готовые решения, которые исходно создавались на других платформах, не говоря о том, что к частям проекта могут предъявляться специфические требования, требующие привлечения специальных средств синтеза и отладки. Нельзя не признать также оправданной использование различных средств описания на разных уровнях представления проекта. Так, для общей спецификации можно рекомендовать представления на языках системного уровня. Высшие уровни структурной декомпозиции удачно и наглядно представляются в графической форме. Для детального представления фрагментов могут использоваться языки описания аппаратуры или схемное представление (Net List). Разработчики САПР должны учитывать подобные потребности.

Поэтому другая тенденция разработок — это обеспечение совместимости различных форм представления в одном проекте, включая возможность импортирования в рабочий проект фрагментов, созданных на разных платформах различными средствами. Несмотря на имеющуюся достаточно острую конкуренцию, фирмы часто заключают соглашения на право взаимного использования программной продукции, а также передачи информации о структуре конфигурационных файлов для программирования микросхем.

Важнейшее влияние на весь процесс проектирования оказывает выбор исходного описания проекта. Традиционным способом представления цифровых устройств является графическое. Определяется набор компонентов, изображения которых размещаются на поле чертежа (в рабочем окне дисплея), и прорисовываются их соединения. Близко к графическому примыкает табличное представление, предусматривающее явное задание списка компонентов и списка соединений. Главным недостатком графического и табличного представления проекта следует считать трудоемкость процедуры ввода, трудность поиска ошибок. Представление проекта и ввод информации в форме текста на языке проектирования аппаратуры (HDL, Hardware Design Language) в значительной мере свободны от этих недостатков. Текстовое описание объединяет в себе возможность компактного представления очень сложных логических проектов с легкостью его понимания и большой скоростью его создания, включая относительную простоту поиска ошибок и внесения модификаций в проект. Достоинством описания проекта в тексто-

вом виде (на языках Verilog, VHDL и др.) является также возможность легкого переноса проекта в различные приборные среды, простота настройки или перенастройки параметров разрабатываемых устройств или их фрагментов, например, разрядности или подмножества реализуемых функций. У разработчика появляются предпосылки расширять возможности языка за счет введения собственных макроопределений, ориентированных на класс реализуемых проектов.

Кроме того, языковое описание является эффективным способом перевода программ пользователя, построенных на базе стандартных универсальных языков (например, С), в языки описания аппаратуры и обратно. Уже сегодня имеются возможности автоматического генерирования HDL-программ на базе других формальных представлений. Развитие языковых средств проектирования сближает методы проектирования аппаратуры и программных средств, расширяя возможности разработчиков, и улучшает взаимопонимание между проектировщиками различных подсистем вычислительных комплексов.

Нельзя не отметить значительное отставание российских проектных организаций от их западных конкурентов не только и не столько в количестве привлекаемых систем проектирования, сколько в готовности персонала использовать современные технологии проектирования. В частности, до сих пор, в отличие от мировой ситуации, в России подавляющее большинство проектов в области цифровой техники реализуется с использованием схемного (графического) ввода проекта. Авторы надеются, что издание этой книги будет способствовать подготовке специалистов, владеющих современными средствами проектирования, и улучшению ситуации.

Структура предлагаемой книги соответствует перечисленным особенностям создания проекта.

Глава 1 представляет обзор рынка программируемых БИС. Авторы не стремились детализировать архитектурные и схемотехнические особенности (такие сведения можно найти в работах [4, 17, 24, 27, 31, 35, 45, 53] и др.), хотя в отношении новейших семейств программируемых ИС соответствующие данные приводятся. Больше внимания уделено сравнительной оценке. Содержание главы должно стать руководством для принятия решений о выборе технических средств на ранних этапах выполнения проекта.

Материал *главы 2* на сегодня не имеет аналогов в общедоступной литературе. Здесь предпринята попытка систематизированного представления не только отдельных проектных процедур применительно к проектированию систем на программируемых микросхемах, но, прежде всего, выявления их взаимосвязи и взаимной обусловленности. Отмечаются отличия современных технологий проектирования от концепций, принимавшихся ранее. Подчеркивается возможность и целесообразность при использовании современ-

ных САПР встраивать в проект разнородные компоненты и использовать разнообразные формы представления.

Глава 3 посвящена изложению наиболее употребительных языков проектирования дискретных устройств. Объединение в одном издании изложения нескольких языков (что не соответствует распространенной практике подготовки индивидуальных изданий для различных языков) позволило рассмотреть их с единых позиций, выделить их общие свойства, но и оттенить различия. Это упростит при необходимости переход проектировщиков на новые языки (в том числе и еще только разрабатываемые) и улучшит взаимопонимание проектных групп, ориентирующихся на разные способы представления.

В *главе 4* представлено несколько конкретных разработок модулей на ПЛИС. На этих примерах иллюстрируется методология проектирования и особенности выбора реализации в зависимости от системных требований. Кроме того, авторы старались показать читателю, что "проектирование на ПЛИС — это очень просто".

ГЛАВА 1

Элементная база электронных устройств и систем. Архитектура и схемотехника интегральных схем с программируемой структурой

1.1. Общие сведения. Предшественники. Классификация

1.1.1. Вводные замечания. Историческая справка

Элементную базу электронной аппаратуры обработки информации и ее хранения составляют интегральные схемы (ИС). В зависимости от характера сигналов, отображающих информацию, средства ее обработки разделяют на цифровые, аналоговые и цифроаналоговые. Классификация цифровых ИС приведена на рис. 1.1.

Прежде всего, по признаку ориентации на массовое потребление или на конкретный заказ цифровые интегральные схемы можно разделить на *стандартные* и *специализированные*. Стандартные ИС приобретаются потребителем как готовые изделия (off-the-shelf) и производятся массовыми тиражами, что позволяет затрачивать большие средства на их проектирование, поскольку его стоимость раскладывается на большое число изделий. Стандартные ИС традиционных видов имеют практически жесткую внутреннюю структуру, и потребитель не может влиять на характер их функционирования. Специализированные ИС (СПИС), имеющие индивидуальный характер функционирования, приходится в той или иной мере разрабатывать (проектировать) по конкретному заказу. Проектирование ИС — процесс сложный и дорогостоящий, поэтому понятно стремление в максимально возможной степени строить аппаратуру на основе стандартных ИС.

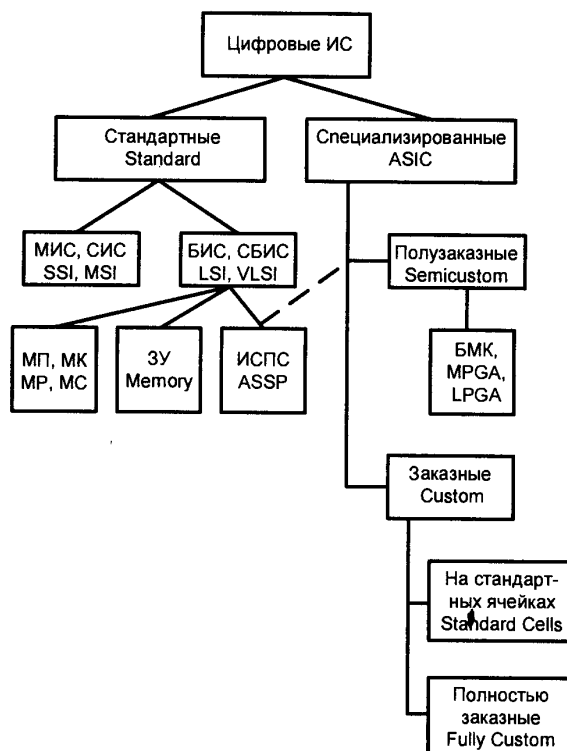


Рис. 1.1. Классификация цифровых интегральных схем

К стандартным относятся следующие виды цифровых ИС.

- Микросхемы малого и среднего уровней интеграции МИС и СИС, в английской терминологии SSI (Small Scale Integration) и MSI (Medium Scale Integration). Этот класс микросхем представлен многочисленными и хорошо известными с давних пор стандартными сериями элементов. Необходимость использования таких микросхем в отдельных ситуациях сохраняется и сейчас, но построение на них сложных устройств ведет к существенному ухудшению технико-экономических параметров аппаратуры в сравнении с теми, которые могли бы быть получены при применении микросхем высокого уровня интеграции.
- БИС/СБИС микропроцессоров и микроконтроллеров, МП и МК (Microprocessors, MP и Microcontrollers, MC). Многочисленные микросхемы МП и МК широко и успешно применяются при решении задач программной реализации алгоритмов.

- БИС/СБИС запоминающих устройств (Memory IC), относящиеся к наиболее массовым ИС с широкой номенклатурой, применяемым в устройствах и системах разного назначения для хранения данных и обмена ими с другими блоками систем.
- Интегральные схемы с программируемой пользователем структурой (ИСПС). Эти схемы ознаменовали появление нового перспективного направления в развитии элементной базы электронного приборостроения. Именно эти БИС/СБИС открывают возможности использования микросхем высокого уровня интеграции не только в массовых изделиях, выпускаемых тиражами в сотни тысяч экземпляров, но и в проектах практически любой тиражности. В ИСПС задание схеме определенной внутренней структуры (конфигурирование схемы) выполняется потребителем, изготовитель в этом процессе не участвует и поставляет на рынок продукцию, пригодную для многих покупателей, создающих аппаратуру разного назначения. Рассмотрению ИСПС посвящены основные материалы этой главы. Отметим, что ИСПС имеют в известной мере двойственный характер: для потребителя они являются специализируемыми, а для промышленности — стандартными (со всеми вытекающими из этого выгодами). Это обстоятельство отражено на рис. 1.1 штриховой линией.

Кроме указанных в классификации, к стандартным микросхемам принадлежат также аналого-цифровые и цифроаналоговые преобразователи.

Специализированные ИС, как уже отмечалось, в той или иной мере проектируются для конкретного заказчика. При этом от заказчика, как правило, требуются затраты значительных средств при длительных сроках разработки СпИС. Для преодоления трудностей создания специализированных ИС был предложен ряд методик их проектирования, соответственно которым существует и несколько разновидностей СпИС (в английской терминологии СпИС обозначаются термином ASIC, Application Specific Integrated Circuit).

Замечание

Особо следует остановиться на термине ASSP (Application Specific Standard Products), который получает все более широкое распространение. Этому термину не всегда придается одинаковый смысл. В данной книге термин ASSP понимается как синоним термина ИСПС и определяет способ конфигурирования схемы под определенную задачу. Для программирования ASIC нужна разработка специальных фотошаблонов с рисунками межсоединений, требующая больших затрат средств и времени. Применяя для целей конфигурирования программируемые логические интегральные схемы (ПЛИС), можно исключить разработку шаблонов и сравнительно простыми способами получить эффективные средства построения цифровых систем, содержащие набор схем с необходимым функционированием. Такие архитектуры и относятся к классу ASSP, причем первая часть аббревиатуры сохраняет преемственность с термином ASIC, а вторая подчеркивает принадлежность микросхем к стандартной для электронной промышленности продукции.

Можно без преувеличения сказать, что вопросы реализации проектов на элементной базе высокого уровня интеграции сейчас привлекают к себе основное внимание разработчиков систем и специалистов электронной промышленности. В этом направлении важнейшим достижением явилось появление ИС с программируемой структурой. До их создания специализированные проекты в зависимости от конкретных условий (сложности проекта, его тиражности, требований к техническим и экономическим характеристикам, отведенного для реализации времени и т. д.) традиционно выполнялись в следующих вариантах:

- на микросхемах малого и среднего уровня интеграции;
- на полужаказных микросхемах (базовых матричных кристаллах);
- на заказных микросхемах, спроектированных по методу стандартных ячеек;
- на полностью заказных микросхемах.

Среди СПИС (ASIC) различают классы *полузаказных* и *заказных* ИС. Разновидностями заказных микросхем являются полностью заказные ИС (Fully Custom IC) и ИС, спроектированные методом "на стандартных ячейках" (Standard Cell Logic). *Полностью заказные схемы* целиком проектируются по требованиям конкретного заказчика. Проектировщик имеет полную свободу действий, определяя схему по своему усмотрению вплоть до уровня схемных компонентов (отдельных транзисторов и т. п.). Для изготовления схемы требуется разработка всего комплекта фотошаблонов, верификация и отладка всех схемных фрагментов. Такие схемы очень дороги и имеют длительные циклы проектирования.

Схемы на стандартных ячейках отличаются от полностью заказных тем, что их фрагменты берутся из заранее разработанной библиотеки схемных решений. Такие фрагменты уже хорошо отработаны, поэтому стоимость и длительность проектирования при их использовании снижаются. Для производства схем тоже требуется изготовление полного комплекта фотошаблонов, но разработка шаблонов облегчена. Потери сравнительно с полностью заказными ИС состоят в том, что проектировщик имеет меньше свободы в построении схемы, т. е. результаты ее оптимизации по критериям площади кристалла, быстродействию и т. д. менее эффективны. Наивысших технических параметров добиваются от полностью заказных схем, однако метод стандартных ячеек популярен, т. к. при небольших потерях в технических характеристиках с его помощью можно заметно упростить проектирование схемы. Полностью заказные схемы разрабатываются за время, превышающее время разработки схем методом стандартных ячеек приблизительно в два раза.

К *полузаказным схемам* относятся базовые матричные кристаллы БМК (в английской терминологии MPGA, Mask Programmable Gate Arrays, или LPGA, Laser Programmable (или Laser Personalised) Gate Arrays). В этом случае имеется стандартный полуфабрикат, который доводится до готового из-

делия с помощью индивидуальных межсоединений. Реализация требует изготовления лишь малого числа фотошаблонов (для MPGA) или цикла операций разрушения некоторых межсоединений лазерным лучом (для LPGA). Хотя стоимость и длительность проектирования в сравнении с полностью заказными схемами сокращаются в 3—4 раза, результат еще дальше от оптимального. Причина неоптимальности заключается в том, что в матричных БИС (МАБИС) менее рационально используется площадь кристалла (на кристалле остаются неиспользованные элементы и т. п.), длины связей не минимальны, и быстродействие не максимально.

Сходство методов проектирования на БМК и стандартных ячейках состоит в использовании библиотек функциональных элементов. Различие — в том, что для схем, проектируемых по методу стандартных ячеек, библиотечный набор элементов имеет более выраженную топологическую свободу. Например, в отличие от БМК, стандартизируются не оба размера ячеек (ширина и высота), а только высота ячеек, длины которых могут быть различными. При проектировании вначале из набора библиотечных элементов подбираются необходимые функциональные блоки, а затем решаются задачи их размещения и трассировки. Существует, естественно, и большое различие в числе шаблонов, которое требуется изготавливать для производства БМК и заказных ИС по методу стандартных ячеек.

Методика, а соответственно, и САПР для проектирования по методу стандартных ячеек более сложны, чем для проектирования на основе БМК, которому свойственны более жесткие топологические ограничения. Ограничения вводятся и для метода стандартных ячеек (постоянство высоты ячеек, предопределенность геометрических размеров и положения шин питания, тактирования и др.), но по мере применения более мощных САПР ограничения ослабляются. Длительность изготовления БИС/СБИС методом стандартных ячеек превышает этот же показатель для МАБИС на основе БМК в 1,3—1,8 раза.

Таким образом, как показано выше, *заказные БИС/СБИС* от начала до конца разрабатываются для конкретного проекта. В них нет лишних элементов, оптимально размещены подсхемы и их межсоединения. Свойственная заказным схемам оптимизация БИС/СБИС по конкретному техническому заданию минимизирует площадь кристалла, т. е. удешевляет его, позволяет добиться максимального быстродействия разрабатываемого устройства и т. д. Но, как уже отмечалось, разработка оригинальной БИС/СБИС (ее проектирование) чрезвычайно дорога и требует спроектировать и изготовить полный комплект фотошаблонов (15—20 штук). Стоимость проектирования и изготовления каждого шаблона составляет десятки тысяч долларов. Современный уровень затрат на создание комплекта фотошаблонов для производства БИС/СБИС можно охарактеризовать следующими цифрами: для технологии с минимальным размером 0,18 мкм это 300—350 тыс. долларов, для технологии с минимальным размером 0,13 мкм — около 500 тыс. долла-

ров, а для минимального размера 0,1 мкм ожидается 750 тыс. долларов или более.

Проекты, реализованные на ИС с программируемой пользователем структурой, стали успешно конкурировать прежде всего с двумя первыми из перечисленных выше методов реализации специализированных устройств. В настоящее время проектирование систем и логических устройств все больше перемещается в сторону широкого использования программируемых схем для всех вариантов разработки электронных аппаратных средств.

Интегральные схемы с программируемой пользователем структурой (ИСПС) существуют уже около 25 лет и к настоящему времени представлены множеством разнообразных семейств. Программирование структур вначале было применено в программируемых логических матрицах (ПЛМ), программируемой матричной логике (ПМЛ) и базовых матричных кристаллах (БМК). Вслед за ними возникли новые классы более сложных ИСПС, продолжающих линии развития матричной логики и базовых матричных кристаллов: CPLD и FPGA, соответственно. Затем были реализованы ИСПС комбинированной (смешанной) архитектуры, сочетавшие признаки CPLD и FPGA. Позднее удалось разработать ИСПС с аналоговыми и аналого-цифровыми элементами, которые можно обозначить как ПАИС (программируемые аналоговые интегральные схемы).

1.1.2. Базовые матричные кристаллы (вентильные матрицы)

В этой книге понятие "*программируемые*" относится к операциям, выполняемым пользователем, поэтому рассмотрение БМК, строго говоря, не соответствует теме. Действительно, класс БМК выпадает из общего ряда перечисленных ИСПС, поскольку для него программирование микросхем производится не пользователем, а изготовителем, т. е. на предприятии электронной промышленности с помощью сложных технологических процессов. Тем не менее, ознакомление с БМК необходимо, т. к. они являются предшественниками FPGA — важного класса ИСПС — имеют с ними ряд родственных черт и тесно связаны с использованием FPGA в процессе так называемой *конвертации* проектов. Кроме того, задача проектирования устройств и систем на основе БМК близка к задаче проектирования на ИСПС, поскольку, как правило, в значительной мере ложится на плечи пользователя.

Базовые матричные кристаллы называют также *вентильными матрицами* (ВМ), что соответствует принятому для БМК английскому термину GA (Gate Array). Термин БМК преобладает в отечественной литературе и поэтому используется в этой книге, хотя термин "вентильная матрица" был бы предпочтителен как не противоречащий международной терминологии.

Базовые матричные кристаллы известны с 1975 года, когда американская фирма Amdahl Corp применила их в составе высокопроизводительной ЭВМ

с целью исключения из нее микросхем малого и среднего уровней интеграции. Результаты оказались удачными, и в последующие годы БМК получили широкое распространение.

БМК — кристалл, на прямоугольной поверхности которого размещены *внутренняя и периферийная области* (ВО и ПО). Во внутренней области по строкам и столбцам (в виде матрицы) расположены *базовые ячейки* — группы нескоммутированных схемных элементов (транзисторов, резисторов). Элементный состав базовой ячейки при разных вариантах межсоединений элементов допускает реализацию некоторого множества схем определенного класса, каждая из которых соответствует определенной *функциональной ячейке* (ФЯ). Для выпускаемого в продажу БМК создается *библиотека функциональных ячеек*, т. е., в сущности, рисунков межсоединений, дающих ту или иную схему. Библиотеки функциональных ячеек БМК насчитывают обычно десятки или сотни типовых узлов, реализованных на одной или нескольких базовых ячейках.

В БМК *канальной структуры* между строками и столбцами базовых ячеек или их компактных групп оставляются горизонтальные и вертикальные свободные зоны (каналы) для межсоединений.

В периферийной области кристалла размещаются *периферийные ячейки*, выполняющие операции ввода/вывода сигналов через контактные площадки, расположенные по краям кристалла.

Переход от биполярной схемотехники к МОП-транзисторным схемам сделал возможным более плотное размещение схемных элементов на кристалле. При росте сложности логической части схемы, т. е. числа логических элементов на кристалле, еще быстрее растет сложность межсоединений. Для снижения потерь площади кристалла, затрачиваемой на трассировочные каналы, была разработана *бесканальная* структура БМК типа "*море вентиля*" или "*море транзисторов*", в которой конструкторы отказались от свободных зон кристалла, заранее отведенных для межсоединений. В таких БМК вся внутренняя область плотно заполняется базовыми ячейками, а межсоединения проводятся только там, где это необходимо, причем находящиеся под ними базовые ячейки оказываются в данной конкретной схеме неиспользуемыми.

Еще одной разновидностью БМК являются *блочные*, в которых на кристалле выделяются специализированные *области-подматрицы* для выполнения заранее определенных функций (логической обработки данных, памяти и др.). Между подматрицами размещаются специальные трассировочные каналы, на периферии подматриц изготавливаются внутренние периферийные ячейки для передачи сигналов по межблочным связям.

Изобретение базовых матричных кристаллов означало появление нового класса специализируемых БИС/СБИС — *полузаказных*.

Для реализации конкретной полузаказной схемы на основе БМК (такие схемы называют МАБИС, т. е. *матричные БИС*, или БИСМ, т. е. БИС мат-

ричные) нужны только шаблоны рисунков межсоединений. Например, при внедрении первых БМК в фирме Amdahl Corp для заказных схем требовалась разработка 13 шаблонов, а для полузаказных МАБИС только трех.

Заказные БИС/СБИС существенно дороже полузаказных в проектировании, но затраты на изготовление каждого экземпляра уже спроектированной схемы у них меньше. Затраты на проектирование (подготовку производства) однократны, затраты на изготовление присущи каждому экземпляру микросхемы. Отсюда видны экономически обоснованные области применения заказных и полузаказных БИС/СБИС. Для массового производства, когда стоимость подготовки производства (проектирования) раскладывается на большое число микросхем, в общей стоимости экземпляра микросхемы будет преобладать стоимость его изготовления и целесообразно заказное проектирование. Так проектируются микропроцессоры, микроконтроллеры, микросхемы памяти. Для специализированных средств с меньшими тиражами заказное проектирование чаще всего неприемлемо. В эпоху появления БМК альтернативным вариантом проектирования БИС/СБИС с тиражностью в десятки тысяч экземпляров стало полузаказное. Позднее возможности упрощения и удешевления процесса разработки БИС/СБИС многократно возросли в связи с появлением новых и эффективных вариантов ИС с программируемыми структурами.

БМК выпускаются рядом зарубежных фирм, а также отечественной фирмой "Ангстрем" (семейства 1806XM1, 1515XM1, 1593XM1, 1537XM1, 1592XM1) и белорусским объединением "Интеграл" (семейства 1532XM, 1540XM, 1547XM, 1574XM, 1575XM). Для ориентировки укажем некоторые параметры отдельных БМК. Предварительно заметим, что оценкой сложности БМК служит число эквивалентных вентилях (обычно, вентилях 2И-НЕ) в его составе.

БМК 1592XM1 содержит 100 тыс. эквивалентных вентилях, имеет библиотеку из 230 функциональных ячеек и рабочую частоту 50 МГц. Корпус БМК имеет 132 внешних вывода, из которых пользовательских (функциональных) 100. Диапазон рабочих температур от -60 до $+85$ °C.

БМК G10, G11, G12 американской фирмы LSI Logic имеют беспрецедентно высокую сложность и содержат до 33 млн. эквивалентных вентилях, а биполярный БМК SH100K фирмы Infinion (ранее Siemens Semiconductor), имеющий до 10 тыс. эквивалентных вентилях и матрицу аналоговых элементов на одном и том же кристалле, работает на тактовой частоте 5 ГГц.

В заключение отметим, что для более четкого отражения места БМК в иерархии микросхем полезно дополнить английскую аббревиатуру GA до аббревиатуры MPGA, Mask Programmable Gate Array.

Заметим также, что наряду с описанным выше вариантом БМК существуют и БМК с лазерным программированием межсоединений (LPGA, Laser Programmable Gate Arrays). В этих БМК первоначально изготавливаются металлизированными дорожками все возможные межсоединения элементов, а

при программировании часть соединений под воздействием точно сфокусированных управляемых лазерных лучей разрывается. Такие БМК подобны обычным в том, что для конфигурирования МАБИС потребитель должен обращаться к изготовителю, т. е. на предприятие электронной промышленности, однако сроки и стоимость выполнения заказа в сравнении с обычными БМК оказываются иными. Сроки выполнения заказа сокращаются, что благоприятно с точки зрения ускорения выхода продукции на рынок, однако в настоящее время БМК с лазерным программированием распространены меньше, чем обычные, хотя имеются фирмы, развивающие их производство, в частности фирма Chip Express.

1.1.3. Классификация логических микросхем программируемой логики

В приводимой ниже классификации термин "логических" обусловлен отсутствием в ней сведений об аналоговых и аналого-цифровых программируемых микросхемах, для которых классификационные признаки еще не вполне сложились. Под программируемостью здесь и ниже следует понимать только возможность программирования микросхемы пользователем. Аббревиатурой для общего обозначения всех классифицируемых микросхем может быть ПЛИС либо ЦИСПС, т. е. "программируемые логические интегральные схемы" либо "цифровые интегральные схемы с программируемой структурой". Первый вариант действительно известен и поэтому будет использован, хотя он и не совсем точен (например, применительно к широко распространенным схемам, в которые, кроме программируемой логики, входят также встроены блоки памяти).

В классификации для пояснения содержательных положений, естественно, использован русский язык. В то же время для многих аббревиатур сохранены английские варианты, поскольку они уже прочно закрепились в практике и не имеют широко признанных русских аналогов.

Микросхемы, программируемые пользователями, открыли новую страницу в истории современной микроэлектроники и вычислительной техники. Они сделали БИС/СБИС, предназначенные для решения специализированных задач, стандартной продукцией электронной промышленности со всеми вытекающими из этого положительными следствиями: массовое производство, снижение стоимости микросхем, сроков разработки и выхода на рынок продукции на их основе. ПЛИС можно классифицировать по многим признакам, в первую очередь:

- уровню интеграции и связанной с ним логической сложности;
- архитектуре (типу функциональных блоков, характеру системы межсоединений);
- числу допустимых циклов программирования;

- ☐ типу памяти конфигурации ("теневой" памяти);
- ☐ степени зависимости задержек сигналов от путей их распространения;
- ☐ системным свойствам;
- ☐ схемотехнологии (КМОП, ТТЛШ и др.);
- ☐ однородности или гибридности (по признаку наличия или отсутствия в микросхеме областей с различными по методам проектирования схемами, такими как ПЛИС, БМК, схемы на стандартных ячейках).

Все перечисленные признаки имеют значение и отображают ту или иную сторону возможных классификаций. Выделяя основные признаки и укрупняя их, рассмотрим классификацию по трем, в том числе двум комплексным, признакам:

- ☐ архитектуре;
- ☐ уровню интеграции и однородности/гибридности;
- ☐ числу допустимых циклов программирования и связанному с этим типу памяти конфигурации.

В классификации по первому признаку (рис. 1.2, а) ПЛИС разделены на 4 класса.

Первый из классов — SPLD, Simple Programmable Logic Devices, т. е. простые программируемые логические устройства. По архитектуре эти ПЛИС делятся на подклассы программируемых логических матриц ПЛМ (PLA, Programmable Logic Arrays) и программируемой матричной логики ПМЛ (PAL, Programmable Arrays Logic, или GAL, Generic Array Logic).

Оба эти подкласса микросхем реализуют дизъюнктивные нормальные формы (ДНФ) переключательных функций, а их основными блоками являются две матрицы: матрица элементов И и матрица элементов ИЛИ, включенные последовательно. Такова структурная модель ПЛМ и ПМЛ. Технически они могут быть выполнены и как последовательность двух матриц элементов ИЛИ-НЕ, но варианты с последовательностью матриц И-ИЛИ и с последовательностью матриц ИЛИ-НЕ — ИЛИ-НЕ функционально эквивалентны, т. к. второй вариант согласно правилу де Моргана тоже реализует ДНФ, но для инверсных значений переменных.

Заметим, что термин "матрица" обозначает в данном случае не более чем "набор", "множество" и обусловлен тем, что схемные элементы ПЛМ и ПМЛ удобнее всего располагать по строкам и столбцам, обеспечивая тем самым регулярность структуры БИС.

На входы первой матрицы поступают m входных переменных в виде как прямых, так и инверсных значений, так что матрица имеет $2m$ входных линий. На ее выходах формируются конъюнктивные термы, ранг которых не выше m . В дальнейшем для краткости конъюнктивные термы называются просто *термами*. Число термов не имеет прямой связи с величиной m и оп-

ределяется конструктивными параметрами матрицы — числом конъюнкторов, которое обозначим через q . Первая матрица идентична для обоих подклассов SPLD, т. е. для ПЛМ и ПМЛ.

Выработанные термы поступают на вход матрицы ИЛИ. Эти матрицы для ПЛМ и ПМЛ различны. В ПЛМ матрица ИЛИ программируется, а в ПМЛ она фиксирована.

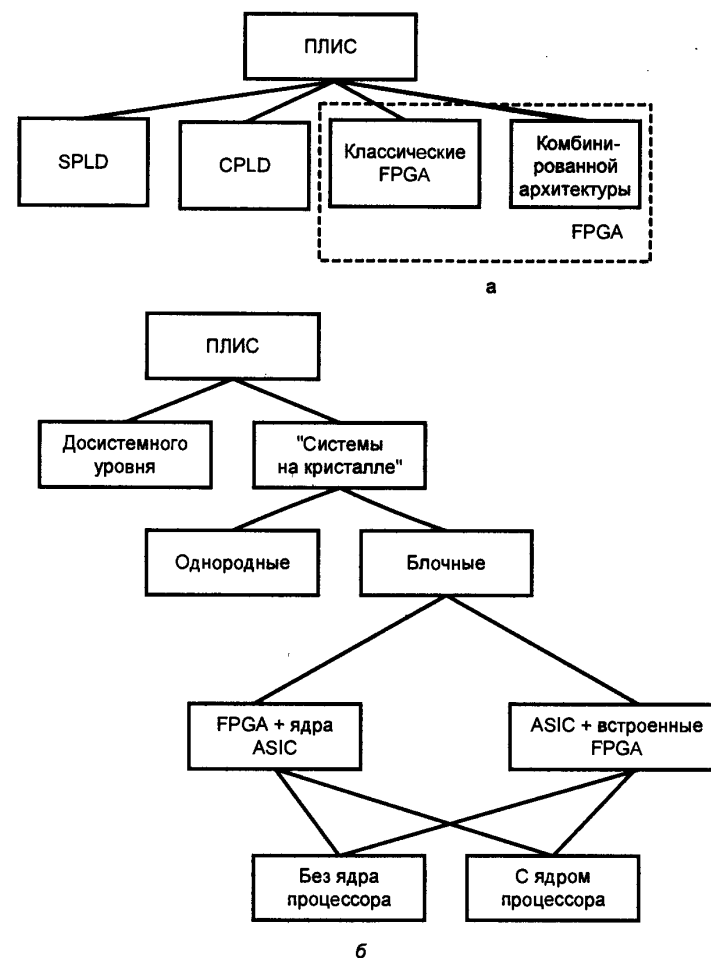


Рис. 1.2. Классификация ПЛИС по архитектурным признакам (а) и уровню интеграции (б)

Программируемая матрица ИЛИ микросхем ПЛМ составлена из дизъюнкторов, имеющих по q входов. На входы каждого дизъюнктора при программировании можно подать любую комбинацию имеющихся термов, причем термы можно использовать многократно (т. е. один и тот же терм может быть использован для подачи на входы нескольких дизъюнкторов).

Число дизъюнкторов в матрице ИЛИ определяет число выходов ПЛМ, которое обозначим через n .

Из изложенного видно, что ПЛМ позволяет реализовать систему из n переключательных функций, зависящих не более чем от m переменных и содержащих не более чем q термов. Поэтому функциональные возможности ПЛМ характеризуются тремя цифрами: m , q , n .

В ПМЛ выработанные матрицей И термы поступают на фиксированную (непрограммируемую) матрицу элементов ИЛИ. Это означает жесткое заранее заданное распределение имеющихся термов между отдельными дизъюнкторами. Каждому дизъюнктору придают свои собственные термы, и если для разных дизъюнкторов окажутся нужными одинаковые термы, придется вырабатывать их в матрице И несколько раз. Однако при этом программируемость матрицы ИЛИ исключается, что для многих задач в итоге существенно упрощает схему ПМЛ в сравнении со схемой ПЛМ.

ПЛМ обладают большей функциональной гибкостью, все воспроизводимые ими функции могут быть комбинациями любого числа термов, формируемых матрицей И. Это полезно при реализации систем переключательных функций, имеющих большие взаимные пересечения по термам. Такие системы свойственны, например, задачам формирования сигналов управления машинными циклами процессоров. Для широко распространенных в практике задач построения "произвольной логики" большое пересечение функций по термам не типично. Для них программируемость матрицы ИЛИ используется мало и становится излишней роскошью, неоправданно усложняющей микросхему. Поэтому ПМЛ распространены больше, чем ПЛМ, и к их числу относится большинство SPLD.

Структуры ПМЛ полнее освещены ниже при описании CPLD, составными частями которых они служат. Более подробные сведения о ПЛМ и ПМЛ имеются в ряде источников, в частности в [27].

В сложных программируемых логических схемах CPLD (Complex Programmable Logic Devices) несколько блоков, подобных ПМЛ, объединяются средствами программируемой коммутационной матрицы. В CPLD могут входить сотни блоков и десятки тысяч эквивалентных вентилях. Архитектуры CPLD разрабатываются фирмами Altera, Atmel, Lattice Semiconductor, Cypress Semiconductor, Xilinx и др. Воздействуя на программируемые соединения коммутационной матрицы и ПМЛ, входящих в состав CPLD, можно реализовать требуемую схему.

Микросхемы программируемых пользователями вентилях матриц FPGA (Field Programmable Gate Arrays) в своей основе состоят из большого числа конфигурируемых логических блоков ЛБ, расположенных по строкам и столбцам в виде матрицы, и трассировочных ресурсов, обеспечивающих их межсоединения. В архитектуре FPGA явно прослеживается большое сходство с архитектурой MPGA. Разница в том, что FPGA, поступающая в распоряжение потребителя, имеет уже готовые, стандартные, хотя и не запрограммированные, трассировочные ресурсы, не зависящие от конкретного потребителя. Получение конкретного проекта на базе FPGA, как и на основе других ПЛИС, реализуется воздействием на программируемые межсоединения, в ходе которого обеспечивается замкнутое состояние одних участков и разомкнутое — других. Обращаться к изготовителю FPGA при этом не требуется.

Архитектуры FPGA разрабатываются фирмами Xilinx, Actel, Altera, Atmel, Agere Systems (ранее Lucent Technologies), QuickLogic и др.

В течение первых лет развития ПЛИС они были представлены архитектурами CPLD и FPGA в "чистом" виде. Каждая из этих архитектур имеет свои достоинства и недостатки. Стремление к сочетанию достоинств CPLD и FPGA и рост уровня интеграции БИС/СБИС привели к появлению *ПЛИС с комбинированной архитектурой*. Класс ПЛИС с комбинированной архитектурой не имеет таких четких границ, как классы CPLD и FPGA, отличается большим разнообразием вариантов и различной степенью близости к тому или иному классическому типу ПЛИС. Не имеет он и общепринятого названия. Тем не менее, представляется целесообразным рассматривать ПЛИС с комбинированной архитектурой как отдельный класс, поскольку принадлежащие к нему схемы трудно квалифицировать как FPGA или CPLD, что подтверждается и разнообразием в названиях, используемых для таких схем различными фирмами. Примером первых ПЛИС с комбинированной архитектурой могут служить микросхемы семейств FLEX8000 и FLEX10K фирмы Altera (FLEX, Flexible Logic Element matrix).

Фактическое существование ПЛИС с комбинированной архитектурой и отсутствие для них общепринятого обобщающего названия вносят ощутимые неудобства в процесс составления классификации ПЛИС. Зачастую ПЛИС с комбинированной архитектурой представляются производителем под каким-либо конкретным именем, в котором не упоминаются ни CPLD, ни FPGA. Таких имен много, и на их основе не провести какую-либо классификацию. Мы не считаем целесообразным вводить свои термины для обозначения классов ПЛИС, т. к. сознаем, что общепризнанной окажется та терминология, которая исходит от крупнейших фирм-производителей микросхем этого типа. В то же время отнесение той или иной микросхемы в соответствующий раздел описания или справочной таблицы требует определенности в трактовке ее типа. Поэтому ниже наряду с узким применяется и *широкое толкование термина FPGA*. При этом выделяются "классические" FPGA с их

канонической архитектурой, а ПЛИС комбинированной архитектуры при необходимости (главным образом, при описании справочных данных) относятся к FPGA в широком смысле этого понятия. Обоснованием такого подхода служит то, что в комбинированных архитектурах черты FPGA обычно проявляются более выражено, чем черты CPLD. Правда, такой подход заставляет применять к некоторым схемам термины, которые самими производителями не используются (это относится, в частности, к фирме Altera, которая никогда не пользуется термином FPGA), но, по нашему мнению, возникающие разночтения являются следствием только рекламно-маркетинговых соображений. Следует заметить, что авторы, не связанные с фирмой Altera, этого правила придерживаются не всегда и называют некоторые микросхемы данной фирмы термином FPGA [44]. Сказанное выше отображается на рис. 1.2, а объединением классических FPGA и ПЛИС с комбинированными архитектурами общим прямоугольником из штриховых линий.

Термин SOPC (System On Programmable Chip), т. е. "система на программируемом кристалле" относится к ПЛИС наибольшего уровня интеграции, содержащим сотни тысяч или даже миллионы эквивалентных вентилей. Такой высокий уровень интеграции достигается только с помощью самых современных технологических процессов (малые топологические нормы проектирования, многослойность систем металлизации и т. д.). На основе прогрессивных технологических процессов обеспечивается одновременно высокий уровень интеграции и высокое быстродействие БИС/СБИС. В результате становится возможной интеграция на одном кристалле целой высокопроизводительной системы.

Классификация по уровню интеграции (рис. 1.2, б) дана кратко и отражает, главным образом, ситуацию последних годов — бурный рост уровня интеграции ПЛИС и выделение из них класса "системы на кристалле". В силу связи между уровнем интеграции и архитектурой, классификация не является строгой, и в ней имеется некоторое смешение двух признаков, однако она принята в показанном виде ради соответствия практически сложившимся понятиям.

ПЛИС с широким диапазоном изменения уровня интеграции (от простых до содержащих сотни тысяч вентилей) отнесены к "досистемным" в том смысле, что для них не рассматривались вопросы создания целых систем на одном кристалле.

ПЛИС мегавентильного уровня интеграции отнесены к "системам на кристалле", обозначаемым далее как SOPC (Systems On Programmable Chip). Заметим, что для программируемых систем на кристалле разные фирмы, как правило, используют свои обозначения (PSOC, CSOC, FIPSOC и т. д.), регистрируемые как товарные знаки. В разд. 1.10 данной главы, посвященном справочным данным об ИС с программируемыми структурами, индивидуальные обозначения фирм сохранены. Для обобщающего термина принято обозначение SOPC, введенное в обиход фирмой Altera.

Класс SOPC делится на подклассы *однородных* и *блочных* систем на кристалле.

В однородных SOPC различные блоки системы реализуются одними и теми же аппаратными средствами, благодаря программируемости этих средств. При разработке систем используются так называемые "единицы интеллектуальной собственности" IP (Intellectual Properties), т. е. заранее реализованные параметризуемые мегафункции для создания тех или иных частей системы. Все блоки системы при этом являются полностью синтезируемыми, перемещаемыми и могут располагаться в разных областях кристалла. Создание IP стало важной сферой деятельности многих фирм, предлагающих на рынке широкий спектр разнообразных решений. Заметим, что приобретение IP обычно требует немалых затрат. Используя IP, проектировщик размещает на кристалле нужные ему блоки, которые будем называть *soft-ядрами* (Softcores).

Блочные SOPC имеют *аппаратные ядра*, т. е. специализированные области кристалла, выделенные для определенных функций. В этих областях создаются блоки неизменной структуры, спроектированные по методологии ASIC (как области типа БМК или схем со стандартными ячейками), оптимизированные для заданной функции и не имеющие средств ее программирования. Такие блоки будем называть *hard-ядрами* (Hardcores). Реализация функций специализированными аппаратными ядрами требует значительно меньшей площади кристалла в сравнении с реализациями на единых однородных программируемых средствах и улучшает другие характеристики схемы, в первую очередь, быстродействие блоков, но уменьшает универсальность ПЛИС. Снижение универсальности сужает круг потребителей ПЛИС, т. е. тиражность их производства, что, в противовес факторам, удешевляющим схему, ведет к их удорожанию. Преобладание того или иного из указанных факторов зависит от конкретной ситуации.

Кристаллы, содержащие как области ПЛИС, так и области ASIC, в различных источниках именуются по-разному. В сетевом журнале Electronic Design News встречается термин "гибридные", по смыслу близкий к принятому нами термину "блочные" (мы воздержались от термина "гибридные" в силу традиции, по которой этот термин применялся для схем со смешанными цифровыми и аналоговыми сигналами).

Сейчас на рынке появилось большое число различных SOPC, и среди них намечились свои подклассы и проблемные ориентации. Не пытаясь детально классифицировать все варианты, целесообразно разделить блочные SOPC хотя бы на две группы: *имеющие аппаратные ядра процессоров* и *не имеющие их*. Первые представляют БИС/СБИС по-настоящему универсальные, т. к. содержат полный комплект блоков, характерных для микропроцессорной системы (имеется в виду цифровая часть системы, но следует отметить, что у некоторых SOPC есть и аналоговые блоки для ввода, предварительной обработки и последующей оцифровки аналоговых сигналов). Вторые специализированы и ориентированы на те или иные конкретные приложения. Но

и здесь нужно отметить наличие у некоторых SOPC второго типа интерфейсных средств для сопряжения с процессором и ОЗУ различных типов, что облегчает построение целостных систем с применением таких SOPC.

Классификация по признаку кратности программирования, определяемой типом памяти конфигурации (называемой также теневой памятью), приведена на рис. 1.3.

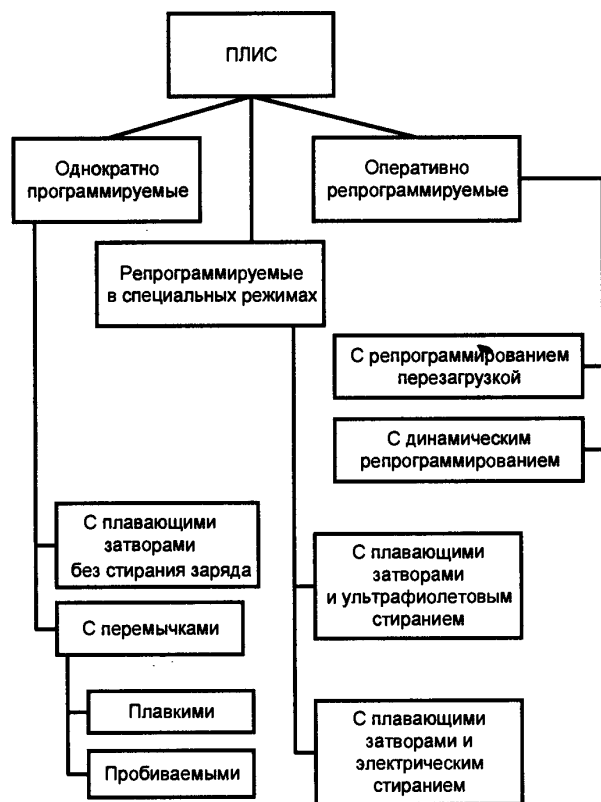


Рис. 1.3. Классификация ПЛИС по признакам кратности программирования (по типу теневой памяти)

В однократно программируемых ПЛИС используются элементы с необратимыми изменениями состояний — специальные *перемычки* или *ЛИЗМОП-транзисторы*. ЛИЗМОП-транзисторы имеют заряжаемые "плавающие" затворы, которые, в общем случае, могут как заряжаться, так и разряжаться. Для однократно программируемых ПЛИС возможности ЛИЗМОП-транзисторов

используются лишь частично: для них применяются такие конструкции, в которых отсутствуют возможности стирания записанной информации.

В простых ПЛИС первых поколений применялись *плавкие перемычки* типа fuse. В таких ПЛИС в исходном состоянии имеются все возможные соединения, а для получения требуемой конфигурации схемы часть перемычек разрушается (пережигается). При программировании плавких перемычек возникает определенный процент брака, кроме того, со временем проводимость разрушенной перемычки может восстановиться из-за явления электромиграции в материалах. В течение многих лет велась большая работа по подбору материалов перемычек и усовершенствованию технологических процессов программирования, тем не менее для ПЛИС высокого уровня интеграции схемы с плавкими перемычками не подошли.

В однократно программируемых FPGA нашли применение *пробиваемые перемычки* типа antifuse. В исходном состоянии сопротивления перемычек чрезвычайно велики, а в пробитом достаточно малы. Перемычки очень компактны — их площадь близка к площади пересечения двух дорожек межсоединений. Паразитные емкости перемычек также очень малы. Больших успехов в разработке техники пробиваемых перемычек добились фирмы Actel (перемычки типа ONO), QuickLogic (перемычки ViaLink), Crosspoint Solution (кремниевые-аморфные перемычки) и Xilinx (перемычки MicroVia).

В третьем варианте (с плавающими затворами) роль программируемых элементов играют *однозатворные ЛИЗМОП-транзисторы*, а кристаллы микросхем размещаются в дешевых корпусах, не имеющих специальных окошек для стирания информации (зарядов в плавающих затворах). Для транзисторов с одним (плавающим) затвором и каналом р-типа до программирования затвор не имеет заряда, и транзистор заперт. Введение в затвор заряда электронов приводит к возникновению в транзисторе проводящего канала. Заряд плавающего затвора сохраняется в течение десятков лет. Память конфигурации с элементами описанного типа называют EPROM-OTP (Electrically Programmable Read-Only Memory — One Time Programmable). Однозатворные ЛИЗМОП-транзисторы компактны и дешевы.

Микросхемы с возможностями *многократного программирования со стиранием* и записью конфигурации в специальных режимах также используют программируемые элементы в виде ЛИЗМОП-транзисторов. От предыдущего варианта эти микросхемы отличаются наличием средств стирания записанной в память информации.

Информация в элементах памяти типа EPROM стирается с помощью облучения кристалла ультрафиолетовыми лучами, что отражается в русском термине для этой памяти — РПЗУ-УФ (репрограммируемые запоминающие устройства с ультрафиолетовым стиранием). Кристалл со стертой конфигурацией можно запрограммировать вновь. Стирание конфигурации занимает десятки минут, а число циклов репрограммирования ограничено величина-

ми порядка десятков-сотен, т. к. ультрафиолетовое облучение постепенно изменяет свойства кристалла.

В репрограммируемых ПЛИС с памятью конфигурации типа EEPROM (Electrically Erasable Programmable Read-Only Memory) стирание старых данных осуществляется электрическими сигналами. Используются двухзатворные ЛИЗМОП-транзисторы. Управление процессами в транзисторе производится с помощью двух затворов — обычного и плавающего. При определенных сочетаниях программирующих напряжений на внешних выводах транзистора (плавающий затвор внешнего вывода не имеет) создаются режимы как заряда плавающих затворов, так и их разряда. В русской терминологии память типа EEPROM называют РПЗУ-ЭС (репрограммируемые запоминающие устройства с электрическим стиранием). Электрическое стирание содержимого памяти не требует извлечения микросхем из устройства, в котором они используются. Число допустимых циклов репрограммирования хотя и ограничено, но на порядки превышает соответствующие числа для памяти с ультрафиолетовым стиранием информации и составляет 10^5 – 10^6 . Элементы памяти с электрическим стиранием вначале заметно проигрывали элементам с ультрафиолетовым стиранием по площади, занимаемой на кристалле, но быстро совершенствуются и становятся преобладающими для класса ПЛИС со стиранием конфигурации в специальных режимах.

К памяти типа EEPROM близка память конфигурации типа Flash. Запоминающие элементы по своему принципу действия у обоих видов памяти идентичны — это ЛИЗМОП с двумя затворами и электрическим стиранием информации. Различия имеются в организации процессов записи и стирания данных и, кроме того, при разработке Flash-памяти достигнут особенно высокий уровень параметров (быстродействия, уровня интеграции, надежности и др.). Разработку Flash-памяти считают кульминационным пунктом десятилетнего развития памяти типа EEPROM. В современных ПЛИС находят применение обе разновидности памяти конфигурации с электрическим стиранием данных.

Последний класс ПЛИС по второму признаку принятой классификации — *оперативно репрограммируемые*. В таких ПЛИС конфигурация задается с помощью загрузки файла в "теневую" триггерную память, т. е. операций, не имеющих какого-либо специального характера. В противоположность предыдущим вариантам для программирования не нужны ни специальные программаторы, ни специальные режимы с повышенными напряжениями и длительностями воздействий на элементы памяти. Память конфигурации — обычная статическая (триггерная), т. е. типа SRAM, Static Random Access Memory. Загрузка памяти производится с высокой скоростью, свойственной статической триггерной памяти, последовательным потоком битов или байтов. Элементом с программируемой проводимостью (режимом "замкнуто-разомкнуто") служит обычный МОП-транзистор, управляемый триггером памяти конфигурации (теневой памяти). Состояние триггера задает режим ключевому транзистору. *Программирование соединения сводится к установке*

триггера в состояние 0 или 1. В рабочем режиме триггер сохраняет неизменное состояние. Стирание старой конфигурации и запись новой могут производиться неограниченное число раз путем перезагрузки памяти конфигурации.

Триггерная память не является *энергонезависимой*, и *выключение питания ведет к разрушению конфигурации ПЛИС*, поэтому при очередном его включении нужно ее восстановить, загрузив в триггеры теневой памяти файл конфигурации из какой-либо энергонезависимой памяти. Загрузка производится введением файла конфигурации в цепочку триггеров теневой памяти и, в зависимости от объема файла, занимает десятки-сотни миллисекунд.

Программируемые соединения с триггерной памятью сложнее, чем предыдущие варианты, т. к. для каждого соединения требуются ключевой транзистор, триггер и цепи выборки и сброса/установки для управления триггером. Тем не менее эти ПЛИС в силу ряда достоинств занимают среди ПЛИС/СБИС с программируемыми структурами очень важное место.

Возможности оперативной реконфигурации, свойственные ПЛИС с триггерной памятью, получили дальнейшее развитие в *архитектурах с динамическим репрограммированием*. В ПЛИС с динамическим репрограммированием конфигурация может быть изменена чрезвычайно быстро. Переход от одной конфигурации к другой не требует ввода извне нового файла конфигурации. Несколько вариантов настроек (файлов конфигурации) уже заранее заготовлены и введены в теневую память и постоянно хранятся в ней. Переход от одной конфигурации на другую делается однократно по команде управляющего сигнала.

ПЛИС с динамическим репрограммированием открывают ряд новых возможностей в области построения устройств и систем с многофункциональным использованием аппаратных ресурсов для решения сложных задач при их разбиении на последовательные этапы и реализации разных этапов на одних и тех же быстро перестраиваемых ПЛИС.

1.1.4. Общие (системные) свойства микросхем программируемой логики

ПЛИС рассматриваются в настоящее время как наиболее перспективная элементная база для построения цифровой аппаратуры разнообразного назначения. Появляются и новые возможности реализации на программируемых микросхемах аналоговых и аналого-цифровых устройств. Перспективность ПЛИС базируется на ряде их достоинств, к числу которых можно отнести перечисленные ниже, справедливые для ПЛИС вообще, безотносительно к их конкретным разновидностям.

- Универсальность и связанный с ней высокий спрос со стороны потребителей, что обеспечивает массовое производство.

- ❑ Низкая стоимость, обусловленная массовым производством и высоким процентом выхода годных микросхем при их производстве вследствие достаточно регулярной структуры.
- ❑ Высокое быстродействие и надежность как следствие реализации на базе передовых технологий и интеграции сложных устройств на одном кристалле.
- ❑ Разнообразие конструктивного исполнения, поскольку обычно одни и те же кристаллы поставляются в разных корпусах.
- ❑ Разнообразие в выборе напряжений питания и параметров сигналов ввода/вывода, а также режимов снижения мощности, что особенно важно для портативной аппаратуры с автономным питанием.
- ❑ Наличие разнообразных, хорошо развитых и эффективных программных средств автоматизированного проектирования, малое время проектирования и отладки проектов, а также выхода продукции на рынок.
- ❑ Простота модификации проектов на любых стадиях их разработки.

Для новейших вариантов ПЛИС с динамическим репрограммированием структур кроме важных с общих позиций свойств следует назвать и дополнительную специфическую черту: *возможность построения на базе динамически репрограммируемых микросхем новых классов аппаратуры с многофункциональным использованием блоков.*

1.2. CPLD — сложные программируемые логические устройства

1.2.1. Структура CPLD

CPLD — микросхемы высокого уровня интеграции, основными частями которых являются:

- ❑ PAL (GAL) — подобные функциональные блоки;
- ❑ система коммутации, позволяющая объединять функциональные блоки в единое устройство, выполненная в виде матрицы соединений.
- ❑ блоки ввода/вывода.

Все составные части CPLD программируются. Обобщенная структура CPLD показана на рис. 1.4.

В структурной схеме приняты следующие обозначения. Через ФБ (FB) обозначены функциональные блоки, число которых N зависит от уровня интеграции микросхемы и изменяется в довольно широких пределах. В каждом ФБ имеется n макроячеек МЯ (МС, Macrocells). Функциональные блоки получают входные сигналы от программируемой матрицы соединений ПМС

(PIA, Programmable Interconnect Array). Число таких сигналов m . Выходные сигналы ФБ поступают как в ПМС, так и в блоки ввода/вывода CPLD (IOBs, Input/Output Blocks, БВВ). ПМС обеспечивает полную коммутируемость функциональных блоков, т. е. возможность подавать сигналы с любого их выхода на любой вход.

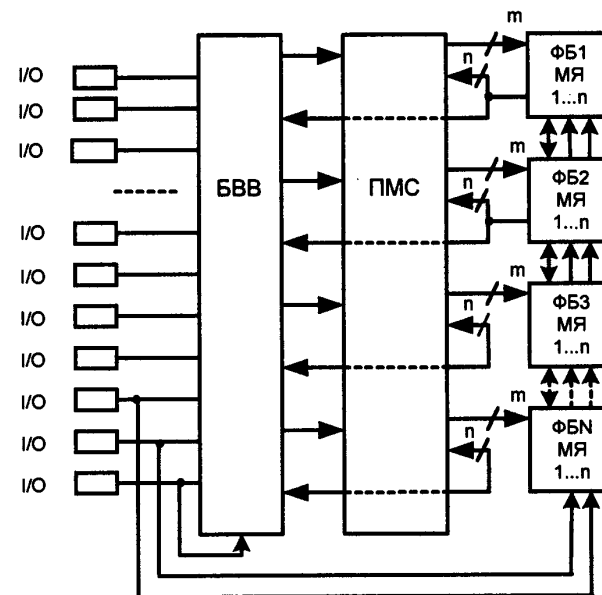


Рис. 1.4. Обобщенная структура CPLD

Блоки ввода/вывода связаны с внешними двунаправленными выводами I/O, которые, в зависимости от программирования, могут быть использованы как входы или как выходы. Три нижних вывода либо специализируются для подачи на матрицу функциональных блоков сигналов GCK (Global Clocks) глобального тактирования, сигналов GSR (Global Set/Reset) глобальной установки/сброса и сигналов GTS (Global 3-state Control) глобального управления третьим состоянием выходных буферов, либо эти же выводы могут быть использованы для операций ввода/вывода. Здесь и далее термин "глобальный" применяется для сигналов, общих для всей микросхемы.

Число контактов ввода/вывода может совпадать с числом выходов всех ФБ, но может быть и меньшим. В последнем случае часть макроячеек может быть использована только для выработки внутренних сигналов устройства

(в частности, сигналов обратной связи). Необходимость в таких сигналах типична для структур большинства цифровых устройств.

Структура на рис. 1.4 несколько упрощена. Кроме показанных блоков в CPLD могут присутствовать контроллеры для управления операциями программирования непосредственно в системе (In System Programmability, ISP), контроллеры для работы с интерфейсом JTAG и др. Вопросы, связанные с использованием интерфейса JTAG и программированием в системе ISP, рассмотрены в разд. 2.6.

1.2.2. Программируемая матрица соединений

В программируемой матрице соединений ПМС (рис. 1.5, а) выходы функциональных блоков ФБ подключаются к вертикальным непрерывным (не сегментированным) линиям, причем каждому выходу соответствует своя линия. Входы ФБ связаны с горизонтальными линиями, пересекающими все вертикальные линии. На пересечениях горизонтальных и вертикальных линий имеются программируемые точки связи, так что любой вход ФБ может быть подключен к любому выходу, чем обеспечивается так называемая полная коммутруемость блоков.

Достоинством ПМС рассмотренного типа является *малая и предсказуемая задержка* коммутируемых сигналов, т. к. для каждого соединения образуется идентичный всем другим канал связи с малым числом программируемых ключей или даже их отсутствием, если передача сигналов из ПМС в ФБ организована так, как показано на рис. 1.5, б. В этом случае программируемых ключей в цепи передачи сигнала нет, программируются только напряжения на нижних входах конъюнкторов, и ФБ получит сигнал от i -й вертикальной линии ПМС ($i = 1, 2, \dots, m$), если транзистор T_i будет заперт, и на нижнем входе i -го конъюнктора будет действовать высокий потенциал логической единицы. Открытый транзистор T_i подключает нижний вход конъюнктора к нулевому потенциалу, создавая на нем и на выходе конъюнктора сигнал логического нуля. Таким образом, задавая триггеру T_i состояние логического нуля, а остальным триггерам состояние логической единицы, можно обеспечить закрытое состояние транзистора T_i и открытое состояние всех других транзисторов, что означает подключение выхода ФБ к i -й вертикальной линии ПМС с образованием так называемого непрерывного соединения.

Замкнутые транзисторные ключи имеют, в первом приближении, схему замещения в виде инерционной RC-цепи и вносят основные задержки в процесс распространения сигнала.

Программируемые матрицы соединений типа показанной на рис. 1.5, а эффективны в схемах с относительно небольшим числом коммутируемых блоков. При большом их числе, характерном, например, для FPGA, подобные

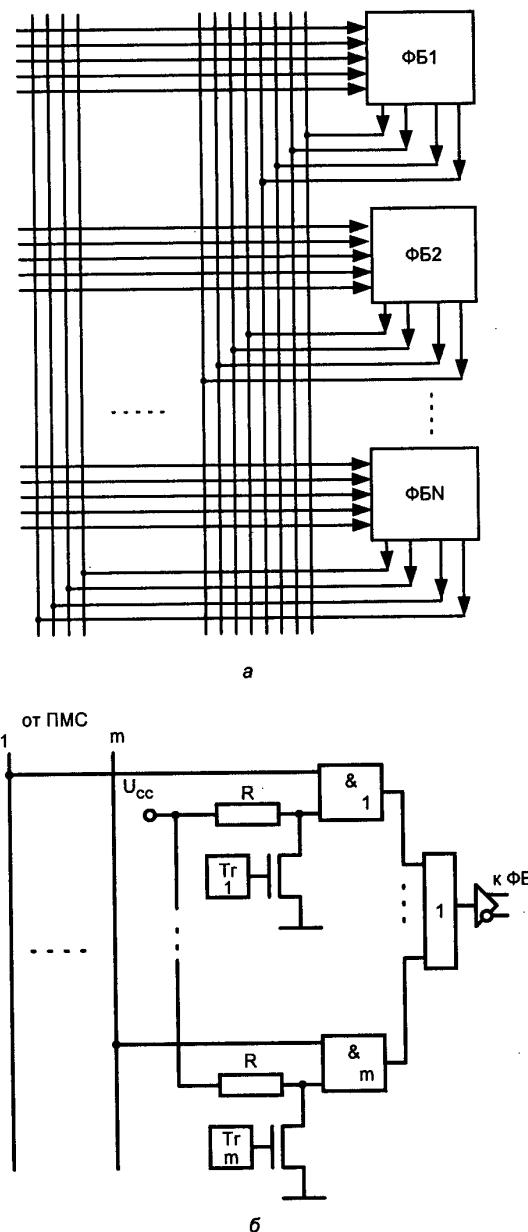


Рис. 1.5. Схема программируемой матрицы соединений CPLD (а) и схема передачи сигналов из этой матрицы в функциональный блок (б)

ПМС были бы чрезмерно сложны, поскольку любое соединение образуется с помощью линий связи, проходящих по всей длине и ширине схемы, тогда как очень многие связи локальны и соединяют близлежащие блоки. Поэтому в FPGA системы коммутации строятся иначе — с помощью сегментированных линий связи.

1.2.3. Функциональные блоки CPLD

CPLD различных фирм-изготовителей и разной сложности имеют функциональные блоки, в принципиальном отношении мало отличающиеся друг от друга по своей архитектуре и составу элементов. Обобщенная структура функционального блока CPLD показана на рис. 1.6.

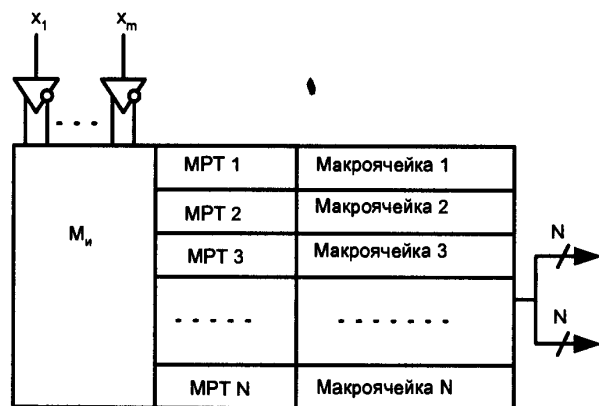


Рис. 1.6. Структура функционального блока CPLD

Основными частями функциональных блоков CPLD являются программируемая матрица элементов И (M_n), матрица распределения термов MPT и группа из нескольких (N) макроячеек. По существу, каждый ФБ представляет собою PAL-подобную структуру с некоторыми отличиями от вариантов, используемых в простых PLD (ПМЛ). Как и в классических PLD, в блоке имеется многовходовая (Wide) матрица M_n , вырабатывающая конъюнктивные термы для их использования в последующих частях блока. В классических PLD типа ПМЛ термы жестко распределяются между дизъюнкторами, формирующими выходные функции в форме ДНФ. Совокупность дизъюнкторов образует фиксированную (не программируемую) матрицу элементов ИЛИ. На рис. 1.7, а показан один из дизъюнкторов (для канала с номером i), вырабатывающий функцию, в которую может входить не более 5 термов.

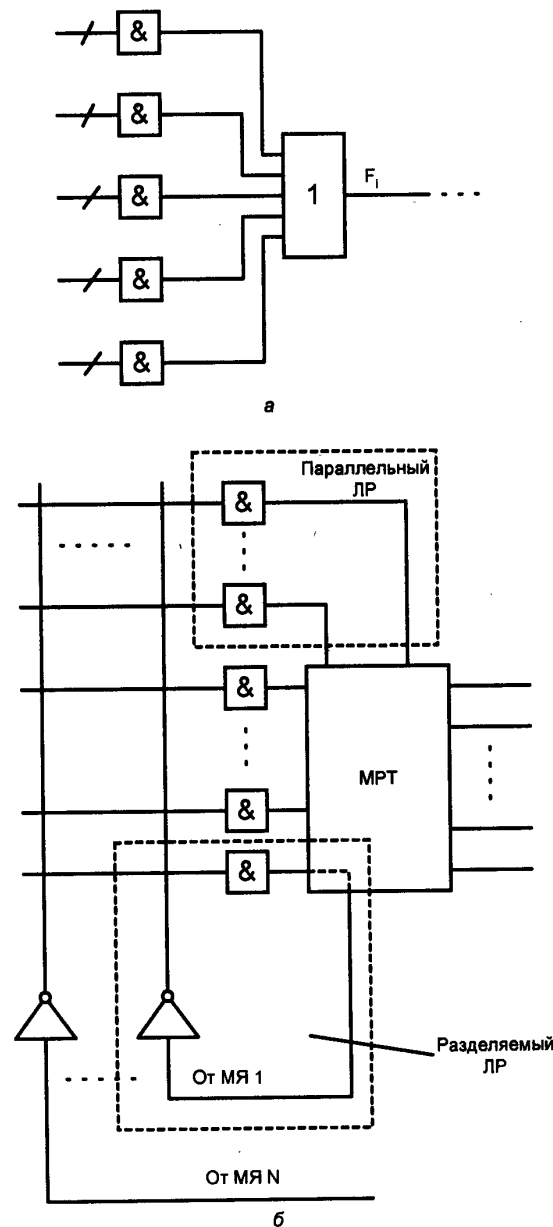


Рис. 1.7. Схема выработки ДНФ логической функции в простейшем варианте CPLD (а) и логические расширители параллельного и последовательного типов (б)

В CPLD матрица элементов ИЛИ чаще всего не является полностью фиксированной, и благодаря введению в схему матрицы распределения термов MPT, возможно варьирование числа термов в вырабатываемой функции F_i . При этом термы заимствуются у других каналов выработки функций или отдаются им. Проще всего организовать коммутацию термов между соседними каналами. Через соседние каналы путем образования цепочечных связей можно собирать в одном канале много термов (в пределах одного функционального блока). Если термы используются не только дизъюнкторами формирования выходных функций, но и другими элементами ФБ, то и для них MPT играет роль "раздатчика термов".

Схемотехнически в операциях распределения термов по каналам ФБ задействованы как непосредственно цепи коммутации между входами и выходами MPT, так и логические расширители последовательного и параллельного типов. *Последовательные (разделяемые, общие)* логические расширители создаются подачей инвертированного значения терма из MPT данного канала обратно на один из входов матрицы M_i (рис. 1.7, б). Переданный в матрицу M_i терм становится доступным для использования во всех каналах данного ФБ. Если, например, этот терм запрограммирован как $q = x1!x2x5!x10$, т. е. его инверсия есть $!q = !(x1!x2x5!x10) = !x1vx2vx5vx10$, то в том канале, где он будет использован вместе с входными термами канала, например, $q1, q2, q3$, будет получена функция

$$F = q1vx2vx3vx!x1vx2vx!x3vx10 = q1vx2vx3vx!(x1!x2x5!x10).$$

Параллельный расширитель позволяет передавать термы одного канала другому. Способность принимать в свой канал термы от соседнего канала обычно означает и возможность приема через него термов и более далеких каналов с образованием цепочки для сбора термов от нескольких каналов (например, в пределах целого функционального блока). Можно, естественно, и отдавать собственные термы или их часть другим каналам (в частности, соседним, а через них и более далеким).

Термы от MPT поступают далее на часть ФБ, называемую *макроячейкой* (МЯ). Макроячейка содержит в качестве основы программируемые мультиплексоры, триггер (или триггеры) и формирует группу выходных сигналов ФБ в нескольких их вариантах.

Замечание

Термин "макроячейка" трактуется в разных источниках по-разному. Наряду с указанным определением встречаются и такие, согласно которым макроячейкой называют часть ФБ, содержащую все его последовательно включенные части (группу элементов И, MPT и все последующие элементы, относящиеся к данному каналу выработки выходных функций).

На рис. 1.8 раскрыта схема одной из идентичных макроячеек ФБ. Преобразование показанной схемы является макроячейкой CPLD XC9500 фирмы Xilinx, не имеющая разделяемого (общего) логического расширителя.

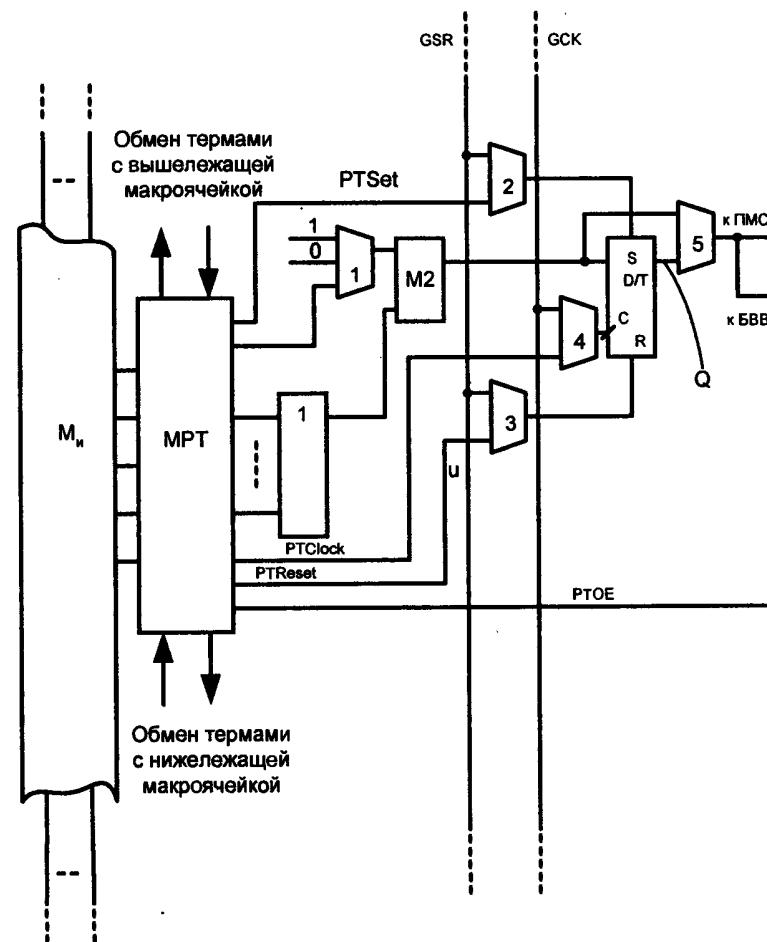


Рис. 1.8. Пример схемы макроячейки функционального блока CPLD

Программируемость мультиплексоров в этой схеме и всех дальнейших не отображается, т. к. она присуща всем имеющимся в схеме мультиплексорам, если не оговорено противоположное. В зависимости от программирования каждый мультиплексор передает на выход сигнал с того или иного входа. Триггер может программироваться на режимы работы триггера типа D или T. Заметим, кстати, что при описании микросхем программируемой логики триггеры в иностранной литературе чаще всего называют *регистрами*. Триггеры тактируются положительными фронтами синхросигналов и имеют входы установки S и сброса R. Выходные сигналы ФБ передаются в ПМС и в блоки ввода/вывода БВВ.

Аргументы $x1, \dots, xM$ реализуемой макроячейкой функции поступают на матрицу M_i из ПМС. Аргументами для МЯ могут быть как входные сигна-

ОЕ с помощью программируемого мультиплексора MUX3 вырабатывается в нескольких вариантах: от термина ПТОЕ, получаемого от микроячейки, от любого из глобальных сигналов управления третьим состоянием (GOE1, GOE2), от константы 1 и от константы 0. Глобальные сигналы управления третьим состоянием образуются с возможностью выбора любой полярности исходных сигналов GTS1 и GTS2.

Выходные буферы конфигурируются для работы с напряжениями питания 5 или 3,3 В при подключении внешнего источника питания с тем или иным уровнем напряжения (эти цифры относятся к рассматриваемому блоку типа XC9500, сейчас у блоков ввода/вывода нередко уровни выходных сигналов могут выбираться из многих возможностей, в том числе из таких низких напряжений, как 2,5 и 1,8 В).

Согласно принятой нами классификации в класс CPLD попадают ПЛИС с уровнем интеграции 600—20 000 эквивалентных вентилях, числом макроячеек 32—512, числом функциональных блоков 2—16 и временем распространения сигнала от любого входа до любого выхода 5—20 нс. Эти CPLD представлены, в частности, такими популярными семействами микросхем, как MAX7000 и MAX3000A фирмы Altera, XC9500 фирмы Xilinx, MACH1...MACH5 фирмы Lattice Semiconductor (ранее семейства MACH выпускались известной фирмой AMD, позднее подразделение этой фирмы Vantis вместе с производством указанного семейства было приобретено фирмой Lattice Semiconductor). В схемах CPLD используется энергонезависимая память конфигурации, причем доминирует память типа EEPROM или Flash. Энергонезависимость памяти конфигурации облегчает засекречивание проектов, т. к. не требует загрузки внешних данных конфигурации, которые доступны для чтения. Содержимое памяти конфигурации на самом кристалле обычно защищается специальным битом секретности, сбросить который можно лишь при стирании всего содержимого памяти.

1.3. FPGA — программируемые пользователем вентильные матрицы

1.3.1. Основные сведения

В наиболее типичном варианте FPGA представляет собою микросхему высокого уровня интеграции, содержащую во внутренней области матрицу идентичных функциональных блоков и систему их межсоединений, размещенную между строками и столбцами матрицы, а в периферийной области — блоки ввода/вывода (рис. 1.10, а). Кроме этого варианта существуют FPGA, в которых функциональные блоки расположены по строкам (строковые FPGA), однако рассматривать эти варианты отдельно нет оснований, поскольку существенные черты FPGA остаются одинаковыми для обоих вариантов.

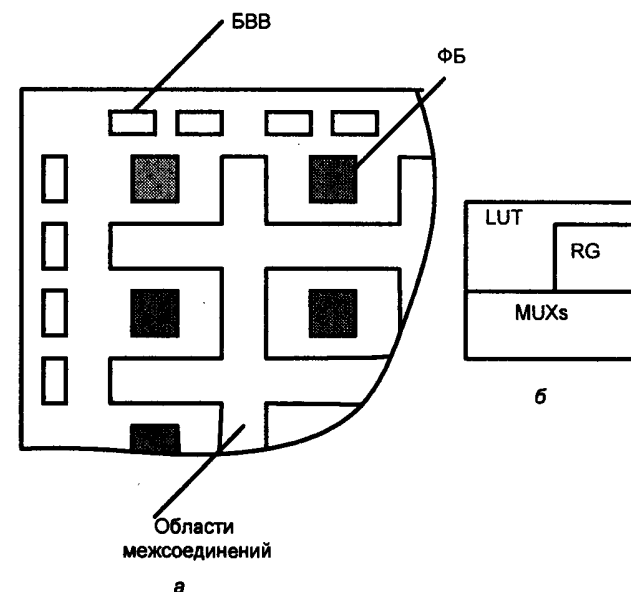


Рис. 1.10. Обобщенная структура FPGA (а) и основные части их функциональных блоков (б)

Все части FPGA (функциональные блоки ФБ, система межсоединений и блоки ввода/вывода БВВ) являются конфигурируемыми или реконфигурируемыми, причем (в отличие от БМК) средствами самих пользователей.

Перечисленные части — основа FPGA. Кроме них современные варианты FPGA, как правило, оснащены дополнительными средствами для автоподстройки задержек в системе тактирования (PLL, Phase Locked Loop или DLL, Delay Locked Loop), средствами поддержки интерфейса JTAG и др.

При конфигурировании FPGA функциональные блоки настраиваются на выполнение необходимых операций преобразования данных, а система межсоединений — на требуемые связи между функциональными блоками. В результате во внутренней области FPGA реализуется схема нужной конфигурации. Расположенные по краям кристалла блоки ввода/вывода обеспечивают интерфейс FPGA с внешней средой. Блоки ввода/вывода современных FPGA можно программировать на выполнение требований множества стандартов передачи данных (число таких стандартов может достигать до 20).

На рис. 1.10, б укрупненно показан состав типичного функционального блока ФБ, в который входят функциональный преобразователь ФП, реализованный в виде программируемого запоминающего устройства (LUT, Look-Up Table), триггер (регистр) и мультиплексоры, играющие роль средств конфигурирования ФБ.

LUT — наиболее распространенная разновидность ФП в FPGA со статической памятью конфигурации. В схемах FPGA с однократным программированием перемычек находят применение ФП в виде простых логических вентилей (SLC, Simple Logic Cell) и логических модулей на основе мультиплексоров [27].

1.3.2. Функциональные блоки FPGA

Табличный ФП типа LUT представляет собою запоминающее устройство, хранящее значения искоемых функций, считываемые по адресу-аргументу. Запоминающее устройство с организацией $2^m \times n$ имеет m адресных входов и n выходных линий. В таком запоминающем устройстве можно хранить таблицу для считывания n функций от m переменных, поскольку каждый разряд хранимого массива данных можно рассматривать как столбец значений переключательной функции. В память можно записывать любые числа, и, следовательно, разрядные столбцы этих чисел могут воспроизводить любые функции m переменных. Время вычисления результата не зависит от воспроизводимой функции и равно времени считывания слова из памяти.

Второй тип ФБ — схема, основой которой служат *программируемые мультиплексоры* (фирма Actel и др.). В этих ФБ выходная величина описывается некоторой так называемой порождающей функцией, соответствующей использованию всех входов схемы как информационных. При программировании на некоторые входы задаются константы 0 и 1, разные сочетания которых порождают целый спектр возможностей воспроизведения логических функций. Подробнее о таких возможностях говорится в работах [27, 30 и др.].

Третий тип ФБ — так называемые *"мелкозернистые"*, блоки, составленные, чаще всего, из транзисторных пар, выделяемых из цепочек транзисторов с n - и p -каналами. Из таких пар собираются традиционные для КМОП-схем логические элементы, методы синтеза которых традиционны.

Типичную структуру ФБ FPGA с триггерной памятью конфигурации рассмотрим на примере микросхемы семейства Spartan фирмы Xilinx (рис. 1.11). Фирма Xilinx изобрела и активно разрабатывает оперативно реконфигурируемые FPGA, ее продукция занимает 30—40% мирового рынка ПЛИС. После семейства Spartan фирма выпустила семейство Spartan II, но нами для иллюстрации особенностей FPGA выбрано первое из этих семейств, т. к. второе по схемотехнике блоков аналогично семействам Virtex и Virtex E, рассмотренным ниже.

По своей структуре микросхемы Spartan продолжают линию развития классических FPGA, популярным представителем которой является семейство XC4000 той же фирмы. В функциональных блоках этих микросхем логические преобразования выполняются тремя LUT-блоками (функциональными преобразователями ФП) G, F и H. Преобразователи G и F — программи-

руемые запоминающие устройства (ЗУ) с организацией 16×1 , способные воспроизводить любые функции четырех переменных, значения которых могут быть переданы на выходы Y и X через мультиплексоры 4 и 6 при соответствующем их программировании (через линии верхних входов мультиплексоров).

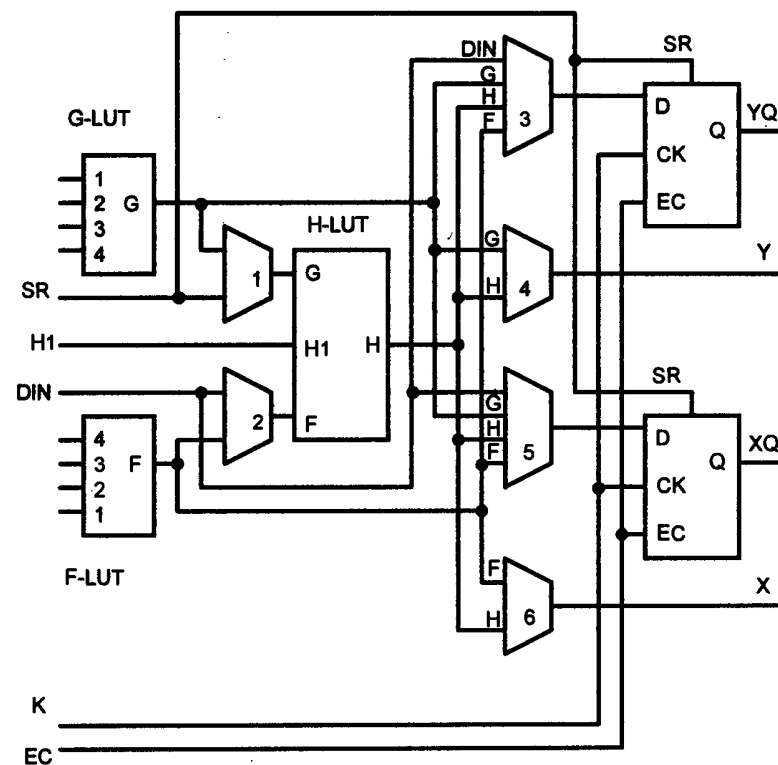


Рис. 1.11. Пример схемы функционального блока FPGA

Заметим, что на рисунках, как и ранее, в обозначениях мультиплексоров не отражена их программируемость, поскольку все они без исключения обладают этим свойством.

Через верхний вход мультиплексора 1 и нижний вход мультиплексора 2 функции G и F могут быть поданы на ФП-Н (ЗУ с организацией 8×1) для образования "функции от функций" с целью получения результирующей функции, зависящей от более чем четырех аргументов. К третьему входу ФП-Н подключен входной сигнал H1, так что $H = f(G, F, H1)$. Аргументами для ФП-Н, поступающими от мультиплексоров 1 и 2, в зависимости от

их программирования может быть не только набор G, F, H1, но также наборы G, H1, DIN; SR, H1, DIN; SR, H1, F. Линии DIN и SR используются либо для передачи в триггер непосредственно входных данных и сигнала установки/сброса (Set/Reset), либо как входы ФП-Н.

Перечисленные ресурсы логической части ФБ позволяют воспроизводить:

- любую функцию с числом аргументов до 4 включительно плюс вторую такую же функцию плюс любую функцию с числом аргументов до трех;
- любую функцию 5 аргументов (одну);
- любую функцию 4 аргументов и одновременно некоторые функции 6 аргументов, некоторые функции с числом аргументов до 9.

Сигналы H1, DIN, SR, EC являются для ФБ входными, они подаются на его внутренние схемы через группу из четырех мультиплексоров MUX "4-1" (на рисунке не показаны), к которым подключены 4 линии внешней шины управления C1...C4. Это позволяет распределять сигналы H1, DIN, SR и EC по линиям C1...C4 в любом желаемом варианте.

Мультиплексоры 3...6 направляют те или иные сигналы данных управления на триггеры 1 и 2. Триггеры могут использоваться для фиксации и хранения выходных сигналов функциональных преобразователей или же работать независимо от них. Входной сигнал ФБ DIN может быть прямым входом для любого триггера. Сигнал H1 тоже можно передавать любому триггеру, но через ФП Н, что вносит в цепь его передачи некоторую задержку.

Оба триггера имеют общие входы CK тактирования от сигнала K, разрешения тактирования EC и установки/сброса SR. Внутренние программируемые цепи в схеме триггера (на рис. 1.11 не показаны) позволяют индивидуально программировать полярность тактирующего сигнала CK. Сигнал EC синхронизирован с сигналом CK, сигнал SR асинхронный и для каждого триггера с использованием внутренних цепей триггера программируется как сигнал установки или сброса. Этот сигнал определяет состояние, в котором окажется триггер после процесса конфигурации микросхемы. Конфигурация определяет и характер воздействия на триггеры импульсов GSR (Global SR) и SR при работе схемы.

1.3.3. Блоки ввода/вывода FPGA

Блоки ввода/вывода, показанные на примере микросхем семейства Spartan фирмы Xilinx (рис. 1.12), обеспечивают интерфейс между выводами корпуса FPGA и ее внутренними логическими схемами. Каждому выводу корпуса придается блок ввода/вывода БВВ, который может быть конфигурирован как вход, выход или двунаправленный вывод.

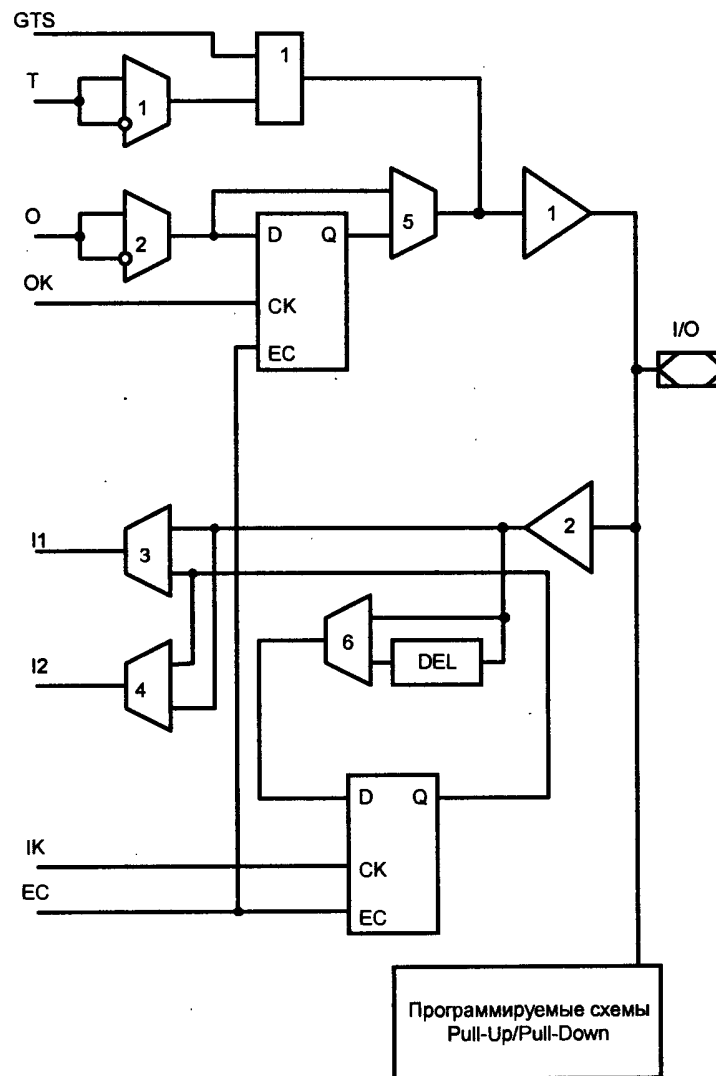


Рис. 1.12. Пример схемы блока ввода/вывода FPGA

Работа БВВ как выходного блока обслуживается следующими элементами: выходным буфером 1, триггером 1, мультиплексорами 1, 2, 5 и логической схемой ИЛИ. Выводимый сигнал O можно получать в прямой или инверсной форме в зависимости от программирования мультиплексора 2. Этот сигнал может передаваться на выходной буфер непосредственно или сниматься с триггера при соответствующем программировании мультиплексора.

ра 5. Сигналы Т и GTS (Global Tri-State), согласно логике ИЛИ, управляют переводом буфера в третье состояние, причем активный уровень сигнала Т программируется с помощью мультиплексора 1. Внутренние программируемые цепи триггера (на рисунке не показаны) позволяют изменять полярность тактирующего фронта. Сам буфер имеет программируемые крутизну фронта выходного сигнала и его уровни (КМОП/ТТЛ). Крутизна фронтов в некритичных к скорости передачи цепях снижается для уменьшения уровня помех на шинах питания и земли. Используется так называемый мягкий старт (Soft Start-Up), снижающий помехи при конфигурировании схемы и переходе ее к рабочему режиму, когда одновременно активизируются многие буферы. Первая активизация автоматически происходит с пологими фронтами перепадов напряжения. Затем вступает в силу заданный выбор той или иной крутизны фронтов в зависимости от принятой конфигурации БВВ.

Тракт ввода сигналов содержит входной буфер 2, триггер 2, программируемые мультиплексоры 3, 4, 6, элемент задержки ЭЗ и программируемые схемы задания определенных потенциалов выводу, к которому не подключен вводимый или выводимый сигнал (схемы Pull-Up/Pull-Down). Вводимый сигнал в зависимости от программирования мультиплексоров 3 и 4 или поступает непосредственно в систему коммутации FPGA по входным линиям I1 и I2, или же фиксируется триггером и с его выхода передается в эти линии. Триггеры могут конфигурироваться как тактируемые фронтом или как защелки (D-триггеры, управляемые уровнем). Выбор осуществляется присвоением триггеру соответствующего библиотечного символа. В цепи передачи сигнала на триггер 2 могут быть включены элементы задержки (при передаче сигнала через нижний вход мультиплексора 6). Включение задержки гарантирует необходимые временные соотношения между входными сигналами триггера D и глобальным сигналом тактирования.

Входной буфер может конфигурироваться для восприятия входных сигналов с пороговым значением ТТЛ (1,2 В) или КМОП (0,5 U_{cc}). Выходные уровни тоже конфигурируются, две глобальные регулировки входных порогов и выходных уровней независимы.

1.3.4. Системы межсоединений

Для систем межсоединений FPGA по изложенным в *разд. 1.2.2* причинам организация непрерывных соединений затруднительна. Для них характерны сегментированные линии связей, составленные из отдельных проводящих отрезков — сегментов. Сегменты соединяются в нужную цепь с помощью программируемых ключей. Выбор длины сегментов должен учитывать, что короткие сегменты затрудняют передачу сигналов на большие расстояния (в длинных связях появится много программируемых ключей, а вместе с ними задержек сигналов и затрат площади кристалла), а длинные сегменты

неудобны для коротких связей. Поэтому система межсоединений имеет, как правило, *иерархический характер*, и в ней сочетаются различные типы сегментов (основные связи, связи двойной длины, прямые связи для близлежащих функциональных блоков, длинные линии, пересекающие кристалл по всей его длине или ширине).

Систему межсоединений FPGA образуют сегментированные линии и переключательные блоки ПБ (PSM, Programmable Switching Matrix). Функциональные блоки имеют квадратные геометрические очертания, их выводы распределены по всем сторонам квадрата для облегчения коммутируемости. Для межсоединений функциональных блоков во внутренней области кристалла имеются три типа связей: одинарной длины, двойной длины и длинные линии. Упрощенная система коммутации FPGA показана на рис. 1.13, а на примере FPGA семейства XC4000 фирмы Xilinx.

На пересечениях вертикальных и горизонтальных каналов расположены переключательные блоки (рис. 1.13, б). В пределах ПБ пересекаются вертикальные и горизонтальные линии связей, и в каждом пересечении имеется цепь из 6 транзисторов для установления того или иного соединения. Сигнал, поступающий в ПБ по какой-либо линии (например, горизонтальной), может быть направлен вверх, вниз или прямо в зависимости от того, какой транзистор будет открыт при конфигурировании FPGA. Возможна и одновременная передача сигнала по нескольким направлениям, если требуется его разветвление.

Линии одинарной длины осуществляют, преимущественно, межсоединения соседних или близлежащих ФБ, линии двойной длины огибают переключательные блоки, соседние по отношению к данному, и проходят к следующим, чем облегчается установление более длинных связей. По три длинных линии, пересекающих весь кристалл по длине (ширине), реализуются сверху, снизу и по обоим бокам ФБ.

Выводы функциональных блоков пересекают горизонтальные и вертикальные каналы трассировки, проходящие непосредственно около них, и могут программируемыми точками связи подключаться к линиям каналов. Дальнейшее направление сигналов в нужные цепи осуществляется переключательными блоками.

Линии двойной длины сгруппированы в пары, имеется по 4 вертикальных и горизонтальных линии, обеспечивающих более быструю и эффективную передачу сигналов на средние расстояния. Длинные линии, рассчитанные на передачу сигналов на большие расстояния и при большой нагрузке, имеют в середине ключ, разделяющий линию на две части ((рис. 1.13, в).

Кроме системы коммутации для функциональных блоков FPGA (в частности, семейство Spartan) может иметь дополнительные трассировочные ресурсы, расположенные в виде кольца вне пределов матрицы ФБ. Эти ресурсы позволяют изменять назначение вводов/выводов микросхемы и облегча-

ют тем самым модификацию проекта, реализованного на FPGA, без влияния на разводку печатных плат, на которых монтируются микросхемы.

В число FPGA по принятой классификации попадают микросхемы с числом эквивалентных вентилях около 200—85 000, системными частотами приблизительно 50—80 МГц, числом пользовательских выводов 100—300. Лидером в производстве однократно программируемых FPGA считают фирму Actel, а в производстве FPGA с триггерной памятью конфигурации — фирму Xilinx.

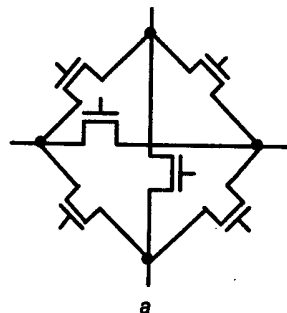
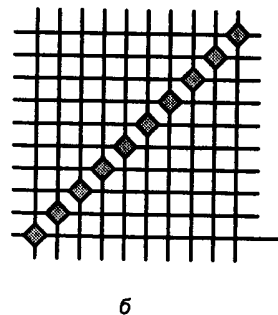
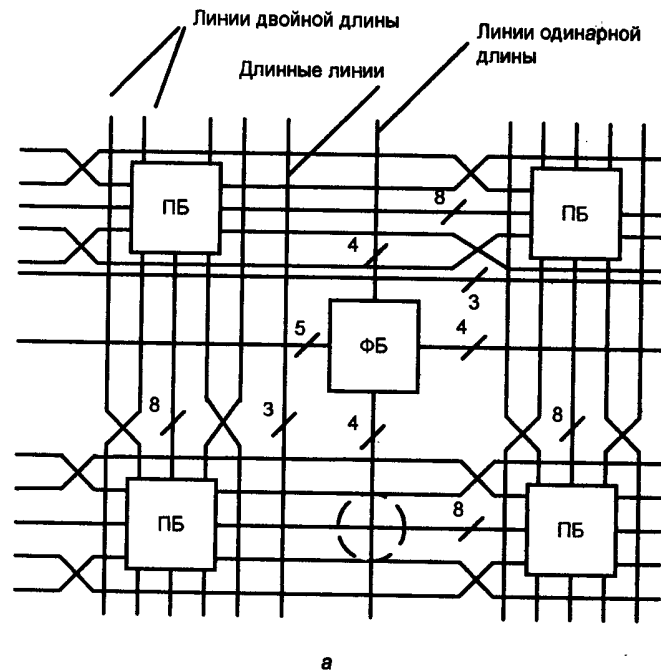


Рис. 1.13. Пример системы коммутации FPGA (а), схема переключательного блока (б) и схема для установления соединений коммутлируемых линий (в)

1.4. ПЛИС с комбинированной архитектурой

1.4.1. Структура микросхем семейства FLEX

По мере роста уровня интеграции ПЛИС их архитектуры усложнялись, появились архитектуры в той или иной мере сочетающие достоинства CPLD и FPGA. К числу появившихся микросхем с такой архитектурой в первую очередь можно отнести семейство FLEX (Flexible Logic Element matrix) фирмы Altera. Комбинированные архитектуры затем нашли применение в многочисленных БИС/СБИС, в том числе типа "система на кристалле". Микросхемы высшей сложности строят по все более оригинальным архитектурам, не только сочетающим черты CPLD и FPGA, но и обладающим новыми особенностями. На рис. 1.14 приведена структура микросхем семейства FLEX.

Микросхемы семейства FLEX имеют функциональные блоки (LABs, Logic Array Blocks) с логическими элементами ЛЭ (LEs, Logic Elements), содержащими функциональные преобразователи ФП табличного типа (LUTs). Функциональные блоки расположены в виде матрицы, между их строками и столбцами проходят горизонтальные и вертикальные трассировочные каналы, что характерно для FPGA. В то же время, трассы в каналах не сегментированы, а непрерывны, что типично для CPLD. Поскольку, как уже отмечалось, в схемах с большим числом функциональных блоков применение единой коммутационной матрицы затруднено, система коммутации имеет два уровня межсоединений — глобальный и локальный. Локальная программируемая матрица соединений (локальная ПМС или ЛПМС) обеспечивает межсоединения логических элементов ЛЭ, из которых составляются функциональные блоки LABs. В состав LAB входят 8 логических элементов. Соединения между блоками LAB обеспечиваются глобальной программируемой матрицей соединений ГПМС, к концам строк и столбцов которой подключаются блоки ввода/вывода (IOBs, Input/Output Blocks).

Начиная с семейства FLEX10K, в составе многих микросхем ПЛИС появились встроенные блоки памяти ВБП (EABs, Embedded Array Blocks). Ранее имелась возможность использования в качестве субмодулей памяти лишь тех ресурсов, которые имеются в ФП типа LUT.

В схеме на рис. 1.14 показан включенный в середине строк встроенный блок памяти (у большинства микросхем семейства FLEX10K его емкость равна 2048 бит). Такой блок может конфигурироваться как ЗУ с организацией 256×8, или 512×4, или 1024×2, или 2048×1 и использоваться не только для хранения данных, но и как табличный ФП для реализации сложных функций с числом аргументов 8—10 (в частности, на блоках EAB строятся быстродействующие арифметико-логические устройства АЛУ, перемножители 4×4 и т. д.).

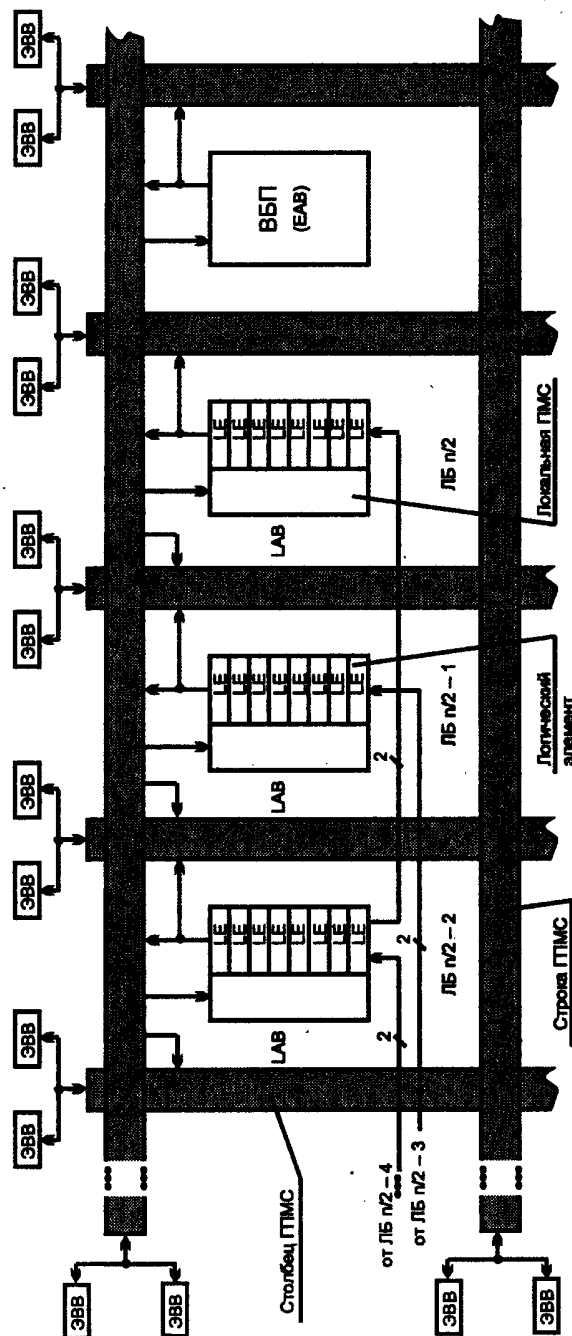


Рис. 1.14. Структура микросхем семейства FLEX

1.4.2. Логические элементы

Логические элементы микросхем семейства FLEX (рис. 1.15) имеют в своей основе 4-входные ФП табличного типа (LUT). Особенностью схем, которые могут быть построены из этих логических элементов, является наличие специальных трактов переноса, образуемых цепочками схем переноса СП, и трактов каскадирования, образуемых схемами каскадирования СК с непосредственными и быстродействующими связями между логическими элементами по указанным трактам.

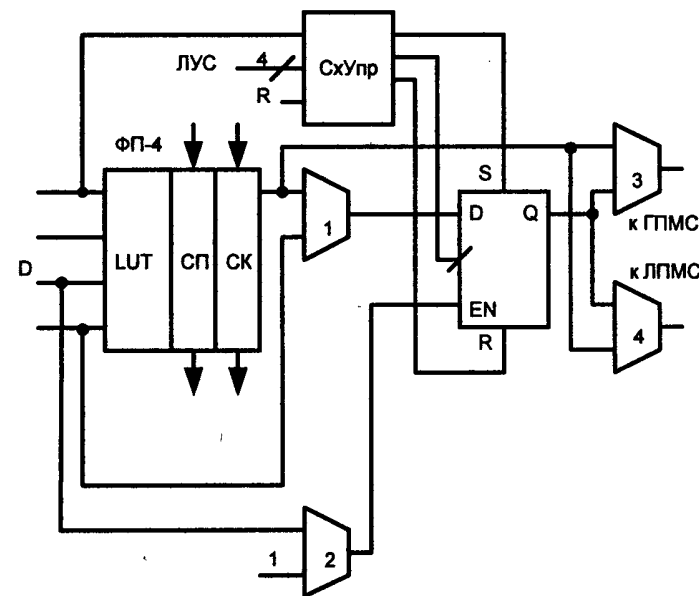


Рис. 1.15. Схема логического элемента микросхем семейства FLEX

ФП с 4 входами имеет 16 бит памяти и для воспроизведения функций 4 аргументов организуется в варианте 16×1 . Те же самые 16 бит можно использовать в виде двух табличных ФП с организацией 8×1 , реализующих две функции 3 переменных, что отвечает, например, потребностям построения разрядных схем сумматоров с последовательным переносом, разрядных схем некоторых счетчиков и т. д. Длинные цепочки переносов формируются в пределах нескольких LAB. В микросхемах семейства FLEX задержка цепи переноса мала (приблизительно 1 нс), что делает целесообразным применение в проектах многих простых схем с последовательными переносами даже для быстродействующих устройств.

Цепочка каскадирования используется для получения функций с числом аргументов более 4. Три соседних ЛЭ можно применить для воспроизведе-

ния частичных функций, а затем с помощью каскадирования сформировать из этих функций окончательный результат (рис. 1.16, а). При получении из частичных функций единой функции многих переменных частичные функции смежных ЛЭ объединяются любой логической операцией над двумя переменными, кроме сложения по модулю 2 и функции равнозначности (на рис. 1.16, а функции обозначены условными значками).

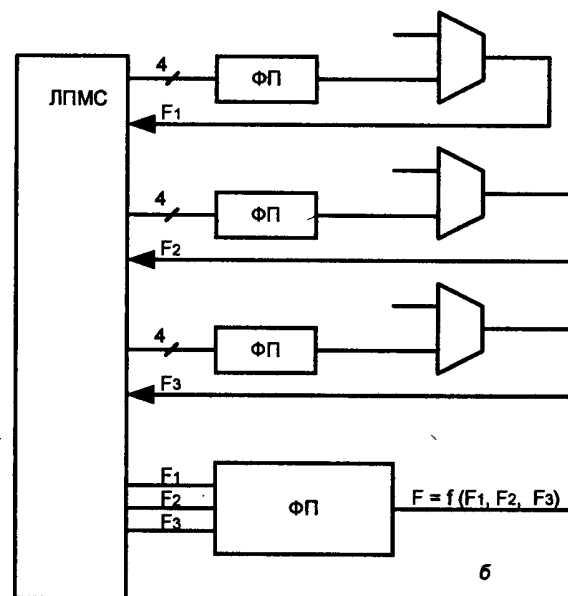
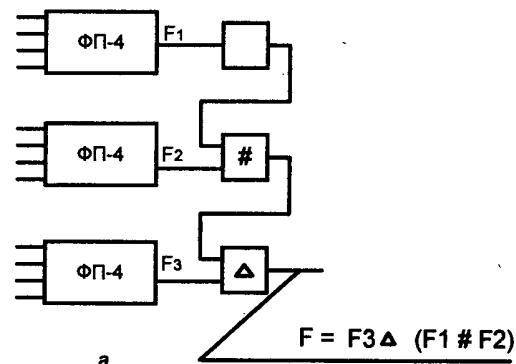


Рис. 1.16. Способы воспроизведения функций многих переменных методами каскадирования (а) и декомпозиции (б)

Функции многих переменных можно получить и другим способом, используя обратные связи. При этом сначала вырабатывается некоторая функция

четырех переменных, затем она вводится в качестве одного из входов в другой ЛЭ и т. д. В результате вычисляется "функция от функций" с числом аргументов, превышающим 4 (рис. 1.16, б).

Синхронный триггер в схеме на рис. 1.15 получает сигналы от схемы управления с входами R (Reset), D (одна из входных линий логического элемента) и ЛУС (четыре линии локальных управляющих сигналов). Один из двух сигналов ЛУС по выбору используется для тактирования триггера, два других вместе с сигналом R и одним из входов D управляют режимами сброса/установки. С помощью программирования можно задавать несколько режимов воздействия на триггер по входам S и R. Все эти режимы (сброс, установка, загрузка в разных вариантах) асинхронны.

Триггер может быть использован для фиксации значений выработанных ФП функций или как отдельный элемент с входом от одной из линий D в зависимости от программирования мультиплексора в линии этого входа.

Выходной сигнал логического элемента через программируемые мультиплексоры может подаваться в глобальную и локальную матрицы межсоединений в программируемых вариантах комбинационного (с обходом триггера) или регистрового (с триггера) выходов.

1.4.3. Встроенные блоки памяти

В состав СБИС семейства FLEX10K были впервые включены встроенные блоки памяти ВБП общей емкостью приблизительно от 6 до 20 Кбит для разных представителей семейства. Отдельные блоки емкостью 2 Кбит были размещены в середине каждой строки матрицы логических блоков. Блоки встроенной памяти можно использовать как по прямому назначению, т. е. как статическое ЗУ, так и для реализации ПЗУ и логических схем (табличных ФП повышенной размерности путем эмуляции ПЗУ с помощью загрузки таблицы в ОЗУ). Такие ФП дают более эффективные решения в сравнении с реализациями сложных функций средствами типовых логических блоков. Например, один встроенный блок памяти при организации 256×8 реализует перемножитель 4×4, способный работать на частотах до 50 МГц. Построение такого же перемножителя на типовых ЛБ потребовало бы занять 8 логических блоков, а частота работы перемножителя не превысила бы 20 МГц.

Блоки встроенной памяти ориентированы также на организацию буферов FIFO, а в микросхемах FLEX10KE и на построение двухпортовой памяти. Несколько блоков можно объединять для создания более емкой памяти. Так как блоки памяти расположены на том же кристалле, что и логическая часть схемы, работа с памятью отличается высоким быстродействием.

В структуре встроенных блоков памяти (рис. 1.17) кроме модуля памяти RAM/ROM имеется несколько синхронных D-триггеров и программируе-

мых мультиплексоров. Локальная программируемая матрица соединений ЛПМС получает 22—26 сигналов от строки глобальной матрицы ГПМС. Регистры 1 и 2 программируются для передачи в модуль памяти данных и адресов разной разрядности в зависимости от заданной конфигурации памяти. В блоке с емкостью 2 Кбит разрядность данных может изменяться от 1 до 8, а разрядность адреса от 11 до 8. Запись в память в зависимости от программирования мультиплексоров 4—6 может быть синхронной (от регистров по сигналам тактирования) или асинхронной (непосредственно от ЛПМС).

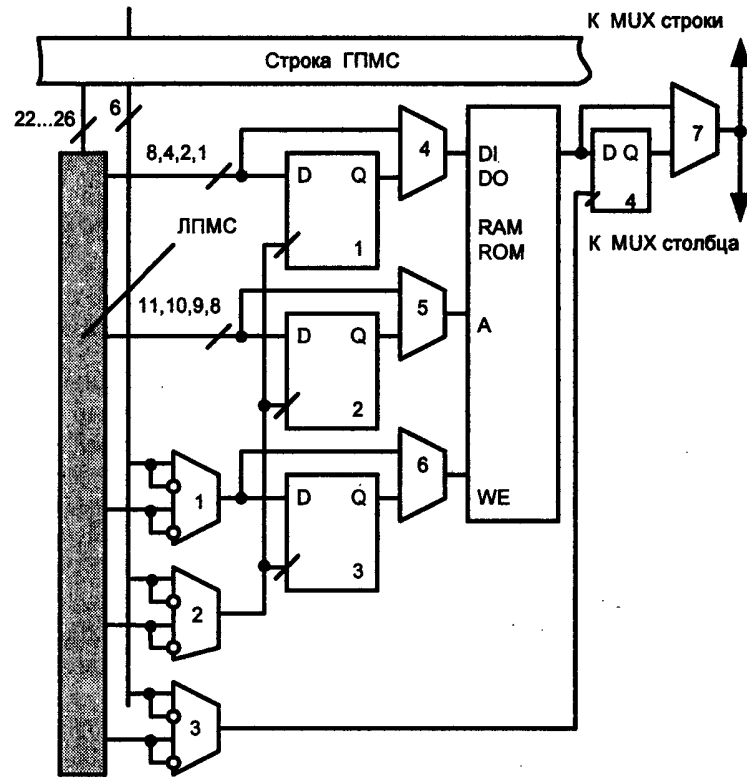


Рис. 1.17. Структура встроенных блоков памяти в микросхемах семейства FLEX10K

Сигналы управления регистрами 1—3 поступают от глобальной шины управляющих сигналов с возможностью выбора их полярности (мультиплексоры 1—3). Выходные сигналы блока памяти с помощью мультиплексоров 7—9 могут передаваться как на линии строки, так и на линии столбца ГПМС в тактируемом или асинхронном вариантах.

1.5. Основные сведения о СБИС типа "система на кристалле"

Уменьшение литографических размеров и другие усовершенствования процессов производства интегральных схем ведут к непрерывному росту уровня их интеграции, который для наиболее сложных СБИС в настоящее время оценивается уже десятками миллионов вентилей. Быстродействие таких СБИС характеризуется рабочими частотами в несколько сотен мегагерц. На кристаллах с подобными параметрами можно разместить законченную систему обработки данных, обладающую к тому же высокой производительностью. В последние годы появились микросхемы, называемые "система на кристалле", "система на программируемом кристалле", "программируемая система на кристалле" и т. д. (обозначаются как SOC, System On Chip, или как SOPC, System On Programmable Chip, или как PSOC, Programmable System On Chip, или как CSOC, Configurable System On Chip и т. д.). Далее в качестве обобщающего термина будет применяться обозначение SOPC.

Стратегическая значимость возможности создания законченной системы на одном кристалле очевидна — такая система обладает высокими качественными показателями. Для подавляющего большинства разработчиков создание системы на одном кристалле осуществимо только на основе программируемых СБИС, т. к. заказное проектирование по экономическим соображениям возможно только при массовом производстве.

Задача создания высококачественных законченных систем на одном кристалле решается более чем десятком крупных фирм в обстановке активной состязательности. Состязательность выражается как в конкурентном развитии двух направлений, отраженных в принятой нами классификации СБИС программируемой логики, так и в разработке многих разновидностей СБИС в рамках каждого из этих направлений.

Представителями первого направления являются SOPC с однородной структурой (с конфигурируемостью всех областей кристалла, с полностью синтезируемыми блоками).

Заметим, что однородность здесь понимается только как признак конфигурируемости всех областей кристалла, поскольку эти области содержат однотипные программируемые схемные ресурсы. Исключение составляют лишь некоторые области кристалла (схемы встроенной памяти, которые стали появляться в архитектурах ПЛИС, начиная с семейства FLEX10K, схемы управления тактовыми импульсами типа PLL или DLL и т. п.). В однородных SOPC реализуемые блоки могут размещаться в разных областях и создаваться соответствующим программированием. По своим архитектурным признакам SOPC, как правило, относятся к структурам комбинированного типа, в которых сочетаются признаки CPLD и FPGA с преобладанием при-

знаков FPGA. В иностранной литературе применительно к однородным в указанном смысле SOPC иногда используют термин generic.

SOPC блочного типа включают в себя как программируемые, так и фиксированные области, в которых жестко реализованы блоки с предопределенными функциями — аппаратные ядра (Hardcores, hard-ядра). В других областях кристалла размещается программируемая пользователем часть, чаще всего типа FPGA.

В ходе развития блочных SOPC характер и сложность аппаратных ядер изменяются. Вначале аппаратные ядра были относительно простыми, сейчас основным ядром сложных блочных SOPC нередко служит микропроцессор или микроконтроллер.

В чем состоят преимущества и недостатки СБИС двух развивающихся направлений? В однородных SOPC уровень интеграции уже позволяет сконфигурировать на кристалле области процессора, памяти и периферийных схем. Современные средства САПР с их приспособленностью к взаимозаменяемым и стандартным решениям позволяют объединять на одном кристалле виртуальные компоненты (soft-ядра, IP) разных разработчиков. Правда, при использовании IP возникает проблема их приобретения, поскольку стоят они достаточно дорого. В однородных SOPC не достигаются предельные быстродействия ядер.

Hard-ядра реализуют блоки, полученные методами проектирования заказных схем. Такие блоки в сравнении с их soft-аналогами занимают на кристалле значительно меньшую площадь (в несколько раз), поскольку они не содержат средств конфигурирования и оптимизированы для выполнения заданной конкретной функции. Если принять площадь, занимаемую цифровым устройством, реализованным по методу "на стандартных ячейках", за единицу, то при реализации того же устройства в технологии LPGA (на БМК с лазерным программированием) площадь составит в среднем 3,3 единицы, для технологии MPGA — 1,6 единиц, а для ПЛИС — 10—20 единиц. По этим же причинам существенно (на 20—50%) возрастает быстродействие hard-ядер в сравнении с soft-ядрами.

В то же время предопределенность функций hard-ядер снижает универсальность микросхемы (как говорят, уменьшает ее функциональную гибкость) и может сузить круг ее потребителей, что с точки зрения экономики является негативным фактором. Hard-ядра фиксированы на площади кристалла, что может затруднять решение задач размещения и трассировки для конфигурируемых областей микросхемы, препятствуя тем самым реализации максимальных показателей логической емкости и быстродействия для программируемых ресурсов схемы.

Во избежание больших потерь универсальности СБИС с hard-ядрами, для этих ядер отбираются только такие функциональные блоки, которые занимают значительную долю рынка. Характерными примерами hard-ядер, по-

мимо микропроцессоров и микроконтроллеров, могут служить блоки для реализации интерфейсов различных шин (в том числе шин PCI, VME), схемы поддержки интерфейса JTAG, множительные устройства для систем цифровой обработки сигналов.

Заметим, что современные ПЛИС с hard-ядрами имеют архитектурную преемственность с прошлыми разработками. Например, уже несколько лет в состав микропроцессорных систем вводят FPGA, создавая тем самым как бы "островки программируемости" среди блоков жесткой структуры. С другой стороны, в составе ПЛИС тоже несколько лет назад стали появляться несложные аппаратные ядра, т. е. участки фиксированной структуры в окружении массива синтезируемой логики. Поэтому говорить о том, что во что встроено, порою можно по-разному. *Чем выше процент синтезируемой части микросхемы, тем больший контроль над реализацией получает разработчик проекта, но тем больше блоков при этом теряют оптимальность своих параметров.*

По поводу перспектив применения двух разновидностей SOPC высказываются разные мнения. Одни считают, что структуры однородного типа с их высокой степенью регулярности схем легче переводятся на новый технологический уровень, поэтому освоение новых технологий для блочных структур должно всегда несколько запаздывать, а замедленный выход продукции на рынок наносит изготовителю экономический ущерб. С другой стороны отмечается, что аппаратные ядра не только гарантируют улучшенные параметры блоков, но и упрощают деловую часть разработки проекта, устраняя для проектировщика взаимоотношения с третьими лицами (поставщиками IP). При этом отпадают лицензионные проблемы, платежи роялти и т. п., и вся техническая поддержка сосредотачивается в руках одного партнера — поставщика микросхем, а все затраты концентрируются в одном показателе (цене микросхемы). Так или иначе, *но при решении вопроса о выборе той или иной SOPC приходится учитывать целый комплекс показателей.*

СБИС программируемой логики образуют динамичный, быстро растущий сектор рынка. Интеграция все большего числа функциональных блоков в одном кристалле сопровождается сокращением площади плат при реализации проектов, повышением быстродействия устройств и систем и их надежности, уменьшением потребляемой мощности и стоимости. Возможность быстрой реконфигурации схем непосредственно в работающем устройстве открывает перспективы не только эффективной отработки прототипа проекта, но и *создания принципиально новых структур с динамическим реконфигурированием и многофункциональным использованием аппаратных средств.*

При обсуждении вопросов применения SOPC возникает проблема выбора между однородными и блочными архитектурами, т. е. между soft- и hard-ядрами. Как отмечалось, в этой сфере имеются различные мнения. *Наиболее бесспорной областью использования блочных SOPC являются системы высшего*

быстродействия, поскольку, как бы соблазнительно ни выглядели рекламные преувеличения, процессоры с рабочими частотами в сотни мегагерц, осуществимые для hard-ядер, находятся за пределами возможностей soft-ядер FPGA. В других, менее очевидных ситуациях, оценивается весь комплекс характеристик SOPC.

В настоящее время развиваются обе разновидности SOPC, и успехи видны в обоих направлениях. Например, для SOPC типа generic фирмой Altera применено процессорное soft-ядро Nios, размещаемое на кристаллах семейств APEX20K/KE и APEXII. Уровень интеграции этих кристаллов настолько велик, что процессор занимает лишь малую долю их логических ресурсов, что считается вполне приемлемым. Фирма Xilinx разработала soft-ядро процессора Microblaze для своего семейства Virtex II. Это ядро обладает наивысшей для soft-ядер производительностью и работает на частотах до 125 МГц.

Впечатляют успехи разработок hard-ядер процессоров. Эти ядра базируются на RISC-архитектурах процессоров фирм ARM Limited, MIPS Technologies и IBM Microelectronics, работают на частотах до 200 МГц и более, имеют 5—6-ступенчатые конвейеры, выполняют скалярные операции, совершают большинство операций за один такт, занимают на кристалле площадь 2—3 мм² и ориентированы на малое потребление мощности. Стандартность архитектур таких процессоров позволяет пользоваться при их применении обширными инструментами и средствами проектирования, что снижает трудоемкость разработок и уменьшает время выхода продукции на рынок.

1.6. СБИС типа "система на кристалле" с однородной структурой (с полностью синтезируемыми блоками)

Однородные SOPC появились в составе продукции фирм Altera (семейство APEX20K), Xilinx (семейство Virtex), Actel (семейство proASIC), Cypress Semiconductor (семейство Delta 39K), Lucent Technologies (семейство ORCA4) и др. Появившиеся семейства, как правило, подвергаются периодическим модификациям с существенным улучшением параметров по мере освоения новых технологических процессов.

1.6.1. Микросхемы семейств APEX20K/KE, APEXII

Микросхемы семейств APEX20K/KE принадлежат к первым промышленным SOPC с конфигурируемостью всех областей кристалла. Память конфигурации — статическая (триггерная). Микросхемы APEX20K имеют напряжение питания ядра 2,5 В и уровни сигналов ввода/вывода 2,5; 3,3 и 5,0 В, а микросхемы APEX20KE напряжение питания ядра 1,8 В, а уровни сигналов ввода/вывода 1,8; 2,5; 3,3 и 5,0 В.

Общий план микросхем показан на рис. 1.18. Архитектура микросхем называется Multicore, в ней сочетаются функциональные блоки типов LUT и PAL. Кроме того, микросхемы имеют встроенные блоки памяти ESB (Embedded System Blocks), содержащие также средства для реализации ДНФ логических функций (что и объясняет термин "системные" в названии блоков).

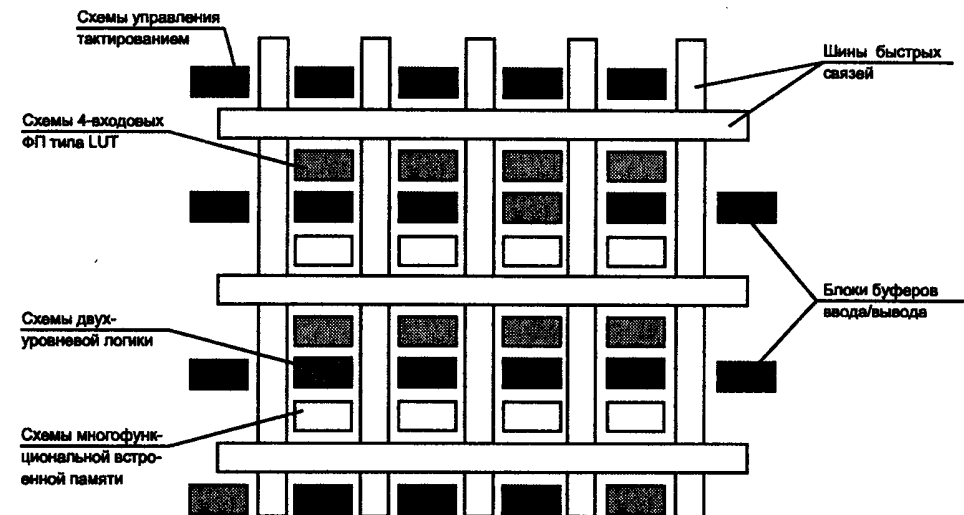


Рис. 1.18. Общий план кристалла микросхем семейства APEX20K/KE

LUT-блоки этого семейства заимствованы от описанных ранее микросхем FLEX10K, блоки типа SOP (Sum-of-Products) — от семейства MAX7000, также описанного выше, блоки памяти близки по структуре к блокам семейства FLEX10KE с емкостью 2 Кбит.

Дизъюнктивные нормальные формы удобны для получения функций управляющей логики, в частности, адресных декодеров и автоматов с памятью. Табличные блоки хорошо приспособлены к реализации сложных алгоритмов обработки сигналов и к построению узлов и устройств, содержащих большие количества триггеров. Совмещение обоих типов блоков в одном кристалле обеспечивает эффективность схемных решений, характерных для проектов разных типов. Этому же способствует наличие на кристалле встроенных блоков памяти высокого быстродействия, организуемых в виде RAM, двухпортовой RAM, FIFO с широким диапазоном программируемых длины и ширины, а также блоков ассоциативной памяти (у микросхем APEX20KE).

Сложность кристаллов семейства APEX обусловила увеличение в них числа иерархических структурных уровней. Наименьшей структурной единицей логического типа является логический элемент ЛЭ (LE, Logic Element), по-

добный логическому элементу семейства FLEX10K/KE (см. рис. 1.15), допускающий, в зависимости от решаемых задач, несколько стандартных конфигураций (работу в нормальном, арифметическом или счетном режиме) и создание цепей переноса и каскадирования (рис. 1.19). В нормальном режиме воспроизводятся большинство логических функций и, при каскадировании логических элементов, дополнительные функции, в частности дешифраторы с большим числом входов. В этом режиме LUT-блоки работают

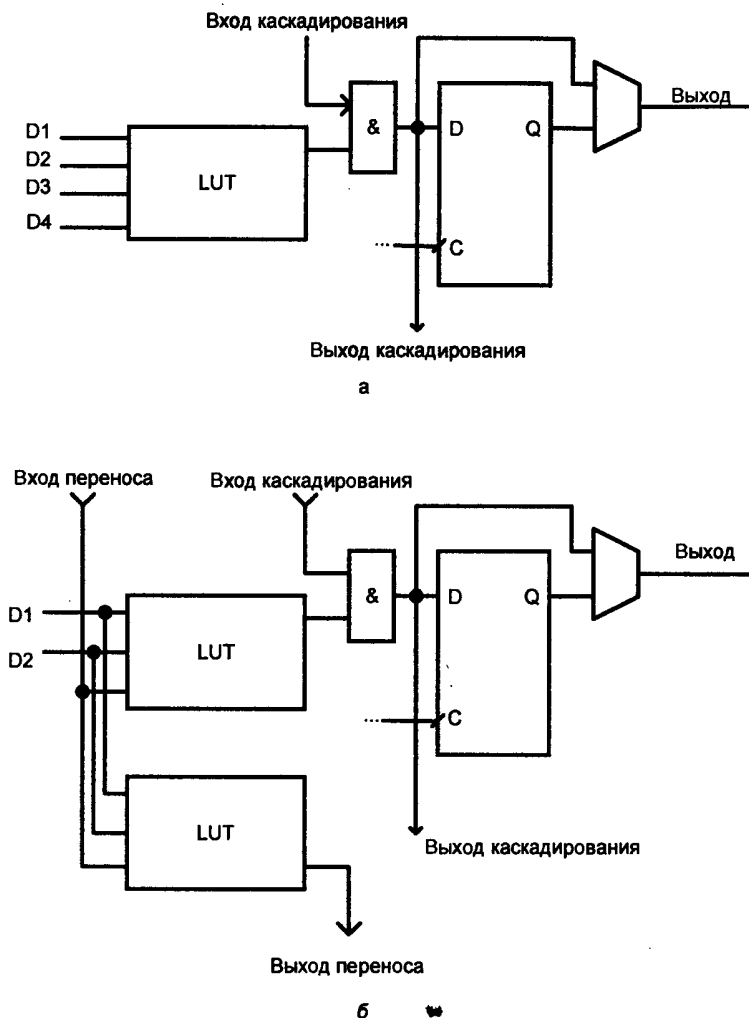


Рис. 1.19. Пояснения к способам конфигурации логических элементов семейств APEX и FLEX (а — нормальный режим, б — арифметический)

как функциональные преобразователи с четырьмя входами. В арифметическом режиме LUT-блок конфигурируется как два трехвходовых функциональных преобразователя, один из которых служит для выработки функций переносов, а другой для функций трех аргументов, реализуемых в разрядных схемах устройства. В счетном режиме (на рисунке не показан) к схеме, соответствующей арифметическому режиму, добавляются элементы, обеспечивающие функции разрешения счета, реверса, сброса и загрузки счетчика.

Десять логических элементов объединяются в логический блок ЛБ (LAB, Logic Array Block), имеющий свой уровень межсоединений — локальную матрицу соединений ЛМС (LI, Local Interconnect). Шестнадцать логических блоков и блок ESB объединяются в мегаблок (MegaLAB). Мегаблоки также имеют свой уровень межсоединений, что не только увеличивает общие ресурсы трассировки кристалла, но и позволяет получать в пределах мегаблока некоторые функционально законченные части системы. Функциональная автономность мегаблоков, когда она возможна, упрощает модификацию системы и закладывает возможности локальной оптимизации схемных решений. Мегаблоки коммутируются между собой по системе глобальной матрицы соединений ГМС (Fast Track Interconnect), линии которой непрерывны и проходят по всей длине (ширине) кристалла. К концам линий ГМС подключаются элементы ввода/вывода ЭВВ (IOE, Input/Output Elements). К системе межсоединений относятся также цепи переноса и каскадирования.

Схемы ввода/вывода отличаются высоким быстродействием и поддерживают более 10 стандартов входных и выходных сигналов. Контакты ввода/вывода разделены на восемь банков, имеющих свои напряжения питания и способных, независимо от других банков, поддерживать те или иные стандарты ввода/вывода. Элементы ввода/вывода обеспечивают совместимость с шиной PCI (64 разряда, 66 МГц).

Большое внимание уделено схемам тактирования кристалла, реализована схема передачи синхросигналов с малыми фазовыми сдвигами, блоки микросхем могут использовать до 8 глобальных синхросигналов. Схемы управления параметрами синхросигналов выполняют функции минимизации их задержек и фазовых рассогласований (ClockLock), умножения частоты в некоторых блоках схемы, что позволяет разводить по кристаллу сигналы тактирования меньшей частоты (функция ClockBoost), функции ClockShift (введения специальных программируемых фазовых сдвигов в линии передачи сигналов).

Микросхемы семейства APEX20K/KE имеют программируемое управление по координатам скорость/мощность. Когда максимальное быстродействие не требуется, можно снижать мощность потребления до 56% с помощью соответствующей опции турбо-бита для блоков ESB. Неиспользуемые блоки ставятся в режим глубокого снижения мощности.

Типичный диапазон числа эквивалентных вентилей для микросхем семейства APEX20K/KE от 60 тыс. до 1,5 млн., емкость встроенной памяти составляет от 32 768 до 442 368 бит, число пользовательских выводов от 204 до 808,

внутренняя рабочая частота достигает значений приблизительно 200 МГц. В новом семействе APEx20Kc с медными проводниками и топологической нормой 0,15 мкм быстродействие схем повышается на 25—35%.

Для микросхем семейства APEx20K/KE разработано soft-ядро процессора Nios — RISC-процессора с изменяемой архитектурой, конфигурируемым файлом регистров, 16-разрядными командами и шиной данных на 16 или 32 разряда по выбору проектировщика. Производительность процессора может достигать 50 MIPS. Микросхемы семейства APEx20KE имеют настолько высокий уровень интеграции, что процессор Nios занимает небольшую долю их логической емкости. Например, для микросхемы EP20K200E ядро процессора Nios занимает 12% логических ресурсов кристалла. При стоимости кристалла около 80 долларов (в единичных поставках) на ядро процессора приходится приблизительно 10 долларов, а при массовых поставках стоимость снижается приблизительно в два раза. Для микросхемы EP20K1500E с 1,5 млн. вентилей доля расхода ресурсов на процессор Nios снижается до 1,5%. Изменяемая архитектура ядра Nios, как и других разработанных soft-ядер, придает SOPC высокую степень гибкости. Однако быстродействие soft-ядер не достигает максимальных значений и в сравнении с быстродействием аппаратных ядер остается умеренным.

Для ядра Nios применима разработанная ранее для семейства FLEX10K довольно обширная периферия: таймеры-счетчики, UART, широтно-импульсные модуляторы, контроллеры дисковой памяти, динамических ОЗУ и др. Применимы и LPM (библиотечные параметризуемые модули), использовавшиеся ранее в микросхемах FLEX10K.

Появление семейства APExII — новое значительное достижение в развитии однородных SOPC. На основе технологии с топологическими нормами 0,15 мкм при напряжении питания ядра 1,5 В и многослойной системе медных межсоединений получены кристаллы с уровнем интеграции до 4 млн. типичных эквивалентных вентилей (90К логических элементов), встроенной памятью в диапазоне от 425 984 до 1 532 712 бит и высокоскоростными каналами связей, в том числе дифференциальных. Большое число каналов связи с пропускной способностью в 1,6 Гбит/с и 624 Мбит/с позволяет получать на кристалле быстродействующую систему из множества блоков, либо включать кристалл в более сложную систему в качестве быстродействующего блока. В настоящее время выпускаются 5 представителей семейства APExII с диапазоном числа типичных эквивалентных вентилей от 600 тыс. до 4 млн.

1.6.2. Микросхемы семейств Virtex, Virtex E, Virtex II

Семейства Virtex и Virtex E с триггерной памятью конфигурации появились в составе продукции фирмы Xilinx в конце 1998 года. Как и микросхемы APEx20K, они имеют мегавентильный уровень интеграции и встроенные

блоки памяти. Системная частота их работы достигает 200 МГц. Напряжение питания ядра у схем Virtex 2,5 В, у схем Virtex E 1,8 В, для уровней внешних сигналов возможен выбор нескольких значений. Тенденция снижения питания ядра микросхемы при сохранении широкого выбора уровней внешних сигналов имеет общий характер. По мере снижения топологических норм, напряжения питания ядер, как правило, проходят следующую последовательность цифр: 3,3; 2,5; 1,8; 1,5 В, и это касается всех фирм-производителей БИС/СБИС.

Общий план кристалла семейств Virtex и Virtex E показан на рис. 1.20.

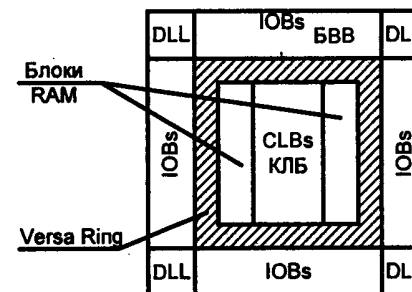


Рис. 1.20. Общий план кристалла микросхем семейств Virtex и Virtex E

Основной частью микросхем является матрица конфигурируемых логических блоков КЛБ (CLB, Configurable Logic Block), окруженная программируемыми блоками ввода/вывода БВВ (IOBs, Input/Output Blocks), и иерархическая система коммутации. КЛБ коммутируются системой локальных связей с генеральной матрицей соединений ГМС (GRM, General Routing Matrix) и горизонтальными и вертикальными каналами трассировки. Каждый КЛБ входит в так называемый VersaBlock (рис. 1.21), в котором имеются локальные ресурсы соединений для коммутации КЛБ с ГМС, встроенные обратные связи в пределах самих КЛБ и прямые пути соединений между горизонтально-смежными КЛБ. VersaBlock может реализовать небольшие части проекта, объединяемые далее средствами коммутации.

Микросхемы имеют также средства коммутации, называемые VersaRing, для создания межсоединений в периферийной области, программирование которых, в частности, облегчает размещение внешних выводов по входам/выходам схемы, реализованной во внутренней области и, тем самым, облегчает и адаптацию логического проекта к существующей разводке печатной платы.

Основная трассировочная нагрузка приходится на горизонтальные и вертикальные каналы основных связей, соответствующих строкам и столбцам матрицы КЛБ. На эти каналы сигналы из КЛБ выходят через переключательные матрицы ГМС, способные соединять линии горизонтальных и вер-

тикальных каналов. От ГСМ к ГСМ по всем четырем направлениям сигналы могут передаваться через линии разных типов. Имеются линии одинарной длины (между смежными ГСМ), линии протяженностью в 6 КЛБ с доступом к ним на концах и в середине и длинные линии, с протяженностью по всей высоте кристалла для вертикальных линий и по всей его ширине для горизонтальных.

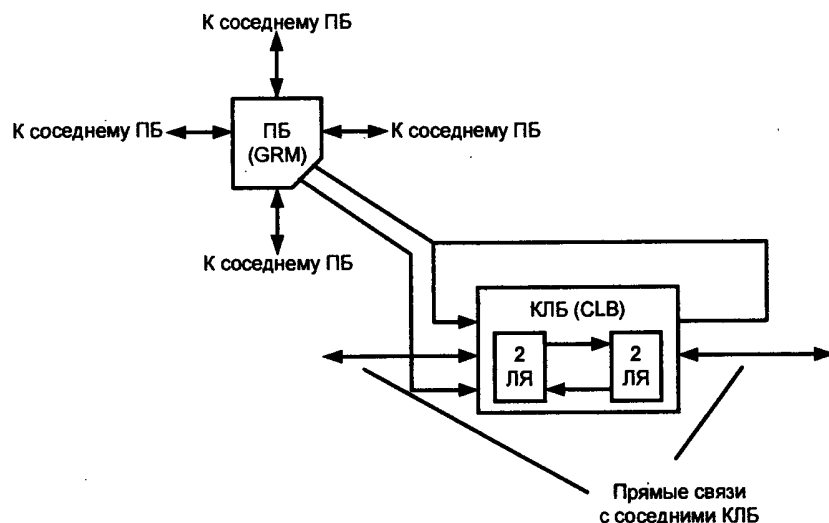


Рис. 1.21. Схема VersaBlock микросхем семейств Virtex и Virtex II

К матрицам ГСМ подключены также блоки памяти по 4 Кбит и 4 схемы DLL для управления фазовыми и частотными параметрами синхросигналов в разных областях кристалла.

Схемы DLL (Delay Locked Loop) или PLL (Phase Locked Loop), называемые в последнее время также схемами типа Clock Manager, стали важными блоками быстродействующих БИС/СБИС. Дело в том, что из-за задержек в цепях передачи синхросигналов по кристаллу они "разъезжаются" во времени, и в разных местах кристалла уже не являются синхронными. Понятно, что это создает проблемы взаимодействия элементов, питаемых такими несинхронными тактирующими сигналами. Схемы DLL и PLL представляют собою следящие системы с фазочувствительным элементом на входе. Фазочувствительный элемент сравнивает корректируемую синхропоследовательность с эталонной и, при наличии сигнала рассогласования этих последовательностей, изменяет фазу корректируемого синхросигнала, чтобы компенсировать задержки в цепях его передачи. Наличие в БИС/СБИС блоков DLL или PLL позволяет заметно повысить частоту синхросигнала по сравнению с той частотой, которая могла бы быть реализована в схеме без DLL или PLL.

Основа КЛБ — логическая ячейка ЛЯ (LC, Logic Cell). В каждом КЛБ размещены 4 логических ячейки, организованные в две идентичные секции (Slices). Логическая ячейка (рис. 1.22) по своей структуре подобна рассмотренным ранее типовым элементам FPGA и содержит 4-входовой LUT-блок с возможностью организации цепи переноса и триггер.

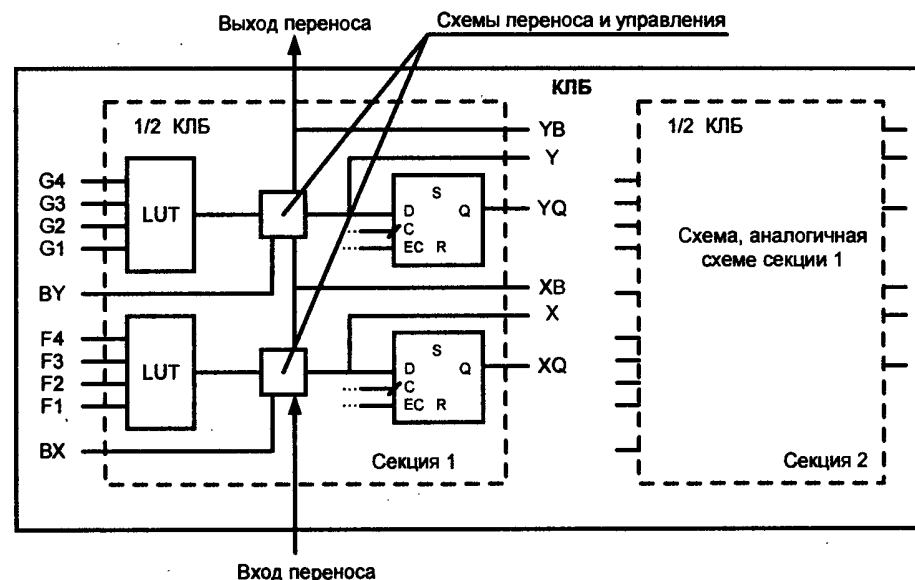


Рис. 1.22. Схема логической ячейки кристаллов Virtex и Virtex E

Выходная величина вырабатывается одновременно в двух вариантах — как комбинационная, снимаемая непосредственно с выхода LUT-блока, или как регистровая, снимаемая с выхода триггера. Дополнительно к 4 базовым логическим ячейкам КЛБ имеет логическую схему для выработки функций от 5 и 6 аргументов в виде комбинации выходов от логических ячеек. Ресурсы LUT-блоков могут использоваться и в качестве обычных RAM. Каждый LUT-блок дает 16 бит памяти с организацией 16×1. В секции может быть образован блок памяти с организацией 16×2 или 32×1 для обычного RAM или 16×1 для двухпортового. LUT-блок может быть сконфигурирован также как 16-разрядный регистр сдвига.

Триггер типа D может программироваться как защелка или как триггер, управляемый фронтом, и принимать данные с выхода LUT-блока или непосредственно от входов секций, и имеет программируемые входы установки и сброса с несколькими режимами.

Блоки встроенной памяти (BlockSelectRam+) топологически расположены в двух столбцах вдоль вертикальных краев внутренней области кристалла,

высота одного блока соответствует четырем КЛБ. Каждый блок представляет собой синхронную двухпортовую RAM емкостью 4 Кбит с независимыми управляющими сигналами для каждого порта (рис. 1.23).

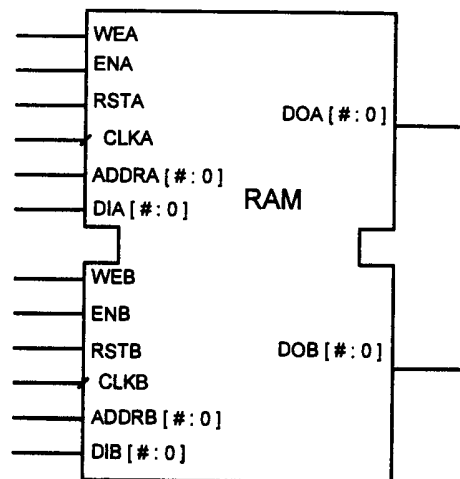


Рис. 1.23. Внешняя организация блока встроенной памяти микросхем семейств Virtex и Virtex E

Разрядности данных, передаваемых через порты, могут программироваться независимо, причем для примитивов, генерируемых библиотекой блоков SelectRam+, разрядности могут быть заданы в вариантах 1, 2, 4, 8, 16, чему соответствуют разрядности шины адреса 12, 11, 10, 9, 8. Блоки RAM имеют также специальные средства межсоединений для связей как с КЛБ, так и с другими блоками RAM.

Сигналы и шины портов А и В имеют следующий смысл:

- CLK — тактирующий сигнал;
- EN — сигнал разрешения, отсутствие которого сохраняет состояния выходов блока в прежнем состоянии, запись данных запрещена;
- WE — разрешение записи, при его активности сигналы с входной шины данных записываются в память по адресу с шины адресов, а вновь записанные данные появляются на выходной шине данных. Пассивный уровень сигнала задает операцию чтения по адресу с шины адресов, и содержимое адресованной ячейки появляется на выходной шине данных;
- RST — сбрасывает выходной регистр-защелку порта, не влияя на ячейки памяти и на операции записи в другом порту;
- ADDR #:0 — шина адреса, выбирает ячейку памяти для операций чтения или записи. Разрядность порта задает требуемую разрядность этой шины соответственно приведенным ранее вариантам конфигурации памяти;

- DI #:0 — входная шина данных, передает значения записываемых в память данных соответственно заданной разрядности порта;
- DO #:0 — выходная шина данных, отображает содержимое адресованной ячейки памяти. При записи отображает новые данные. Разрядность шины соответствует заданной разрядности порта.

Н-активность или L-активность сигналов CLK, EN, WE, RST могут быть заданы независимо для каждого порта. При работе памяти действуют правила разрешения конфликтов при одновременном обращении портов к одной и той же ячейке.

Новое семейство Virtex II должно значительно расширить возможности создания SOPC. Уровень интеграции доведен в нем до 10 млн. системных вентилей, число слоев межсоединений до 8, используются медные проводники, усовершенствованы конфигурируемые логические блоки. Активные цепи межсоединений увеличивают скорость распространения сигналов и предсказуемость их задержек. Микросхемы содержат до 3,5 Мбит памяти RAM/ROM, имеют умножители размерностью 18×18 бит, широкий набор интерфейсов ввода/вывода и 122 980 Кбит памяти в составе LUT-блоков. Существует возможность выбора внутрикристалльных резисторов-терминаторов с автоматическим согласованием их номиналов с внешними эталонными резисторами (Digitally Controlled Impedance Technology). Предусмотрена возможность удаления с кристалла областей soft-ядер и размещение на их местах соответствующих hard-ядер с сохранением всех имевшихся межсоединений и всех функциональных возможностей блоков.

Фирма Xilinx сообщила о разработке быстреего в мире soft-ядра процессора Microblaze, работающего на частоте до 125 МГц, имеющего 32-разрядные шины для команд и данных в Гарвардской архитектуре. Процессор предназначен для использования в сложных системах (сетевых, телекоммуникационных и т. п.) на основе семейства Virtex II. Для совместной работы с процессором Microblaze созданы также soft-ядра периферийных устройств (арбитров, UART, контроллеров прерываний, интерфейсных схем для связи с внешними 3У типов Flash и SRAM и т. д.).

1.7. СБИС типа "система на кристалле" с блочными структурами (с аппаратными ядрами)

1.7.1. Вводные замечания

Блочные SOPC — самое новое направление в развитии "систем на кристалле". Согласно оценкам специалистов, их реализация становится особенно эффективной при топологических нормах производства, достигнутых лишь

в последнее время. К первым продавцам блочных SOPC без процессорных ядер относятся фирмы Lucent Technologies (впоследствии Agere Systems), QuickLogic. Блочные SOPC с ядрами процессоров появились впервые в продукции фирм Triscend и Atmel, затем к ним присоединились фирмы Altera, Cypress, Xilinx и др.

Следует заметить, что между SOPC с конфигурируемостью всех областей кристалла и SOPC с аппаратными процессорными ядрами расположена *многочисленная группа микросхем с более простыми hard-ядрами*, начало которым положили кристаллы со встроенными ядрами интерфейса PCI. В этой книге основное внимание будет уделено SOPC с процессорными hard-ядрами, именно наличие встроенного аппаратного микропроцессора или микроконтроллера будем считать признаком истинной "системы на кристалле".

В составе блочных SOPC с процессорными ядрами можно выделить четыре основные части: процессор, FPGA, интерфейс между ними и интерфейс с внешней средой. В первую очередь, именно интерфейсы различаются друг от друга в SOPC разных фирм, и, главным образом, это определяет специфику тех или иных микросхем. Различия, имеющиеся во внутренней организации процессоров и FPGA, вторичны и при проектировании на основе SOPC проявляют себя в меньшей степени.

1.7.2. SOPC с блочной структурой, не содержащие ядер процессоров

Появившиеся в последние годы блочные SOPC с процессорными ядрами ознаменовали создание в полном смысле универсальных "систем на кристалле". Этим SOPC предшествовали "усеченные" варианты с более простыми аппаратными ядрами, ориентированные на специализированные приложения. Такие SOPC оказались полезными для многих, в том числе и самых новых, проектов. В силу этого существование "заполненной" зоны между БИС/СБИС с простейшими и самыми сложными ядрами обуславливается не только временными причинами технологического характера, но и устойчивыми потребностями системотехников. Кстати говоря, с точки зрения технологических возможностей (и связанных с ними параметров БИС/СБИС) считается, что реализация простых аппаратных ядер становится эффективной при топологической норме производства не более 0,25—0,35 мкм, а реализация сложных — при норме не более 0,18 мкм.

Блочные SOPC со специализированными ядрами выпускаются несколькими фирмами. К числу пионеров этого направления относятся фирмы Lucent Technologies, QuickLogic, Cirrus Logic, Cypress Semiconductor и др.

Фирмой Lucent Technologies первая микросхема OR3TP12 типа FPSC (Field-Programmable System Chip) выпущена в 1998 году на основе известной серии микросхем ORCA3. Схема содержит массив программируемой логики с мат-

рицей логических ячеек 18×18 емкостью до 60 тыс. эквивалентных вентиля и аппаратное ядро PCI (64 разряда/66 МГц). В следующей микросхеме OR3LP26B был удвоен объем программируемой логики FPGA и увеличена пропускная способность межсоединений тракта "ядро-FPGA", а в микросхеме ORT4622 ядро PCI заменено полностью дуплексным 4-канальным синхронным интерфейсом с пропускной способностью 622 Мбит/с. Последующий вариант ORT8850, основанный на разработках серии ORCA4, продолжил линию развития микросхемы ORT4622, имея 8 каналов и пропускную способность 850 Мбит/с. Продукция фирмы Lucent Technologies ориентирована на рынок интерфейсных схем для систем коммуникаций. Микросхемы этой фирмы отличаются высоким схемотехнологическим уровнем, в частности, тем, что имеют минимальный размер 0,13 мкм при 7 слоях металлизации.

Фирма QuickLogic разработала семейство блочных SOPC под названием ESP (Embedded Standard Products, в других источниках эта же аббревиатура расшифровывается как Embedded Standard Platforms) с подсемействами QuickRAM, QuickPCI, QuickPC, QuickDSP, QuickSD. По своим классификационным признакам семейство ESB принадлежит к классу ASSP. Конфигурируемые части указанных микросхем построены с использованием программируемых перемычек ViaLink типа antifuse, что придает этим частям известные свойства: однократность программирования, но при этом высокую компактность межсоединений, высокое быстродействие схем, повышенную стойкость к воздействиям температуры и радиации, защищенность от рассекречивания проекта.

Кристаллы QuickRAM (1998 г.) содержат FPGA и встроенные блоки памяти емкостью до 25 374 бит с программированием организации как RAM, ROM, FIFO при способности работать на частотах до 160 МГц.

Микросхемы QuickPCI имеют FPGA и ядро PCI (ядро PCI фиксировано и, тем самым, определяется как аппаратное, но, что своеобразно, также выполнено с использованием перемычек antifuse). Для ядер PCI возможен выбор вариантов на 32 и 64 разряда с частотами 33, 66 и 75 МГц. Для связи ядра PCI с другими частями схемы введен набор буферов FIFO.

Схемы QuickPC имеют аппаратно реализованный канал Fibre Channel со скоростью передачи данных до 2,5 Гбит/с и 32-разрядным синхронным системным интерфейсом с буферами FIFO.

Кристаллы QuickDSP базируются на специальных арифметических ячейках ECU (Embedded Computational Units), выполняющих однотактно операции умножения 8×8 с задержкой 4,53 нс, 16-разрядное сложение с задержкой 2,54 нс, умножение с накоплением с задержкой 7,07 нс. На основе таких ячеек строятся схемы реализации алгоритмов цифровой обработки сигналов.

В микросхемах QuickSD комбинируются быстродействующие FPGA и блоки SERDES (Serializer-Deserializer), выполняющие преобразования данных из

параллельной формы в последовательную и наоборот. Три представителя семейства имеют в своем составе 6 или 8 каналов SERDES и от 334 до 658K системных вентиляей. Каждый порт SERDES может передавать последовательные данные со скоростью до 1 Гбит/с. Две программируемые схемы синхронизации каналов SERDES могут тактировать передачи, если синхросигналы не содержатся в самих данных. Имеются два программируемых блока PLL и от 24 до 36 блоков двухпортовой SRAM, а также 12—18 блоков QMAC умножения-накопления, которые в большой степени ускоряют вычисления при цифровой обработке сигналов. Результирующая скорость передачи составляет 8 Гбит/с, чего достаточно для большого числа применений. Логика FPGA может выполнять операции кодирования и декодирования информации, балансировки дифференциальных сигналов по постоянному току, обрамления пакетов данных, управления памятью и др., в том числе реализацию некоторых интерфейсов.

Каждый канал SERDES может быть запрограммирован на выработку последовательных кодов для 1, 4, 7, 8, 10 или 20 линий параллельной шины, что позволяет блоку принимать сигналы с многоразрядных внутренних шин и расщеплять их для максимального использования ресурсов кристалла.

В качестве примера области применения микросхем QuickSD можно указать интерфейс с дисплеем высокого разрешения. Поток данных для такого дисплея обычно передается через многоразрядный кабель. При наличии скоростного последовательного канала кабель можно исключить и заменить более дешевым последовательным интерфейсом. При этом возможности микросхем QuickSD способны удовлетворить требования систем с самыми современными дисплеями.

Внутренние передачи "регистр-регистр" производятся на частотах до 600 МГц, частота передач "кристалл-кристалл" свыше 225 МГц.

Фирмой Cypress Semiconductor разработаны блочные SOPC семейства PSI (Programmable Serial Interface), содержащие блоки SERDES-SONET, канал Fibre Channel, гигабитный интерфейс сети Ethernet и др.

1.7.3. SOPC семейства FPSLIC фирмы Atmel

Начнем ознакомление с SOPC, имеющими ядра процессоров, с семейства FPSLIC (Field Programmable System-Level Integration Chip) фирмы Atmel, разработка которого была признана лучшим проектом США в области электроники в 1999 году.

До выпуска семейства FPSLIC (микросхем AT94K) фирмой Atmel уже производились как хорошо известные микроконтроллеры типа AVR, так и FPGA семейства AT40K, которые и были объединены в одном кристалле FPSLIC. Кроме них, к основным блокам этого кристалла относится статиче-

ская память SRAM. FPGA получает от процессора команды и исходные данные и передает ему выработанные результаты.

В архитектуре микросхем FPSLIC (рис. 1.24) представлены следующие блоки:

- ☐ процессорное ядро AVR и его периферия;
- ☐ память программ и память данных;
- ☐ FPGA AT40K с логической емкостью от 10 до 40K эквивалентных вентиляей.

Используется Гарвардская архитектура процессора, обеспечивающая повышение быстродействия ядра вследствие совмещения во времени процессов выборки и исполнения команд. В блоке FPGA размещена стандартная структура микросхем AT40K с различной логической емкостью у разных представителей семейства, способная работать на системных частотах до 100 МГц и имеющая собственные блоки встроенной памяти емкостью от 2048 до 18 432 бит.

Структура микросхем AT40K, в основных чертах, типична для классических FPGA, хотя и имеет некоторые оригинальные особенности. Квадратная область, занимаемая FPGA на кристалле, от края до края плотно заполнена идентичными логическими ячейками. Помимо этих ячеек в области FPGA реализованы повторители (Repeaters), сегментированные программируемые связи и блоки RAM. Повторители соединяют смежные сегменты в горизонтальных и вертикальных трассах передачи сигналов, размещенных через каждые четыре ячейки по обеим координатам. На пересечениях рядов, содержащих повторители, находятся блоки RAM емкостью 32×4 бита каждый. В трактах трассировки выделяются локальные линии связи и экспресс-линии. В экспресс-линиях повторители включены реже, чем в локальных, а длина их сегментов больше.

Особенностью ячеек FPGA является их восьмиугольная форма. У квадратных ячеек, если они в дополнение к другим ресурсам трассировки имеют прямые связи с соседними ячейками, четыре направления передач: вверх, вниз, влево, вправо (в американских фирменных справочниках направления часто соответствуют странам света и обозначаются как северное: N, южное: S, западное: W и восточное: E). Восьмиугольные ячейки имеют, кроме ортогональных, и диагональные связи (NW, NE, SW, SE). Увеличенные возможности межсоединений на уровне прямых связей и увеличенные ресурсы программируемых связей внутри самих ячеек позволяют строить некоторые блоки (например, особо быстродействующие умножители), не прибегая к использованию шинных ресурсов трассировки.

В каждую из ячеек входят два табличных функциональных преобразователя (LUT), триггер типа D и полтора десятка программируемых мультиплексов, направляющих сигналы по тем или иным путям внутри самой ячейки и

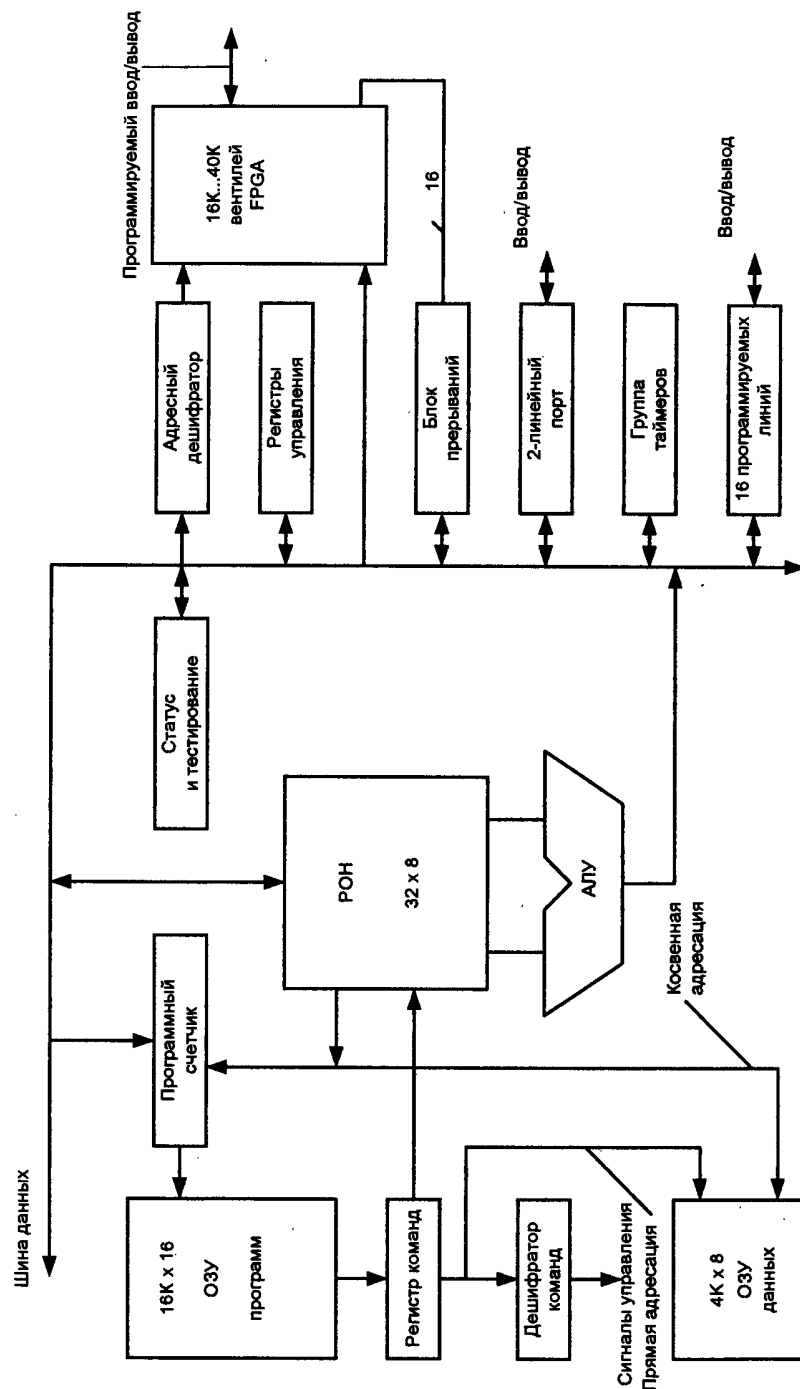


Рис. 1.24. Структурная схема микросхем семейства FPSLIC

на ее входы и выходы, а также создающих некоторые дополнительные функциональные возможности для самой ячейки. Функциональные преобразователи воспроизводят функции трех переменных (одних и тех же для обоих LUT) и могут быть объединены для выработки функций четырех переменных. Логические ячейки имеют несколько стандартных конфигураций: для воспроизведения функций четырех переменных (режим Synthesis), двух функций трех переменных (режим Arithmetic), двух функций от аргументов A, B, C и D (один из трех аргументов образован как конъюнкция двух входных переменных, режим имеет название DSP/MPL), режим с конфигурацией ячейки в виде схемы разряда счетчика (режим Counter) и режим реализации мультиплексора "2-1" с буфером (режим Tri-State/MUX).

Такой набор возможных режимов ячейки обеспечивает удобство разработки проектов с интенсивными вычислениями, таких как системы цифровой обработки сигналов (ЦОС), криптографические системы и другие мультимедийные проекты, для которых характерен большой объем вычислений, выполняемых в реальном масштабе времени. На основе блока FPGA реализуются и общеупотребительные интерфейсные функции (UARTs, PCI и др.).

В FPGA имеются 6 внешних и 2 внутренних линии тактирования, внутренние линии получают сигналы от ядра AVR. FPGA имеет собственную статическую память (FreeRAM) с временем доступа 10 нс и несколькими вариантами возможной организации: синхронной или асинхронной, одно- или двухпортовых RAM, буферов FIFO и др.

Система межсоединений в ядре FPGA иерархична и включает в себя локальные шины, экспресс-шины и прямые связи ячеек с ближайшими соседями. В связях имеются программируемые вентили, с помощью которых формируются шины с тремя состояниями.

Интерфейс между FPGA и AVR (рис. 1.25) предусматривает для FPGA 16 входных линий декодированного адреса от AVR и 16 выходных линий запросов прерываний с различными приоритетами для AVR.

Четырехразрядный адрес из адресного пространства процессора декодируется для образования 16 линий адресации объектов в FPGA. Данные передаются по 8-разрядной шине данных процессора. Таким образом, устройствам, реализованным в FPGA и имеющим адреса в адресном пространстве ввода/вывода AVR, придается возможность иметь собственный запрос прерывания. Кроме того, AVR посылает в FPGA сигналы стробов чтения и записи, управляющие двунаправленной шиной данных, образованной линиями экспресс-шин. Видно, что ядро AVR взаимодействует с устройствами, созданными в FPGA, подобно тому, как обычно процессор взаимодействует с внешними устройствами.

Для функций заказной периферии, которая может быть реализована в FPGA, существует макробиблиотека.

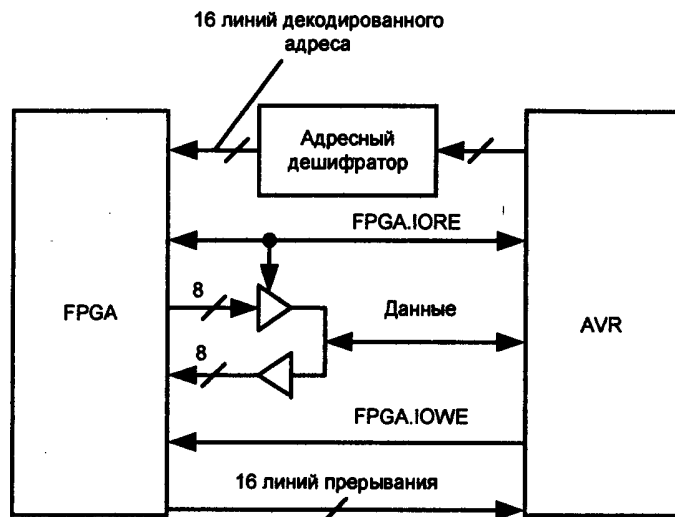


Рис. 1.25. Схема интерфейса между блоками FPGA и AVR микросхем семейства FPSLIC

Другой аспект взаимодействия FPGA и AVR состоит в их общем доступе к двухпортовой памяти SRAM с временем доступа 15 нс (рис. 1.26).

Между FPGA и процессором имеется 36 Кбайт двухпортовой памяти. Часть ее, имеющая емкость 20 Кбайт при организации 10К×16, всегда используется со стороны процессора как память программ, другая часть емкостью

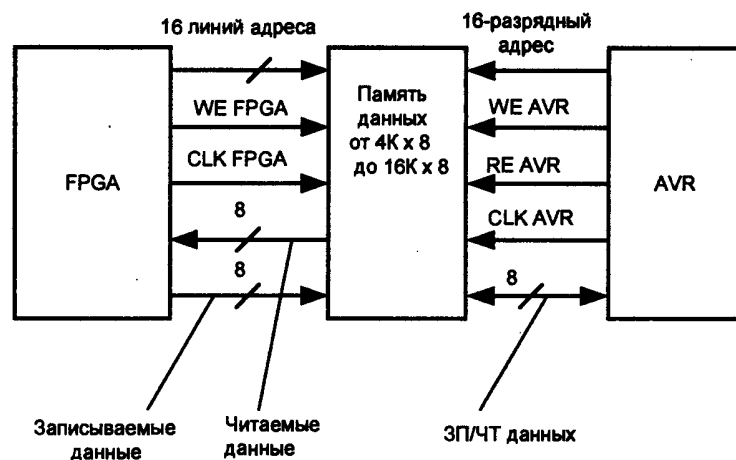


Рис. 1.26. Схема доступа блоков FPGA и AVR к общей двухпортовой памяти

4 Кбайта — всегда как память данных. Остальные 12 Кбайт могут полностью или частично присоединяться к памяти программ или памяти данных (с дискретностью 2К×16 или 4К×8, соответственно). Устройства, реализованные в FPGA, имеют непосредственный доступ к памяти данных, запрещаемый при загрузке конфигурации. Для записи и чтения в память используются экспресс-шины системы коммутации FPGA, расположенные на соответствующем краю матрицы. Со стороны FPGA вырабатываются только сигналы разрешения записи WE и отсутствуют сигналы разрешения чтения, которое всегда разрешено. Одновременно с операциями записи или чтения со стороны FPGA могут производиться и такие же операции со стороны процессора, поэтому при появлении одного и того же адреса от FPGA и AVR должен быть специально предусмотрен арбитраж. В большинстве случаев он заключается в ограничении доступа в память со стороны FPGA при возникновении конфликта.

Важная особенность микросхем FPSLIC — способность к реализации концепции кэш-логики (Cache Logic), впервые введенной фирмой Atmel. Кэш-логика — новое достижение в области разработки адаптивных систем, она позволяет полностью или частично конфигурировать систему "на лету" без потери имевшихся данных и нарушения работы ее неизменяемой части. Данные, которые были выработаны к моменту перестройки той или иной части системы, сохраняются. Та часть аппаратуры, в которой в данный момент идет обработка информации, представлена "нормальной" схемой соответствующей конфигурации, созданной в FPGA, а пассивная часть аппаратуры представлена данными, сохраняемыми в недорогих устройствах памяти. В итоге экономно выполняются требуемые преобразования. Когда активизируются новые операции, новая конфигурация записывается поверх старой.

Как известно, решаемые задачи обработки информации в конечном счете раскладываются на множество стандартных операций низшего иерархического уровня (сдвиги, сложения, умножение, мультиплексирование и т. п.). Ясно, что каждая из подобных функций может быть использована многократно в разных обстоятельствах, причем в одно и то же время активна только малая часть операторов, соответствующих этим функциям. Исключая избыточность и контролируя условия появления каждой операции, можно так организовать систему, что сложные функции будут воспроизводиться с помощью небольшого числа простых схем и, следовательно, на недорогих средствах программируемой логики. Так, например, в одном из известных примеров требуемая задача решалась обычным способом с использованием 10 тыс. вентилях, а на основе программируемых схем и концепции кэш-логики потребовались всего 2 тыс. вентилях и запоминание неизменной в данное время информации в дешевой системной памяти.

Концептуально различают *предопределенную* и *динамическую кэш-логику*. Первая подразумевает использование предопределенных функций и макро-

сов, хранимых во внешней энергонезависимой памяти, заранее отработанных и имеющих уже сгенерированные битовые потоки конфигурирования. Выполнением этих функций управляют средства самой кэш-логики или процессор.

Вторая разновидность кэш-логики должна определять требуемые функции с их размещением и трассировкой и генерировать битовые потоки настройки в реальном масштабе времени. Такой режим ассоциируется с работой адаптивных систем, и пока существует только как концепция, не имеющая законченной физической реализации.

Перспективы применения кэш-логики расцениваются как достаточно широкие. Ее применение снижает сложность программируемой части аппаратуры и, в конечном счете, повышает ее надежность (вследствие сокращения числа физически существующих схемных компонентов), снижает потребляемую мощность и стоимость. Предполагается применение кэш-логики, в первую очередь, в портативной аппаратуре, системах компьютерной графики и т. п.

Разработчики таких сложных СБИС, как FPSLIC, подчеркивают, что подобные микросхемы не следует рассматривать как просто микроконтроллеры с программируемой пользователем периферией, поскольку для таких относительно ограниченных целей могут найтись и более простые решения. Для микросхем типа FPSLIC и им подобных эффект достигается прежде всего при достаточно полном использовании их новых функциональных возможностей.

1.7.4. SOPC блочного типа фирмы Triscend

Важное место на рынке SOPC занимает корпорация Triscend, концентрирующая свои усилия исключительно на производстве микросхем типа "система на кристалле". Корпорация Triscend выпустила *первую в мире промышленную SOPC с аппаратным процессорным ядром* в то время, когда семейство FPSLIC только анонсировалось фирмой Atmel (апрель 1999 г.). Сейчас архитектуры SOPC фирмы Triscend принадлежат к числу наиболее популярных.

В микросхемах семейства E5 фирмы Triscend на одном кристалле объединены: 8-разрядное ядро микроконтроллера "турбо 8032", совместимое со ставшей в последнее время фактическим стандартом архитектурой 8051 фирмы Intel, FPGA, системная шина, память типа SRAM и несколько вспомогательных периферийных устройств. Для следующего семейства A7 фирмой Triscend взято 32-разрядное процессорное ядро ARM7TDM1. На рис. 1.27 показаны основные блоки микросхем семейства E5 и их взаимосвязи.

В состав микросхем этого семейства входят следующие основные блоки:

- Микроконтроллерное ядро с длительностью командного цикла 4 такта и тактовыми частотами до 40 МГц, что обеспечивает производительность

до 10 MIPS. В состав ядра входит ОЗУ (SRAM) с организацией 256×8, три таймера-счетчика, сторожевой таймер, блок прерываний на 12 запросов, два указателя данных и универсальный асинхронный приемопередатчик UART.

- FPGA с числом логических ячеек типа "LUT+триггер" от 256 до 3200 для различных микросхем семейства.

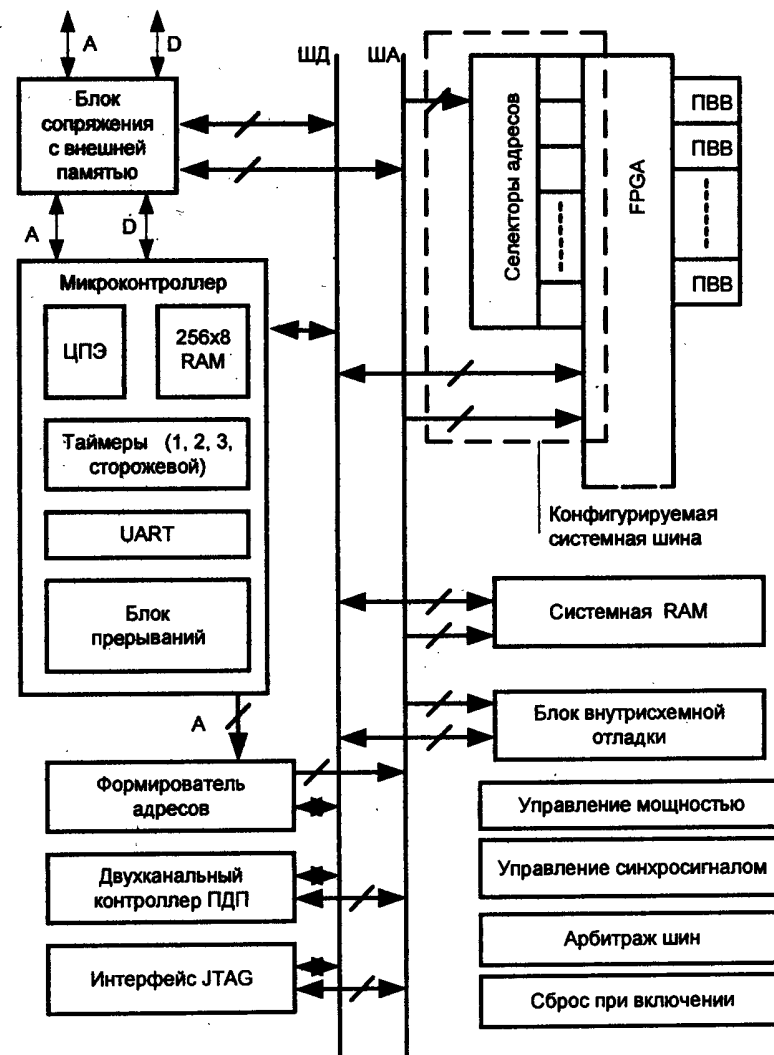


Рис. 1.27. Структура микросхем семейства E5 фирмы Triscend

- ❑ Высокопроизводительная конфигурируемая системная шина CSI (Configurable System Interconnect) с двумя 8-разрядными шинами данных (одной для чтения, другой для записи), 32-разрядной шиной адреса и адресными селекторами. Шина связывает FPGA, периферийные узлы и ядро микроконтроллера при скорости передач до 40 Мбайт/с, поддерживает возможность обращения к внешним устройствам, режим циклического арбитража и реализацию циклов ожидания. Адресные селекторы обеспечивают доступ к устройствам, реализованным в блоке FPGA, и при этом на создание адресных дешифраторов не расходуются ресурсы программируемой логики.
- ❑ Блок программируемых портов ввода/вывода ПБВ (PIO, Programmable Input-Output port) с программированием обычных или повышенных выходных токов. Схемы слежения за состояниями входов при отсутствии воздействия на них активного источника сигнала удерживают на входах последнее активное состояние. Возможно индивидуальное переключение линий ввода/вывода в энергосберегающий режим. Уровни сигналов совпадают со стандартом TTL.
- ❑ Блок сопряжения с внешней памятью для подключения к SOPC внешних запоминающих устройств с параллельным или последовательным интерфейсом и емкостью до 256К×8. Возможно дополнительное увеличение числа адресных линий до 32 и доступ через блок сопряжения с внешней памятью к внешним устройствам с внутренней системной шиной.
- ❑ Контроллер прямого доступа к памяти DMA (Direct Memory Access) с двумя независимыми каналами и скоростью передач до 40 Мбайт/с. Передачи сопровождаются подсчетом циклической контрольной суммы. Возможен режим передач "память-память" для организации обмена между различными областями памяти без участия процессорного ядра.
- ❑ Блок интерфейса JTAG для загрузки памяти конфигурации и исполняемой программы и для внутрисхемной отладки устройств, содержащих микросхемы семейства E5. Через этот блок возможен доступ к внутренней системной шине и адресуемым регистрам кристалла.
- ❑ Специальный блок внутренней отладки (In-System Debugging Hardware Breakpoint Unit), позволяющий задавать различные условия останова при выполнении микроконтроллером программы. Одновременно можно задавать две точки останова. В состоянии останова разрешен доступ к адресуемым регистрам кристалла через JTAG-интерфейс.
- ❑ Блок управления энергопотреблением для индивидуального задания энергосберегающих режимов устройствам системы и линиям портов ввода/вывода. В режиме снижения мощности для всех устройств ток, потребляемый кристаллом, не превышает 50 мкА.
- ❑ Система тактирования с внешним или внутренним задающим генератором. При применении внутреннего генератора его частота задается внешним кварцевым резонатором.

1.7.5. SOPC блочного типа фирмы Altera

К числу фирм, выпускающих блочные SOPC, в 2000 году подключилась и Altera. Производство SOPC этой фирмой началось с рассмотренного ранее в разд. 1.5 семейства APEX20K типа generic (с soft-ядрами). Позднее появились блочные SOPC с hard-ядрами, которые базировались на разработках фирм ARM Limited и MIPS Technologies. Аппаратные ядра процессоров этих фирм вместе с soft-ядром Nios образовали семейство встроенных процессоров Excalibur.

Процессорные ядра ARM и MIPS оптимизированы для эффективной интеграции с программируемой логикой семейства APEX20K/KE. Реализация в виде hard-ядер позволяет достичь производительности процессоров в сотни миллионов операций в секунду.

Общий план микросхем Excalibur (рис. 1.28) показывает основные блоки кристалла, в котором роль процессора играют ядра ARM или MIPS. Кроме них на кристалле находятся оперативная память RAM, кэш-память, интерфейсные схемы для связи с внешними устройствами и последовательный порт UART.



Рис. 1.28. Общий план кристалла микросхем семейства Excalibur

Микросхемы семейства Excalibur с аппаратными процессорными ядрами позволяют создавать системы с высокой производительностью и умеренной функциональной гибкостью.

Планируется развитие линии Excalibur, в частности, реализация 64-разрядных процессорных ядер и расширение набора устройств, реализуемых в области программируемой логики, а также и дополнительных средств для решения задач телекоммуникаций, цифровой обработки сигналов и под-

держки обновленных интерфейсных стандартов. Ведется и разработка ядер для семейства ACEX.

К числу фирм-производителей блочных SOPC относится и фирма Cypress Semiconductor. Особенностью SOPC этой фирмы является наличие подсистемы аналоговой программируемой периферии, в связи с чем их рассмотрение приводится в следующем разделе.

1.8. Микросхемы с программируемыми аналоговыми и аналого-цифровыми структурами

1.8.1. Общие сведения

Многие управляющие и вычислительные системы оперируют с информацией, представленной не только в цифровой, но и в аналоговой форме. Аналоговые сигналы вырабатываются многими датчиками физических величин, сведения о которых должны поступать далее в цифровую часть системы. До ввода этих величин в цифровые устройства и при их выводе из них часто необходимы некоторые операции (нормирование, фильтрация, аналого-цифровое преобразование, цифроаналоговое преобразование и др., что иногда обозначается термином End-Front Design). Таким образом, *постановка задачи интеграции всей системы на одном кристалле нередко приводит одновременно и к задаче реализации на этом кристалле программируемых средств обработки аналоговых сигналов.*

Аналоговые и аналого-цифровые фрагменты уже давно встраиваются в структуры микропроцессорных систем в виде отдельных микросхем малого и среднего уровней интеграции, использующих дискретные (навесные) операционные элементы. Технология БМК также нашла применение в области аналого-цифровой техники, но программирование самим пользователем аналоговых и аналого-цифровых схем, размещенных на кристаллах высокого уровня интеграции, до последнего времени практически не было освоено. Трудности освоения аналоговых интегральных схем в значительной мере объяснялись их пониженными точностными возможностями в сравнении со схемами на дискретных компонентах. Блоки АЦП и ЦАП с относительно давних пор встраивались в БИС (фирмы Analog Devices, Intel), но с очень нестабильными результатами. Появление "систем на кристалле" сделало особенно актуальной проблему интеграции надежных программируемых аналоговых и аналого-цифровых схем и цифровой части системы. На основе микросхем с программируемыми структурами возможно быстрое проектирование подсистем аналоговой и аналого-цифровой обработки сигналов, их отладка, создание промышленных образцов и быстрый выход на рынок.

Несколько крупных фирм-производителей микросхем с программируемыми структурами (Motorola, Lattice Semiconductor, Cypress) уже отреагировали на требования времени и уделили внимание разработкам программируемых аналоговых и аналого-цифровых структур, выполненных как на отдельном кристалле, так и совместно с цифровой частью системы.

Цифровые сигналы принимают лишь два значения, одно из которых соответствует логической единице, а другое — логическому нулю. Проблема точного задания этих сигналов отсутствует — требуется лишь надежно отличать один из этих сигналов от другого. Совершенно иным является положение в аналоговой технике, где *сигнал должен передавать точное значение величины с погрешностью в десятые или сотые доли процента*, т. е. требуется "дозирование" сигналов с разрешающей способностью в тысячи или даже более уровней. Традиционно (до конца 70-х — начала 80-х гг.) роль дозирующих параметров в схемах нормирующих усилителей, сумматоров и т. п. играли отношения сопротивлений точных резисторов. Постоянные времени RC (также масштабирующие параметры в схемах интеграторов, фильтров и др.) задавались совместно значениями сопротивления точного резистора и емкости операционного конденсатора. Так, например, в известной схеме масштабирующего усилителя, т. е. устройства умножения сигнала, заданного напряжением постоянного тока, на константу используются два точных резистора $R1$ и $R2$, от соотношения сопротивлений которых зависит функциональная характеристика схемы, в идеализированном виде имеющая вид $U2 = (-R2/R1)U1$, где $U1$ и $U2$ — входное и выходное напряжения, соответственно. Интегратор имеет идеализированную функциональную характеристику вида:

$$U2 = (-1/RC) \text{INT } U1(t) dt,$$

в которой роль масштабирующего коэффициента играет произведение сопротивлений точного резистора входной цепи на емкость конденсатора цепи обратной связи.

В схемотехнике с дискретными (навесными) схемными элементами проблема реализации точных резисторов имеет удовлетворительное решение. Для технологии интегральных схем эта проблема намного сложнее, но существует *альтернативное схемное решение*, благодаря которому резисторы имитируются цепями, содержащими коммутируемые (переключаемые) конденсаторы (рис. 1.29).

В такие цепи входят конденсатор C и ключевые транзисторы $T1$ и $T2$, управляемые тактирующими напряжениями $U1$ и $U2$. Транзисторы $T1$ и $T2$ под воздействием тактирующих напряжений замыкаются поочередно, и конденсатор C попеременно заряжается через замкнутый ключевой транзистор до напряжения $U1$ или $U2$. В момент замыкания ключевого транзистора заряд конденсатора изменяется на величину $q = q1 = q2 = C(U1 - U2)$. Изменение заряда осуществляется короткими импульсами тока, протекаю-

щими через конденсатор при замыкании соответствующего ключевого транзистора.

Среднее значение тока в цепи между точками 1 и 2 составляет величину

$$i = q/T = (U_1 - U_2)C/T,$$

где T — период тактирующих импульсов.

Из полученного выражения видно, что в определенном смысле (для средних значений сигналов) цепь ведет себя как резистор с сопротивлением $R = T/C$, сопротивление которого пропорционально периоду тактирующих импульсов, т. е. обратно пропорционально их частоте. Разница состоит в том, что резистор регулирует поток передаваемого через него заряда плавно, а цепь с переключаемыми конденсаторами — импульсно.

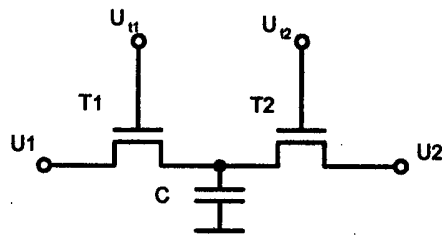


Рис. 1.29. Цепь с переключаемыми конденсаторами, моделирующая резистор

На основе схем с переключаемыми конденсаторами можно строить разнообразные операционные звенья, аналогичные известным из традиционной аналоговой схемотехники, путем замены резисторов эквивалентными им цепями. Сопротивления эквивалентных цепочек управляются значениями тактовой частоты $f = 1/T$, которые можно изменять и дистанционно. А это ставит аналоговые блоки в один ряд с цифровыми с точки зрения возможностей программирования в системе на расстоянии (например, через Интернет). В схемотехнике с переключаемыми конденсаторами строятся схемы с зависимостью функциональных характеристик только от отношения емкостей, которое может задаваться с высокой точностью. Именно схемотехника на основе переключаемых конденсаторов дает возможность реализовать АЦП и ЦАП с высоким уровнем параметров. Параметры емкостей мало критичны к изменению температуры и старению. Резко (в сотни раз) снижается площадь, занимаемая цепями с переключаемыми конденсаторами в сравнении с цепями, содержащими точные резисторы. Таковы технологические достоинства схемотехники переключаемых конденсаторов. В то же время применение цепочек с переключаемыми конденсаторами имеет и свои недостатки. В цепях с непрерывными сигналами (без переключаемых конденсаторов) отсутствует проблема отделения полезной информации (среднего значения пульсирующей величины) от сопровождающих ее паразитных высокочастотных составляющих, что благоприятно влияет на дина-

мические характеристики устройств. Кроме того, схемы с непрерывными сигналами имеют лучшие шумовые характеристики.

1.8.2. Практические разработки

Впервые о создании БИС с массивом программируемых пользователем аналоговых элементов объявила фирма Motorola (1997 г.). Эти БИС были анонсированы под названием MPAA020 (Motorola Field Programmable Analog Arrays), но не были доведены до промышленного выпуска. В 1999 году фирма Lattice Semiconductor выпустила семейство внутрисхемно программируемых (In-System Programmable) аналоговых схем типов ispPAC10 и ispPAC20, а затем и ispPAC80. Фирма Cypress Semiconductor выпустила микросхемы класса SOPC с реализацией цифровой и аналоговой части программируемой логики в пределах одного кристалла.

В микросхемах фирмы Motorola для построения операционных звеньев используется схемотехника с переключаемыми конденсаторами, в микросхемах фирмы Lattice Semiconductor — традиционные решения с применением масштабирующих резисторов и, следовательно, непрерывных сигналов, в SOPC фирмы Cypress Semiconductor — оба вида схемотехнических решений, т. е. и схемы с точными резисторами, и схемы с переключаемыми конденсаторами.

Микросхемы семейства ispPAC фирмы Lattice Semiconductor архитектурно просты (имеют немного конфигурируемых ресурсов и контактов ввода/вывода). Память конфигурации реализована по технологии EEPROM и может загружаться через специально выделенные контакты JTAG-интерфейса. Допустимое число циклов репрограммирования не менее 10 тыс. Конфигурация может быть закрыта от несанкционированного доступа битом секретности. В состав конфигурируемых ресурсов включены не только аналоговые, но и цифроаналоговые средства (схема ispPAC20 содержит встроенный восьмиразрядный цифроаналоговый преобразователь).

Макрочаежки ИС ispPAC10 и ispPAC20 (рис. 1.30 и 1.31), называемые PACblocks, состоят из двух входных (инструментальных) усилителей IA и выходного усилителя OA. В микросхеме ispPAC10 таких макрочаеек четыре, а в микросхеме ispPAC20 — две.

Назначение других блоков микросхем указано на самих рисунках. Выходы IA подключены к потенциально заземленным (суммирующим) точкам выходных усилителей. Входные и выходные цепи усилителей выполнены по дифференциальной схеме, защищенной от воздействий других сигналов, что делает качество передачи сигнала независимым от положения линии связи на кристалле и улучшает другие характеристики макрочаеек. Входные сопротивления усилителей, реализованных в схемотехнике КМОП, настолько велики, что практически снимают вопрос нагрузки на источники сигналов, в том числе на датчики исходных физических величин. Взаимосоединения блоков программируются. Из нескольких макрочаеек можно строить схемы

перестраиваемых активных фильтров. Восьмиразрядный цифроаналоговый преобразователь микросхемы ispPAC20 может воспринимать параллельные или последовательные данные. Для компараторов возможно программирование работы с гистерезисом или без него. В схемах выходных усилителей программируется подключение или отключение резисторов обратной связи. Емкости конденсаторов обратной связи могут программироваться на величины от 1 до 63 пФ (всего 128 возможных значений).

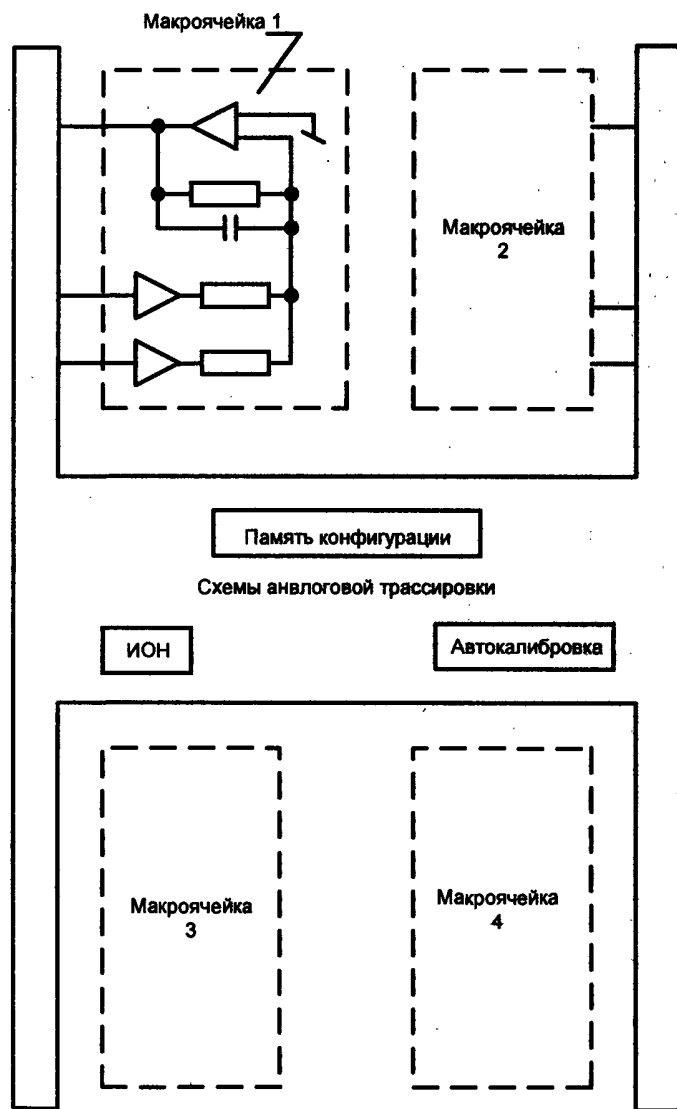


Рис. 1.30. Структура PACblock микросхем ispPAC10

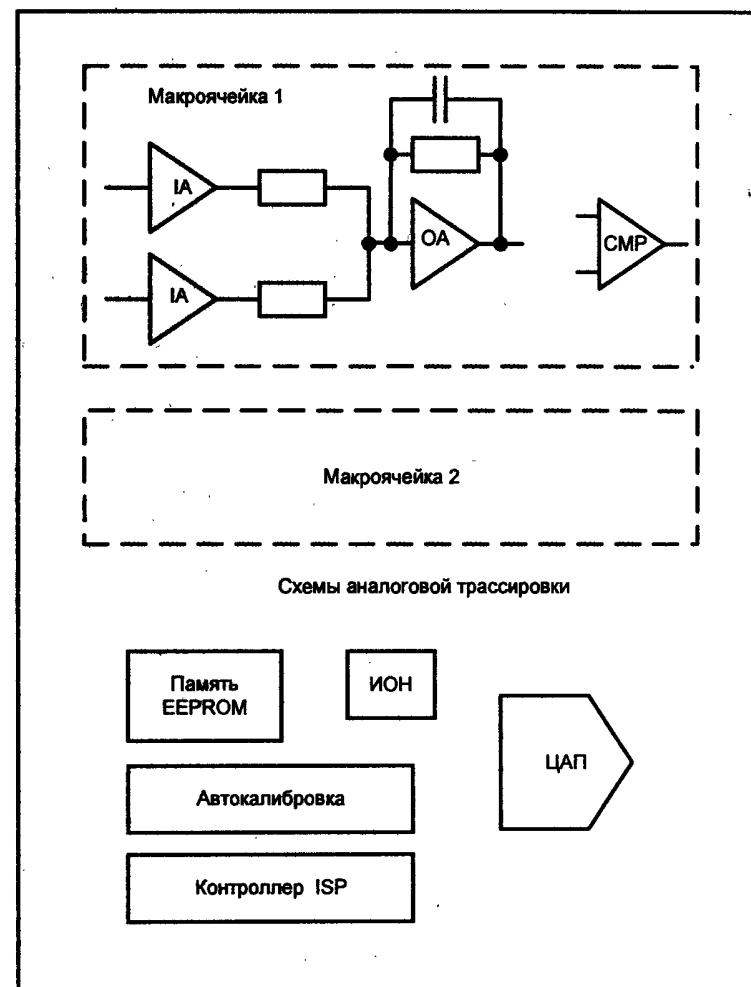


Рис. 1.31. Структура PACblock микросхем ispPAC20

Микросхемы ispPAC80 предназначены для создания активных фильтров пятого порядка с программируемыми характеристиками. Могут строиться фильтры с аппроксимацией частотных характеристик по Баттерворту, Бесселю, Чебышеву, фильтры с аппроксимацией эллиптического типа.

Аналоговые блоки следует характеризовать совокупностью их функциональных и точностных характеристик с учетом статических и динамических погрешностей. Укажем основные параметры семейства ispPAC. Микросхемы работают от источника питания 5 В и имеют источник образцового напряжения 2,5 В, позволяющий выводить рабочую точку усилителей в середину питающего напряжения для восприятия и выработки знакопеременных сиг-

налов. Уровень шумов равен приблизительно $40 \text{ нВ/Гц}^{0,5}$, что дает динамический диапазон около 100 дБ в диапазоне частот до 100 кГц. Типовое значение коэффициента гармоник на частоте 10 кГц составляет -88 дБ, а на частоте 100 кГц — -67 дБ. Автоматическая калибровка смещения нуля (длительностью 100 мс) обеспечивает максимальное смещение менее чем на 1 мВ при типовом значении 200 мкВ и любых конфигурациях схемы и коэффициентах усиления. Коэффициенты усиления IA программируются в пределах от -10 до +10 с шагом 1. Типовая точность задания коэффициентов усиления IA характеризуется погрешностью 1,5%, максимальная погрешность 4%. Температурный дрейф 20 ppm/°C. Диапазон частот для создаваемых фильтров 10—100 кГц, значения частот задаются с погрешностями не более 3,5—5%. Эксплуатационный диапазон температур от -40 до +85 °C. Время установления выходного напряжения цифроаналогового преобразователя с погрешностью 0,1% не превышает 6 мкс. Дифференциальная нелинейность преобразователя не более единицы младшего разряда. Типовое время переключения компаратора при амплитуде перепада 10 мВ составляет 750 нс. Максимальные токи потребления микросхем приблизительно 20 мА.

Относительно скромные точностные возможности микросхем ispPAC, тем не менее, приемлемы для построения на них ряда работоспособных функциональных узлов. Программное обеспечение фирмы Lattice Semiconductor под названием PAC-Designer ориентировано на использование в PC-совместимых компьютерах с ОС Windows. САПР позволяет вводить информацию о проекте, моделировать функционирование схем, компилировать проекты и загружать результаты компиляции в память конфигурации БИС.

Фирма Cypress Semiconductor выпустила семейство CY8C25/26, сочетающее цифровые и аналоговые программируемые блоки.

SOPC CY8C25/26xxx имеют восьмиразрядное процессорное ядро M8C с умножителем/аккумулятором. Возможен выбор частот тактирования. Емкости памяти программ и данных в зависимости от типа микросхемы варьируются (от 4 до 16 Кбайт Flash-памяти для программ и от 128 до 256 байт SRAM-памяти для данных).

В микросхемах CY8C25/26 вокруг быстродействующего процессорного ядра размещены и цифровые, и аналоговые программируемые блоки (SOC-блоки), функционирование которых задается при конфигурировании системы (рис 1.32).

Выделение в отдельную часть цифровых блоков является вариантом организации интерфейса между процессором и FPGA. SOC-блоки могут соединяться друг с другом двумя основными способами: параллельным и последовательным. При параллельных соединениях достигается повышение точности блоков (увеличение разрядности аналого-цифровых преобразователей, повышение разрешающей способности таймеров и т. п.). При последовательных соединениях создаются тракты из нескольких операционных звень-

ев, например, последовательное соединение предварительного усилителя, фильтра и аналого-цифрового преобразователя.

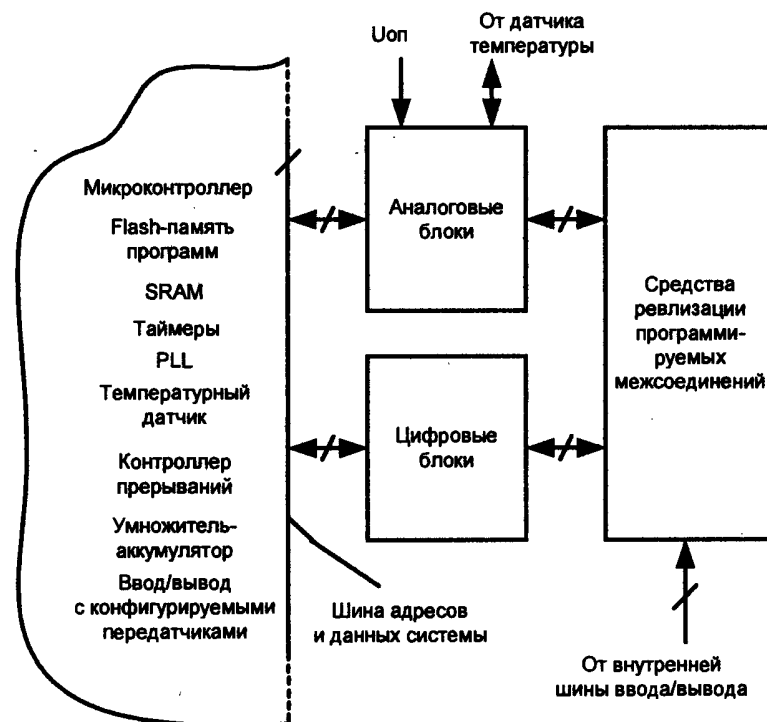


Рис. 1.32. Структура с цифровыми и аналоговыми SOC-блоками микросхем семейства CY8C25/26

В состав SOC-блоков первого из выпущенных представителей семейства CY8C25/26 входят 12 аналоговых и 8 цифровых блоков. Из 12 аналоговых блоков 4 реализованы в схемотехнике с масштабирующими резисторами (блоки с непрерывными сигналами (CT, Continuous Time building blocks)) и 8 построены на переключаемых конденсаторах (SC, Switched-Capacitor blocks). Блоки с непрерывными сигналами оптимизированы для построения усилителей с программируемыми коэффициентами усиления и мультиплексируемыми каналами входов и выходов, дифференциальных усилителей и быстродействующих компараторов. Блоки, использующие технику переключаемых конденсаторов, ориентированы на создание программируемых фильтров, регистров последовательного приближения, аналого-цифровых и цифроаналоговых преобразователей и т. п. Цифровые блоки реализуют такие функции, как таймеры, универсальные асинхронные приемопередатчики, широтно-импульсные модуляторы и др.

С помощью аналоговых и цифровых средств коммутации, содержащих мультиплексоры и ключи, SOC-блоки соединяются в так называемые *модули пользователя* (User modules), для формирования которых подготовлена специальная библиотека. Библиотека содержит несколько типов аналого-цифровых преобразователей (последовательного приближения, следящего типа, с дельта-сигма модуляцией), цифроаналоговые преобразователи, компараторы, устройства выборки и хранения аналоговых данных, разнообразные виды фильтров (низкой частоты, высокой частоты, полосовых), генераторы и детекторы напряжений специальной формы, модуляторы и демодуляторы (всего около 40 аналоговых модулей пользователя).

Цифровая часть библиотеки (около 30 модулей пользователя) включает в себя таймеры, широтно-импульсные модуляторы, приемопередатчики и законченные структуры UART, интерфейс SPI, CRC-генераторы, генераторы псевдослучайных последовательностей и схемы управления двигателями.

Программируемые аналоговые микросхемы FPAD (Field Programmable Analog Devices) появились также в серии TRAC фирмы Fast Analog Solutions, входящей в группу ZETEX.

Появилось сообщение о выпуске фирмой Anadigm Inc. микросхем FPAА с программированием операционных усилителей и схем их межсоединений. В микросхему входят 20 таких блоков, выполняющих обычные функции обработки аналоговых сигналов в диапазоне частот от нулевой до 500 кГц. На процесс реконфигурации схемы затрачивается 100 мкс.

1.9. О некоторых проблемах разработки и использования БИС/СБИС с программируемой структурой

1.9.1. Конвертация проектов

Промышленные изделия в своем жизненном цикле проходят *несколько стадий*. Крупными стадиями являются *разработка изделия, стадия активной реализации, стадия спада производства* в связи с устареванием изделия и заменой его более новым и совершенным. Для создания изделия электронной промышленности, как известно, *можно пользоваться разными средствами* (стандартными ИС, программируемыми ИС, полузаказными и заказными ИС). Учет особенностей разных стадий жизненного цикла изделия в сочетании с возможностями тех или иных вариантов его реализации необходим для эффективной организации процесса производства.

На стадии разработки проекта трудно переоценить роль применения микросхем, репрограммируемых пользователем, как чрезвычайно полезных средств *отработки прототипа* изделия, позволяющих легко и быстро вносить в про-

ект изменения, доводя его до полностью работоспособного устройства. Останется ли вариант с репрограммируемыми микросхемами конечной продукцией зависит от объема его производства. Если изделие имеет высокий спрос, то, возможно, целесообразно перевести его на реализацию в виде полузаказных или даже заказных схем, т. к. это позволит *устранить из устройства средства программирования его структуры, что снизит стоимость изготовления каждого экземпляра микросхемы и, при необходимости, позволит повысить ее быстродействие*. Затраты на проектирование некоторого числа фотошаблонов, необходимых для производства полузаказных или заказных схем, и другие однократные технологические затраты (NRE, Non Recurrent Expendts) не зачеркнут указанный выигрыш, если тиражность изделия достаточно велика. На последней стадии жизненного цикла изделия объем его производства снижается и становится неустойчивым. При этом может оказаться выгодным возврат от варианта с полузаказными и заказными микросхемами к БИС/СБИС с программируемыми структурами (FPGA, CPLD, FLEX, SOPC), т. к. для них не существует проблемы заказа в виде только больших партий, и можно избавиться от риска экономических потерь при уценках и распродажах. Кроме того, несложно выпустить и дополнительную партию изделий любого объема, если конъюнктура этого потребует.

Как видно из сказанного, между разными средствами реализации одних и тех же изделий должна существовать взаимосвязь, поскольку иначе процесс перехода от одних средств к другим (*конвертация проектов*) окажется затрудненным. Такая взаимосвязь оказывает влияние на архитектуры, электрические характеристики и конструктивные параметры как БИС/СБИС программируемой логики, так и БМК. Например, были выпущены специализированные репрограммируемые FPGA с триггерной памятью конфигурации, предназначенные для макетирования схем, реализуемых на определенных масочных БМК (фирма Quickturn System Inc.). Были созданы также специализированные масочные БМК (Hard Wire), по архитектуре идентичные схемам известных репрограммируемых FPGA. И в дальнейшем FPGA подстраивались под конкретные БМК с целью их эффективного макетирования, а определенные БМК приспособлялись по логической структуре, библиотеке схемных решений и электрическим параметрам к конкретным FPGA.

Проиллюстрируем высказанные тезисы некоторыми примерами. Практика фирмы Xilinx дает следующие цифры: почти все проекты с логической емкостью меньше 20 тыс. вентилях остаются реализованными на микросхемах программируемой логики и в качестве конечной продукции. Для проектов с логической емкостью в диапазоне от 20 до 50 тыс. вентилях могут приниматься различные решения на основе конкретной ситуации. Для проектов с логической емкостью больше 50 тыс. вентилях конвертация, как правило, целесообразна. Указанные цифры довольно быстро изменяются в сторону увеличения и, соответственно этому, конвертации в направлении "ПЛИС-

ASIC" подвергается все меньший процент проектов. Большинство конвертируемых проектов переводится в схемы на БМК. В схемы, реализованные по методу стандартных ячеек и, тем более, в полностью заказные схемы конвертация производится реже, т. к. это обычно экономично только для очень сложных или высокотиражных проектов.

Для конвертации своих проектов фирма Xilinx несколько лет назад создала микросхемы HardWire FpgASIC, обеспечивающие полную совместимость контактов ввода/вывода при замещении ими конвертируемых FPGA. Каждая микросхема HardWire FpgASIC соответствует функционированию и особенностям конкретной FPGA. Применение этих микросхем в два раза ускорило процесс конвертации в сравнении с традиционным процессом проектирования на основе БМК. Конвертация основана на заранее созданной и проверенной базе данных, тогда как обычный перевод схемы из реализации на FPGA в реализацию на БМК нуждается в повторной верификации, что сопряжено с большими затратами времени.

Если исходный проект выполнен на нескольких FPGA, то можно удешевить его переводом лишь некоторых FPGA в базис HardWireFpgASIC. Замена лишь части FPGA на HardWireFpgASIC и оставление других FPGA в обычном виде позволяет использовать их в традиционном варианте с возможностью реконфигурации. Перевод схемы в реализацию на HardWireFpgASIC позволяет также исключить соответствующую память загрузки конфигурации, что дополнительно упрощает и удешевляет схему.

В противовес сказанному следует отметить, что реальная ситуация с продукцией HardWire не совпала с прогнозом, и это направление не стало "серьезным бизнесом", имеющим достаточное число заказчиков. Ограниченное применение конвертации имеет несколько причин. Во-первых, часто не оправдываются ожидания большой тиражности выпуска изделия, во-вторых, нередко через полгода-год у производителя возникает желание внести в проект изменения, для чего хороши именно ИСПС, в третьих, с усложнением схем и появлением в них разнородных областей все сложнее воспроизвести ИСПС в БМК или других ASIC с сохранением временных соотношений сигналов и работоспособности и быстродействия фрагментов ИСПС и схем в целом. Так или иначе, но фирма Xilinx прекратила усилия по внедрению микросхем HardWire и сосредоточилась на удешевлении своих FPGA.

Почти одновременно с прекращением предложений средств конвертации HardWire фирмой Xilinx фирма Altera выступила с новым решением задачи перевода проектов, реализованных на ПЛИС, в проекты типа ASIC, объявив о выпуске микросхем HardCору. Это решение дает удобства для тех, кто использовал ПЛИС фирмы на стадии проектирования и затем намеревается переключиться на реализацию по методу стандартных ячеек для тиражной продукции. Сообщается, что перевод проекта на микросхемы HardCору дает 70% экономии площади кристалла при снижении стоимости микросхем и

и совместимости исходного и конечного вариантов. Переход от проекта, реализованного на ПЛИС, к эквивалентному проекту на ASIC производится приблизительно за 8 недель. С учетом времени, которое будет затрачено заказчиком на верификацию проекта, время до выпуска продукции оценивается приблизительно в 16 недель. Технология HardCору, по мнению фирмы Altera, дает хорошие предпосылки для повышения эффективности процесса конвертации, причем она нацелена, в основном, на конвертацию проектов высокой сложности.

Фирма AMI (American Microsystems Inc.) извещает об эффективной конвертации в ASIC-реализацию таких сложных ПЛИС, как семейство Virtex фирмы Xilinx и семейство APEX фирмы Altera. Имея для такой конвертации отработанный набор схемных решений, фирма AMI обеспечивает быструю конвертацию проектов, содержащих разнообразные блоки, в том числе DLL, PLL, PCI, блоки памяти с разнообразными организациями, различные варианты схем ввода/вывода. Микросхемы семейства XL-3 фирмы AMI дешевле своих аналогов из семейств Virtex и APEX приблизительно в 4 раза.

9.2. Конфигурирование БИС/СБИС программируемой структурой

Способ конфигурирования БИС/СБИС ПЛ, т. е. настройки их на определенное функционирование, зависит от типа программируемых элементов. Для микросхем с необратимым изменением состояний программируемых элементов (типов fuse и antifuse) и реконфигурируемых микросхем с энергонезависимой памятью конфигурации (EPROM, EEPROM, Flash) для программирования используются специальные воздействия электрическими сигналами, характеристики которых существенно отличаются от характеристик рабочих (логических) сигналов. Конфигурирование таких микросхем может производиться вне создаваемой системы с помощью программаторов или же в составе системы (т. е. при сохранении монтажа микросхемы на плате), но также с использованием специальных режимов. При этом для процессов записи информации требуются повышенные напряжения программирования (для микросхем семейств MAX7000 и MAX9000 фирмы Altera, например, это 12 В при рабочем напряжении питания 5 В). Для старых микросхем вследствие этого требовались несколько источников питания, более новые стали снабжаться внутренними преобразователями рабочего напряжения в повышенное напряжение программирования.

Для БИС/СБИС ПЛ со статической памятью конфигурации ее загрузка не требует каких-либо специальных электрических режимов, и процесс конфигурирования состоит в передаче в микросхему информации по заданному протоколу и с фиксированными форматами данных. Переданная информация обеспечивает создание требуемых соединений в логических блоках, блоках ввода/вывода и подключение их к трассам межсоединений. Опера-

ция конфигурирования выполняется после каждого включения питания, причем, если установлены специальные загрузочные БИС, сам факт очередного включения питания автоматически инициирует процесс конфигурирования, который может повторяться неограниченное число раз. Отсутствие специальных электрических режимов для записи информации в память конфигурации обеспечивает возможность ее проведения в работающей схеме, причем возможна и частичная реконфигурация, относящаяся лишь к части системы.

Конфигурирование БИС/СБИС ПЛ со статической (триггерной) памятью конфигурации представляет собой запись во внутренние регистры (триггеры) данных, задающих структуру блоков системы и их межсоединений. Каждый бит настроечных данных задает состояние соответствующему триггеру, управляющему программируемым ключом в настраиваемой схеме.

БИС/СБИС ПЛ обычно имеет несколько возможных режимов конфигурирования (например, у FPGA XC4000 фирмы Xilinx их 6, у СБИС семейства Virtex той же фирмы 4, у микросхем семейства Spartan — 2). У способов конфигурирования микросхем разных фирм и разных типов много общего. Рассмотрим, например, как конфигурируется СБИС ПЛ типа SOPC семейства Virtex.

Для конфигурирования этой микросхемы используются как специализированные выводы, так и выводы, которые после завершения конфигурирования могут играть роль выводов общего назначения. К специализированным относятся выводы M2, M1, M0, на которых задается код того или иного режима, вывод для синхросигналов процесса конфигурирования, выводы PROGRAM, DONE и выводы периферийного сканирования TDI, TDO, TMS, TCK (см. разд. 2.6). Вывод синхросигнала может быть выходом, когда этот сигнал генерируется микросхемой, или входом, когда поступает извне.

Возможные способы конфигурирования:

- ☐ пассивный последовательный (Slave-serial mode);
- ☐ активный последовательный (Master-serial mode);
- ☐ байт-последовательный (SelectMAP mode);
- ☐ периферийного сканирования.

В *пассивном последовательном режиме* микросхема получает данные конфигурирования в виде потока битов из последовательной памяти PROM или другого источника. Синхронизация осуществляется от внешнего источника, каждый положительный фронт синхросигнала вводит бит данных от входа DIN. Несколько микросхем могут быть соединены в цепочку для конфигурирования в едином процессе от общего потока битов. В этом случае после завершения конфигурирования очередной микросхемы данные конфигурации для следующих микросхем появляются на выводе DOUT микросхемы, завершившей конфигурирование.

В *активном последовательном режиме* выходной синхросигнал микросхемы подается на последовательное ЗУ, с которого на вход DIN микросхемы поступает последовательный поток битов конфигурации. Микросхема воспринимает каждый бит под управлением положительного фронта синхросигнала. После загрузки очередной микросхемы, входящей в цепочку, данные для следующей снимаются с выхода DOUT той микросхемы, которая закончила конфигурирование. Для синхронизации процесса можно выбирать частоту из широкого диапазона значений. По умолчанию используется наименьшая частота 2,5 МГц. Максимальная частота — 60 МГц. Устанавливаемые частоты, естественно, должны соответствовать возможностям используемых PROM и включенных в цепочку микросхем. При включении питания устанавливается частота 2,5 МГц. Если не поступит команда на изменение этого значения (опция ConfigRate в программе генерации битового потока конфигурации), процесс продолжится до конца на частоте 2,5 МГц.

В *байт-последовательном режиме* время конфигурирования минимально. Используется байт-последовательный поток данных, которые записываются в микросхему с учетом флажка ее готовности BUSY. Байтовый поток задается от внешнего источника, как и сигналы тактирования, разрешения работы CS и WRITE. В этом режиме данные могут и читаться. Если сигнал WRITE пассивен, то данные конфигурации читаются из микросхемы (этот процесс есть часть процесса Readback). В режиме SelectMAP также можно конфигурировать одновременно несколько микросхем, но в этом случае они включаются параллельно по входам синхронизации, данных, WRITE и BUSY. Загружаются микросхемы поочередно путем соответствующего управления сигналами разрешения их работы CS.

В *режиме периферийного сканирования* конфигурирование осуществляется исключительно через выводы порта тестирования TAP (Test Access Port) интерфейса JTAG. Используется специальная команда CFG_IN, позволяющая входным данным от вывода TDI преобразовываться в пакеты данных для внутренней шины конфигурации микросхемы.

Процесс конфигурирования для микросхем Virtex состоит из трех этапов: очистки памяти конфигурации, загрузки в нее данных и активизации логических схем, участвующих в процессе.

Конфигурирование начинается автоматически после включения питания, но может быть и задержано пользователем с помощью сигнала PROGRAM, снятие которого запрещает конфигурирование. Завершение очистки памяти выявляется с помощью сигнала INIT, а завершение всего процесса — с помощью сигнала DONE.

Данные для загрузки памяти конфигурации формируются системой автоматизированного проектирования.

Реконфигурация в системе (ISP, In-System Programmability) — одно из важнейших достоинств СБИС ПЛ, позволяющее легко изменять логику их ра-

боты. Потребности в изменениях возникают для устранения не выявленных при первоначальном тестировании ошибок, для модернизации (Upgrade) систем и в системах с многофункциональным использованием блоков. Наличие ISP облегчает работу с современными СБИС ПЛ, корпуса которых имеют большое число миниатюрных и легко повреждаемых выводов, что делает однократность установки микросхем на плату весьма желательной. Кроме того, реконфигурация микросхемы на расстоянии с использованием средств телекоммуникации или сети Интернет дает и экономическую выгоду, поскольку обходится дешевле, чем вызов техника для проведения этой операции у заказчика.

Возможности программирования в системе растут, если при проектировании часть функциональных возможностей СБИС ПЛ оставлять свободной, имея также запас по скорости и ресурсам межсоединений. При реконфигурации в системе должно сохраняться назначение внешних выводов, иначе потребуются изменить монтаж печатных плат.

Среди СБИС ПЛ имеются и такие, в которых *реализованы одновременно триггерная и энергонезависимая память конфигурации*. В этом случае конфигурирование СБИС ПЛ можно производить без внешних источников данных путем автоматической загрузки триггерной памяти из энергонезависимой. Конфигурирование СБИС ПЛ выполняется с помощью интерфейса JTAG.

1.9.3. Засекречивание проектов

Проблема защиты интеллектуальной собственности для многих вариантов микросхем программируемой логики приобретает *особую остроту*. Имея дело с программируемыми схемами, во многих случаях легко воспользоваться плодами чужого труда, т. к. при этом не потребуется что-либо разрабатывать и изготавливать, а нужно лишь получить сведения о содержимом памяти конфигурации и затем загрузить их в готовую микросхему, купленную у поставщика. Возможности легкой кражи проектов создают большой соблазн для недобросовестных людей, которых следовало бы называть просто ворами, но мы, во избежание бытового оттенка изложения, назовем их взломщиками.

Взломщики могут преследовать цели простого неосознанного дублирования чужих проектов для получения работающих схем без знания их внутреннего устройства. Такое дублирование называют *клонированием* проектов (Cloning). Более сложна задача расшифровки чужих проектов с раскрытием их архитектуры и деталей реализации. Такую операцию можно назвать *реконструкцией* проектов (Reverse-engineering), для нее подходит и встречающийся иногда термин *"реинжиниринг"*. Имея реконструированный проект, можно внести в него какие-либо несущественные изменения и попытаться обойти вопросы лицензирования.

Уязвимость микросхем программируемой логики по отношению к клонированию или реконструкции проектов зависит от характера проекта и схемотехнологии микросхемы.

В начале развития программируемых микросхем, когда они были представлены простыми комбинационными ПЛМ и ПМЛ, проекты можно было реконструировать, подавая на схему все возможные комбинации входных сигналов и фиксируя соответствующие им выходные комбинации сигналов. Из полученных сведений можно вывести булевы функции, воспроизводимые схемой, и, следовательно, реконструировать ее в соответствующем логическом базисе.

Более сложные схемы программируемой логики, практически, не поддаются такому методу реконструкции. Они имеют большое число вводов/выводов, назначение которых заранее не известно (например, у двунаправленных выводов). Уже одно это создает очень большие сложности для логического анализа проектов, т. к. неизвестно, подавать ли на вывод входной сигнал, снимать ли с него выходной сигнал, или же использовать его поочередно в обоих вариантах. Сложность внутренней структуры микросхем, наличие в них фрагментов, являющихся последовательностными схемами, и различных встроенных функций чрезвычайно затрудняют логический анализ проектов.

Для клонирования проектов нужно раздобыть сведения о содержимом памяти конфигурации микросхемы (битовом потоке конфигурирования). По возможностям защиты этой информации от несанкционированного доступа *микросхемы разных схмотехнологий существенно различаются*.

Однократно программируемые БИС/СБИС с пробиваемыми при конфигурировании перемычками наиболее защищены от взлома. Для их эксплуатации битовый поток конфигурирования не нужен, поскольку программирование перемычек завершается на стадии изготовления микросхемы. В распоряжении взломщика находится лишь сам кристалл. Для раскрытия проекта требуется определить состояние всех перемычек, получив для каждой ответ на вопрос "замкнута-разомкнута". Это чрезвычайно трудно сделать по следующим причинам. Число перемычек очень велико (сотни тысяч). Наблюдением поверхности кристалла нельзя выявить не только состояние перемычек, но и их местоположение, т. к. по виду они не отличаются от простого пересечения шин. Чтобы рассмотреть перемычку и определить ее состояние (наличие проводящей нити), нужно сделать в ней несколько поперечных срезов, а это, вероятнее всего, разрушит остальную часть кристалла. Таким образом, для выявления состояний всех перемычек потребуется испортить столько кристаллов, сколько перемычек имеется на каждом из них. Задаваясь вопросом, есть ли проекты, ценность раскрытия которых оправдывает затраты на их взлом, можно ответить, что *возможность раскрытия проекта в данных ситуациях есть категория скорее теоретическая, чем практическая*.

Описанная ситуация касается микросхем фирмы Actel с перемычками PLICE (ONO), фирмы QuickLogic с перемычками типа ViaLink, фирмы Xilinx с перемычками типа MicroVia.

Репрограммируемые БИС/СБИС с энергонезависимой памятью конфигурации (EPROM, EEPROM, Flash) в рабочих режимах также не используют файлы конфигурирования. Взломщик, как и в предыдущем случае, имеет в своем распоряжении сам кристалл, в котором скрыта информация о проекте. Чтение памяти конфигурации может быть запрещено битом секретности, сбросить который можно только при операции стирания всего содержимого этой памяти (заметим, что в литературе имеются сведения о случаях проникновения в запертую память конфигурации с помощью специальных электрических режимов микросхемы). Исследовать сам кристалл проще, чем кристалл с перемычками, но также очень нелегко. Если предположить, что расположение транзисторов с плавающими затворами, состояния которых программируются, известно или может быть визуально определено, то задача раскрытия проекта сводится к исследованию состояний каждого из многих тысяч или даже миллионов транзисторов. Наличие или отсутствие заряда в плавающем затворе можно выявить без разрушения кристалла несколькими способами. Заряды создают электрические поля, которые можно обнаруживать специальными методами, можно использовать также электронный микроскоп или материалы, накладываемые на кристалл и изменяющие цвет под воздействием электрических полей. Трудоемкость и стоимость исследования состояний плавающих затворов остаются все же очень высокими, и подобные исследования имеют смысл только для высокоценных проектов.

Далее, даже если станут известны местоположение и состояния всех транзисторов, то для простого клонирования проекта нужно транслировать эти сведения в битовый поток конфигурирования БИС/СБИС, а для реконструкции проекта еще и в саму схему. Решение этих задач очень сложно, поэтому *схемы с энергонезависимой памятью конфигурации можно считать хорошо, хотя и не абсолютно, защищенными от взлома.*

Самыми уязвимыми для взломщиков являются *БИС/СБИС программируемой логики с триггерной памятью конфигурации*, которую нужно загружать при каждом включении питания от внешнего источника хранимых данных. Для клонирования проекта достаточно прочесть содержимое этой внешней памяти и использовать его для конфигурирования клонов. Установление соответствия недокументированного битового потока конфигурирования и внутренней структуры схемы (реконструкция проекта) является более сложным, но не считается невозможным.

С целью повышения защищенности проектов, в первую очередь для БИС/СБИС с триггерной памятью конфигурации, принимается ряд мер: организационных, юридических, конструктивных и др.

К организационным мерам можно отнести договоренность с поставщиком о поставке немаркированных кристаллов, что эффективно затрудняет попытки взлома проектов.

К юридическим мерам можно отнести встраивание в проект некоторого недокументированного идентификатора (например, инициалов автора проекта).

К конструктивным мерам можно отнести покрытие кристалла и его связей с другими кристаллами непроницаемым слоем (например, эпоксидной смолой). Естественно, это имеет и отрицательные последствия, не позволяя в дальнейшем дорабатывать проект и ухудшая тепловой режим кристалла. Сходен с этим прием размещения трассы передачи битового потока конфигурирования между энергонезависимой памятью и микросхемой в скрытых внутренних слоях печатной платы.

Можно вообще обойтись без энергонезависимой памяти конфигурирования, если вместо нее разместить на плате автономный источник питания (литиевую батарею), который будет сохранять запрограммированную конфигурацию в самой триггерной памяти микросхемы. Однако в этой ситуации потребуются специальные схемы для изоляции источника автономного питания от всех цепей, кроме памяти конфигурации, что увеличивает площадь кристалла и снижает его быстродействие. Кроме того, при истощении батареи схема все же разрушится. Разрушение может произойти и от мгновенной потери питания вследствие удара, действия помехи и т. д.

В качестве приема защиты проекта используется и разделение его между частями с триггерной и энергонезависимой памятью конфигурации при наличии в обеих частях одного и того же уникального идентификатора, который должен согласовываться.

Можно использовать в ПЛИС с энергонезависимой и с триггерной памятью конфигурации сдвигающих регистров с линейными обратными связями, причем первая ПЛИС имеет целью только разрешать работу второй. Сложность расшифровки функционирования сдвигающего регистра связана с его структурой и разрядностью.

Существуют и другие возможности затруднить процесс взлома проектов, реализованных в БИС/СБИС программируемой логики [45].

1.10. Способы оценки параметров ПЛИС. Обзор новой продукции ведущих фирм-производителей ПЛИС

В силу своих достоинств ИС с программируемыми структурами занимают *быстро растущую долю рынка.*

Подводя итоги сказанному ранее, можно выделить *две крупнейшие положительные черты ПЛИС:* во-первых, *улучшение параметров* разрабатываемых устройств и систем, включая их компактность, быстродействие, надежность, стоимость, низкую потребляемую мощность, и, во-вторых, *чрезвычайную*

легкость и быстроту изменения проектов, малое время выхода продукции на рынок. Для БИС/СБИС с программируемыми структурами модификация реализуемых алгоритмов производится без изготовления наборов фотошаблонов.

Области применения интегральных схем разных видов в качестве окончательной продукции, как показано на рис. 1.33, зависят от тиражности их производства (предполагается, что тиражность растет в направлении слева направо).

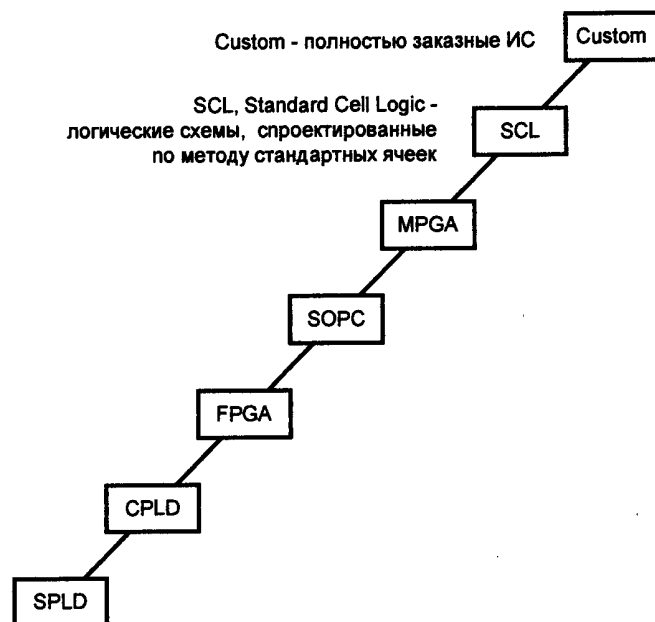


Рис. 1.33. Связь объемов производства изделий с элементной базой, целесообразной для их реализации

При объеме производства изделия меньше 50–70 тыс. шт. применение FPGA и CPLD, согласно мнению ряда специалистов, становится стоимостью-эффективной альтернативой для базовых матричных кристаллов и более сложных ASIC. Подробные соображения по выбору элементной базы при разных условиях разработки проектов приводятся в гл. 2.

Область применения программируемых схем в качестве прототипов при отработке систем значительно шире.

Микросхемы с программируемой структурой конкурируют со стандартными ИС, с базовыми матричными кристаллами и с заказными схемами, реализованными по методу стандартных ячеек. Активнее всего программируемые схемы отвоевывают области применения у стандартных ИС невысокого

уровня интеграции и у базовых матричных кристаллов. В меньшей степени они вытесняют ИС со стандартными ячейками, хотя, возможно, это положение вскоре может измениться. Нельзя не отметить, что в последнее время ИСПС хорошо дополняют схемы со стандартными ячейками, позволяя создавать системы разного назначения с помощью одного и того же набора фотошаблонов за счет модификации программируемой части кристалла. Подробнее об областях применения различных типов микросхем также говорится в гл. 2.

Достоинства ПЛИС рожают спрос на них, спрос стимулирует предложение, в результате появилось много разновидностей ПЛИС, в которых не просто ориентироваться без понимания особенностей различных архитектур. Знание архитектур и особенностей существующих ПЛИС нужно для успешного выполнения ряда этапов процесса проектирования устройств и систем на их основе, в частности этапа выбора элементной базы и оценки, а при необходимости, и "ручной" коррекции результатов работы компилятора САПР по синтезу схемной части проекта.

Свойства и возможности ПЛИС выражаются многими параметрами, поэтому разумно ранжировать их по степени важности для того или иного применения и, после выбора претендентов на использование, переходить к детальному ознакомлению с их характеристиками. Разнообразие и насыщенность все более расширяющегося рынка ПЛИС не отменяет, а даже обостряет для системотехников необходимость глубокого знания особенностей и сравнительных характеристик существующих микросхем. Подробное описание сложной программируемой микросхемы в фирменных справочниках может занимать десятки или даже сотни страниц, и его изучение требует немалого времени. Сравнительная оценка ПЛИС затрудняется не только многочисленностью их параметров, но и тем, что методика определения некоторых параметров у разных фирм не только различна, но порою приводит к существенно расходящимся результатам.

При выборе микросхем для применения в конкретном проекте приходится оценивать многие их параметры, и, как обычно в подобных ситуациях, возникают трудности учета разнородных характеристик. Тем не менее, не приходится говорить о формировании многокритериальной функции качества, поскольку практика проектирования не подтвердила, как и следовало ожидать, эффективности такого подхода.

1.10.1. Об оценке логической сложности (уровня интеграции) ПЛИС

Сложность программируемых микросхем с давних пор оценивается, как уже было сказано выше, с использованием понятия "число эквивалентных вентиля", причем под вентилем понимают обычно логический элемент 2И-НЕ.

В свое время программируемые схемы (базовые матричные кристаллы) содержали именно такие или близкие к ним вентили, поэтому подсчет числа вентилях на кристалле давал достаточно точную оценку его логической емкости (логической мощности, уровня интеграции). Однако для последующих ПЛИС, более разнообразных по схемотехнике, стало невозможным просто подсчитывать число вентилях на кристалле, поскольку схемы кристалла отнюдь не состояли из таких вентилях. На начальном этапе развития ПЛИС разноразной в подходе разных производителей к оценке логической сложности схем очень затруднял их сопоставление по этому параметру.

На ранней стадии развития ПЛИС упорядочить эту ситуацию удалось консорциуму компаний PREP (Programmable Electronics Performance Corporation). Согласно широко признанной в свое время методике подсчета по рекомендациям этого консорциума, оценка сложности ИС проводится с помощью набора эталонных схем. Для измерения сложности кристалла определяется максимально возможное число реализаций в нем каждой схемы из этого набора. Из практики работы с взятым для образца БМК сложность, выраженная в числе эквивалентных вентилях, для каждой эталонной схемы известна. Далее можно получить и сложность оцениваемого кристалла по данной эталонной схеме, определяемую как произведение ее сложности на максимально возможное число реализаций таких схем на кристалле. В качестве эталонных схем были выбраны 10 типовых функциональных узлов (регистры, счетчики, дешифраторы и т. д.). Усредненная по всем эталонным схемам величина характеризует итоговый (окончательный) показатель сложности микросхемы в целом.

Идея эталонного (тестового) набора функциональных узлов явилась основой и для созданных другими авторами, в частности в работе [4], методик измерения сложности (а также и быстродействия) программируемых микросхем.

Далее ситуация осложнилась, стали широко применяться крупнозернистые функциональные блоки сложной структуры, и вместе с ними появились спорные интерпретации различных параметров. Разноразной в методиках оценки логической емкости кристалла увеличивался, и параметр "число эквивалентных вентилях" становился все более шатким. Этот параметр, тем не менее, чаще всего фигурирует в качестве одного из основных в таблицах, приводимых в справочниках фирм-изготовителей ПЛИС, хотя нередко вместо этой единой оценки сложности кристалла производители стали указывать отдельно размерности разных по функциональному назначению областей кристалла или даже переходить к прямому сравнению своей продукции со сходной продукцией других фирм, выполняемому по выбранным конкретным ситуациям.

Сложность ПЛИС первых поколений достаточно точно оценивалась числом эквивалентных вентилях в двух вариантах — общим числом (Total) и/или пользовательским (Usable), поскольку не все элементы схемы кристалла мо-

гут быть использованы системотехником в его проекте. К сожалению, не всегда в документации на схему четко указывается, о каком варианте оценки идет речь (и это относится не только к параметру "число эквивалентных вентилях"). В последнее время, как правило, приводится число пользовательских эквивалентных вентилях. В дальнейшем изложении определение "эквивалентные" для краткости будет иногда заменяться термином "число вентилях".

В ходе дальнейшего развития ПЛИС положение с оценкой их сложности осложнилось еще более. В качестве функциональных блоков стали применять табличные преобразователи (LUT-блоки), которые допускают неоднозначное использование — они способны воспроизводить логические функции или работать в качестве блоков памяти с той или иной организацией. В зависимости от режима работы существенно меняется и оценка сложности одного и того же фрагмента схемы, поскольку эквивалентные реализации разных режимов работы LUT-блоков средствами БМК сильно различаются по сложности (в несколько раз). На этом этапе возникли понятия числа логических вентилях и числа системных вентилях. При подсчете числа логических вентилях предполагается работа функциональных блоков в режиме воспроизведения логических функций. При подсчете числа системных вентилях предполагается, что часть функциональных блоков работает в режиме блоков памяти и, следовательно, имеет другую эквивалентную сложность. Естественно, числа логических и системных вентилях в одной и той же микросхеме не совпадают.

Позднее положение с оценкой сложности ПЛИС дополнительно осложнилось из-за появления на кристалле специализированных областей встроеной памяти и других структурных неоднородностей. Встроенная память, в частности, также может применяться как по прямому назначению, так и для воспроизведения логических функций табличным методом.

Подсчитывая число вентилях в своих микросхемах, три крупнейших производителя ПЛИС занимают неодинаковые позиции. К методике оценки логической емкости программируемой микросхемы, принятой фирмой Xilinx, склоняется также фирма Altera, тогда как фирма Actel пользуется другой методикой, приводящей к иным результатам, и считает показатели фирм Xilinx и Altera завышенными. Например, фирмы Actel и Cypress, указав для одной из своих ПЛИС семейства rASIC сложность в 12 тыс. эквивалентных вентилях, сопроводили свою оценку замечанием, что по методике измерений фирм Altera и Xilinx это соответствовало бы сложности в 36 тыс. эквивалентных вентилях. Из сказанного следует, что при рассмотрении параметра сложности ПЛИС нужно внимательно следить за тем смыслом, который вкладывается в каждом конкретном случае в приводимый показатель.

Методика фирмы Altera основана на сравнении аппаратных затрат при реализации одних и тех же устройств в двух вариантах: на оцениваемой ПЛИС

и на известном базовом матричном кристалле LCA300K, логической основой которого являются вентили, подобные принятым в качестве эквивалентных.

Основная проблема оценки логической емкости ПЛИС создается наличием в ней как логической части (FPGA, CPLD), так и модулей памяти. Для оценки логической емкости микросхемы в этих условиях используются несколько параметров [4]:

- емкость логического массива;
- логическая емкость массива памяти при реализации логических функций;
- логическая емкость массива памяти при реализации запоминающих устройств;
- диапазон изменения логической емкости;
- типичная логическая емкость.

Емкость логического массива определяется как произведение логической емкости одного логического элемента q на их число M_1 в данной ПЛИС. Для нахождения емкости одного логического элемента рассматриваются схемы, которые могут быть в нем реализованы. Эти схемы сопоставляются с библиотечными модулями эталонного БМК (LCA300K), что дает их оценку в числах эквивалентных вентилях. За логическую емкость элемента q принимается число эквивалентных вентилях, усредненное по всему множеству проанализированных схем. Таким образом, емкость логического массива $L_1 = qM_1$.

Логическая емкость массива памяти при реализации логических функций L_2 определяется как произведение логической емкости модуля памяти p на число M_2 таких модулей в данной ПЛИС. Параметр p определяется опять-таки на основе анализа библиотечных модулей БМК — это среднее число эквивалентных вентилях, необходимых для воспроизведения функций, реализуемых модулем памяти. Таким образом, $L_2 = pM_2$.

Логическая емкость массива памяти при реализации запоминающих устройств L_3 определяется как произведение числа s вентилях БМК, необходимых для создания схемы хранения одного бита данных, на емкость M массива памяти ПЛИС. Таким образом, $L_3 = sM$.

Диапазон изменения логической емкости. Значение логической емкости ПЛИС, как можно видеть, зависит от способа использования модулей памяти. Оценки сложности отдельных схемных элементов, принятые фирмами Altera и Xilinx, таковы, что вариант с использованием памяти по прямому назначению даст верхнюю границу диапазона значений логической емкости ПЛИС, а вариант с использованием всей памяти для воспроизведения логических функций — нижнюю. Следовательно, минимальная логическая емкость ПЛИС $L_{\min} = L_1 + L_2$, а максимальная $L_{\max} = L_1 + L_3$.

Типичная логическая емкость. Для оценки типичного значения логической емкости кристалла привлекаются статистические данные о доле массива памяти, используемой для реализации логических функций, усредненные по множеству проанализированных проектов. Как правило, эта доля лежит в пределах 15—35%. Обозначив эту долю через d , для нахождения типичной логической емкости кристалла $L_{\text{тип}}$, выраженной в числе системных вентилях, можно записать соотношение $L_{\text{тип}} = L_1 + dL_2 + (1 - d)L_3$.

Приведенные формулы справедливы для оценки основной части ПЛИС, занятой функциональными ячейками и памятью. Наличие областей с блоками ввода/вывода и другими блоками в некоторой мере изменяет соотношение между указанными параметрами.

В справочных данных фирм часто указывается *типичный диапазон числа вентилях*, в котором нижняя граница соответствует числу вентилях в составе функциональных блоков, работающих в режиме воспроизведения логических функций, а верхняя определяется при указании доли этих блоков, реализующих функции памяти (обычно это 20—30%).

Укажем для примера цифры, полученные фирмами Altera и Xilinx для своих популярных семейств. Фирма Altera для семейства FLEX10K приняла следующие параметры: $q = 12$, $p = 150$, $s = 4$. Фирма Xilinx для семейства Virtex II считает каждую логическую ячейку в логическом режиме соответствующей 12 системным вентилям, а в режиме распределенной памяти — 64-м (16 бит $\times$ 4 ЛЭ/бит). Каждая схема автоподстройки по задержкам синхросигналов DLL дает 7К вентилях, каждая логическая ячейка в режиме программируемого элемента задержки — 112 вентилях, каждый бит блока памяти — 4 системных вентилях.

Цифра 4 как оценка эквивалента одного бита памяти при реализации памяти с помощью логических вентилях, принятая многими фирмами, объясняет резкий рост оценок уровня интеграции современных ПЛИС относительно их недавних предшественников. Действительно, введение в состав ПЛИС относительно больших массивов встроенной памяти при оценке их сложности числом эквивалентных вентилях путем умножения емкости в битах на четыре сразу дает большую прибавку к оценке уровня интеграции. Иными словами, скачкообразный рост расчетного уровня интеграции современных сложных ПЛИС только отчасти объясняется совершенствованием процессов их производства и во многом стал следствием принятой методики оценки логической емкости микросхем.

При выборе подходящей ПЛИС для своего проекта системотехник не может полагаться на какой-либо параметр микросхем или даже 2—3 параметра. Проблема выбора достаточно сложна и требует проведения квалифицированного анализа совокупности параметров ПЛИС с учетом специфики подлежащего реализации проекта.

Почему нельзя полагаться на параметр "число вентиляей"? Во-первых, из-за различия архитектур ПЛИС. Логические блоки ПЛИС уже не являются "мелкозернистыми" и представляют собой довольно сложные схемы. *Какой будет эффективность использования логических ресурсов кристалла при реализации данного проекта?* Называя цифру "число вентиляей", производители ПЛИС часто имеют в виду 100% их использования (некоторые фирмы указывают более скромные цифры). На самом деле, в составе сложных логических блоков может оказаться значительная доля неиспользуемых ресурсов. Кроме того, *проект может исчерпать трассировочные ресурсы раньше, чем будет использована имеющаяся на кристалле логика*, и, таким образом, часть логических ресурсов "пропадет".

Очень спорны *вопросы пересчета встроенной (дискретной) и распределенной памяти кристалла*, т. е. памяти, реализованной в виде специальных блоков, и находящейся в составе функциональных блоков типа LUT, в число эквивалентных вентиляей. В этом случае получаемое число эквивалентных вентиляей резко зависит от того, в каком качестве будет служить память. Если она применяется по своему назначению для создания той или иной конфигурации ЗУ (очень часто в проектах интенсивно используются блоки FIFO), то ее присутствие сильно увеличивает оценку числа эквивалентных вентиляей на кристалле. А какую роль будет играть память в реализуемом проекте — это еще вопрос.

Далее, большую роль играет *технология проектирования схем*. Например, схема может быть реализована с помощью неэффективного компилятора САПР или же вручную доработана опытным специалистом. Ясно, что затраты оборудования на реализацию проекта для этих вариантов могут существенно отличаться. Ясно также, что, пользуясь оценками по затратам оборудования на построение разных функциональных узлов, фирмы-производители обеспечивали высококачественное проектирование. Сможет ли потребитель для своего проекта обеспечить такое же качественное проектирование?

Кроме того, *разные производители включают в состав средств для счета вентиляей разные ресурсы*. Могут быть включены или не включены в этот состав буферные элементы, схемы граничного сканирования по интерфейсу JTAG, дополнительные схемы быстродействующих декодеров и т. д. Понятно, что это тоже вносит разночтения в приводимые оценки.

Отмеченные обстоятельства не исчерпывают проблематику сложности оценки логической емкости кристалла, достоверной для конкретного проекта, но убедительно свидетельствуют о *сугубо ориентировочном* характере группы параметров, связанных с понятием "число эквивалентных вентиляей".

Наряду с числом эквивалентных вентиляей в числе параметров ПЛИС *указываются числа макроячеек* (для CPLD), *логических блоков или логических ячеек* (для FPGA и микросхем комбинированной архитектуры). Сведения об этих

параметрах, пожалуй, более репрезентативны, т. к. говорят о числе функций, которые могут быть воспроизведены данной микросхемой. Однако и здесь не все однозначно, поскольку воспроизводимые функции неидентичны, и малое число сложных функций может дать больше функциональных возможностей, чем большое число простых. Кроме того, в схемах используются различные функциональные расширители, цепи быстрых переносов и каскадирования и другие фрагменты, которые не отображаются как самостоятельная логика, но могут существенно влиять на функциональные возможности кристалла, причем степень такого влияния опять-таки зависит от характера конкретного проекта.

В связи со сказанным в приведенном ниже обзоре новых микросхем с программируемой структурой даны сведения о них как по числу вентиляей, так и по числу логических элементов (макроячеек, логических блоков).

1.10.2. Об оценке быстродействия ПЛИС

Быстродействие ПЛИС характеризуется либо задержками распространения сигналов по указанным путям, либо максимальными частотами их работы в указанных условиях.

Пути распространения сигналов в классических CPLD жестко предопределены их архитектурой и *задержки по путям от любого входа до любого выхода* (Pin-to-pin, Corner-to-corner, Clock-to-pin) *могут достаточно хорошо характеризовать быстродействие CPLD*.

Пути распространения сигналов в FPGA разнообразны, и для них с целью оценки быстродействия целесообразно пользоваться максимально возможной частотой тактирования, причем эта частота может быть указана для различных схем. Часто указывают либо максимально возможную *частоту работы счетчика* F_{CNT} , либо частоту работы (тактирования) схемы в целом F_s , называемую *системной*. Обычно системная частота приблизительно вдвое ниже, чем частота работы счетчика.

Могут быть указаны и другие частоты, например, *частота тактирования передач типа "регистр-регистр"* F_r , или частота F_{max} , называемая просто *"максимальной"* без конкретного уточнения схемы, к которой она относится. Именно эта частота фигурирует в описаниях многих микросхем, хотя ясно, что без указания схемы, о тактировании которой идет речь, приводимая цифра теряет смысл. Иногда понятно, как следует трактовать величину F_{max} (например, в случае получения результата при однократном прохождении сигналов от входов к выходам микросхемы). Но в более сложных ситуациях (например, при использовании в схеме обратных связей) требуется конкретная оценка максимальных частот тактирования для данного устройства, реализованного в той или иной микросхеме. Тракт прохождения сигналов в цифровом устройстве можно представить в виде последовательности секций

"регистр-комбинационная схема-регистр". Согласно формулам расчета параметров синхросигналов для устройств, содержащих подобные секции [27], минимальный период синхроимпульсов для однофазной системы синхронизации определяется как

$$T = 2\Delta + T_{su} + T_{тг.мах} + T_{кц.мах},$$

где Δ — нестабильность положения фронта синхроимпульса во времени, T_{su} — параметр предустановки информационных сигналов на входах триггеров регистра-приемника, $T_{тг.мах}$ — максимальная задержка от поступления на вход триггера регистра-передатчика тактового сигнала до установления данных на его выходах, $T_{кц.мах}$ — максимальная задержка сигнала в комбинационных схемах и цепях связи между регистром-передатчиком и регистром-приемником.

Величина Δ есть "поле допуска" для временного положения фронта синхросигналов, т. е. интервал появления фронта синхросигнала задается от $t - \Delta$ до $t + \Delta$, где t — теоретический момент появления фронта синхроимпульса.

Минимальный период тактирования схемы при данных параметрах (T_{su} и $T_{тг.мах}$) синхронизируемых триггеров (регистров) и нестабильности фронтов синхросигналов зависит от $T_{кц.мах}$, т. е. от задержки выработки логической функции комбинационной схемой и задержки в цепях связи для соответствующей секции (яруса схемы или ступени конвейера). Секция с максимальным значением T ограничит возможную частоту тактирования устройства. Максимальная частота синхронизации $F_{\max} = 1/T$.

Нередко в сведениях о ПЛИС приводятся тактирующие частоты для синхронизации внутреннего ядра и цепей ввода/вывода микросхемы. Встречаются различные соотношения между этими частотами. Внутренняя частота при многоразрядных трактах передач информации, как правило, выше, чем частота ввода/вывода, но в других ситуациях может наблюдаться и противоположная картина. В частности, в новых сложных ПЛИС все чаще используются линии последовательного ввода/вывода, которые в сочетании с мало-сигнальными дифференциальными схемами передачи сигналов (например, по стандарту LVDS) обеспечивают чрезвычайно высокие скорости обмена. В таких случаях даже при параллелизации данных внутри приемника скорости передач могут оказаться приемлемыми. Например, для линии со скоростью передачи 2,5 Гбит/с переход к байт-последовательной передаче даст частоту передачи байтов выше 300 МГц. Четыре таких скоростных линии последовательной передачи смогут обеспечить передачу с частотой 300 МГц 32-разрядных слов, а 8 линий смогут передавать 32-разрядные слова с частотой 600 МГц. Преобразования данных из последовательной формы в параллельную и наоборот связаны с применением сдвигающих регистров типа SIPO (Serial Input — Parallel Output) или PISO (Parallel Input — Serial Output). Естественно, при этом требуется строгая синхронность подачи последовательных данных и синхросигналов, управляющих сдвигами данных в реги-

страх. При работе на высоких частотах обеспечение синхронизации становится сложной проблемой. Для ее решения в состав соответствующих ПЛИС включают блоки CDR (Clock Data Recovery).

Оценка быстродействия ПЛИС дается также с применением понятия *градация быстродействия*. Этот параметр входит в состав кода обозначения схемы в виде цифры, перед которой ставится дефис (-3, -4, -5 и т. д.). Для архитектур типа CPLD с характерной для них независимостью задержек распространения сигналов от их конкретных путей цифра, приводимая на позиции "градация быстродействия", приблизительно совпадает с округленным значением задержки распространения сигнала через данную ПЛИС. Для архитектур типа FPGA и комбинированных архитектур цифра, определяющая градацию быстродействия, имеет лишь относительное значение. Она ранжирует микросхемы по быстродействию, но не указывает абсолютные значения динамических параметров, являясь, таким образом, как бы указателем для их поиска в таблице. Указание градаций быстродействия целесообразно для сравнительной оценки ПЛИС одного и того же семейства, но будет бесполезно для сравнения микросхем разной архитектуры [4]. Для сравнения разнотипных ПЛИС лучше оценивать тактовые частоты устойчивой работы типовых функциональных узлов, реализованных в данной микросхеме.

Важной экономической характеристикой ПЛИС является их *стоимость*. Стоимость ПЛИС, как и других БИС/СБИС, меняется со временем. Цена новых сложных ПЛИС вначале высока, но довольно быстро снижается и затем устанавливается на относительно стабильном уровне, испытывая лишь небольшие конъюнктурные колебания. Стоимость микросхем во многом определяется типом их корпусов — в зависимости от этого фактора цена одного и того же кристалла может измениться более чем в 10 раз. Значительно повышаются и цены микросхем, имеющих специальную приемку (военную и др.). Стоимость, как и для других видов продукции, зависит от размера приобретаемой партии микросхем (лота) и ряда других факторов. Кстати говоря, одним из преимуществ ПЛИС перед БМК и другими микросхемами, объявляемым многими фирмами-производителями, является отсутствие минимального порога при их заказе: допускается заказ любого количества микросхем.

1.10.3. Сведения о современной продукции крупнейших фирм-производителей микросхем с программируемой структурой

В заключение главы приведем краткую сводку сведений о продукции основных производителей ИС с программируемыми структурами. В материалах этого раздела использованы справочные данные фирм-производителей, а также обзора [44].

Сведения, не включающие описаний блочных "систем на кристалле" с процессорными ядрами

Ниже приводятся рисунки и комментарии к ним, отображающие основные параметры и свойства ИСПС. На рисунках представлены только новые семейства, перспективные для будущих разработок. Не показаны семейства, которые пока не сняты с производства и могут применяться по тем или иным соображениям, но имеют более современных преемников. Сведения о некоторых таких семействах, не показанных на рисунках, даются в тексте. Хотя приводимые ниже материалы освещают достаточно широкую область в производстве ИСПС, они все же не являются полностью исчерпывающими и не претендуют на описание всех без исключения разработок.

Микросхемы сгруппированы по признаку кратности программирования (т. е. по типу памяти конфигурации), а внутри этих групп по фирмам-производителям (маркировки микросхем не приводятся). При первом упоминании фирмы указывается и ее интернет-адрес. Каждое семейство ИСПС изображается горизонтально расположенной прямоугольной областью с несколькими поперечными вертикальными линиями, которым соответствуют микросхемы-члены семейства. Против каждой вертикальной линии сверху (а при нехватке места, и снизу) проставлена цифра, указывающая число эквивалентных вентилях в данной микросхеме. Если в справочных данных для микросхемы имелась конкретизация параметра "число вентилях", то это отображается и на рисунке, где приняты следующие обозначения: N_v — число эквивалентных вентилях, $N_{вс}$ — число системных эквивалентных вентилях, $N_{в\text{тип}}$ — типовое число эквивалентных вентилях. Данные о числе вентилях приводятся для всех членов семейства. Рядом с названием семейства записывается напряжение питания ядра микросхемы. Левее прямоугольника, при необходимости (если на одном и том же рисунке встречаются разнотипные архитектуры ИСПС), указывается архитектура микросхемы.

Остальные параметры приводятся только в виде диапазона их значений для данного семейства (указываются значения для младшего и старшего членов семейства).

Сведения о быстродействии включают в себя указание на ту или иную максимально допустимую частоту. Системная частота F_c или F_{max} — максимальные частоты, на которых могут работать собранная в данной микросхеме сложная схема, содержащая множество функциональных блоков, или (если это отмечено) конкретное устройство, F_{CNT} — максимальная частота работы счетчика (чаще всего в качестве испытательного образца берется 16-разрядный двоичный счетчик), $F_{\text{т}}$ — максимальная частота передач "регистр-регистр".

К параметрам, задаваемым диапазоном значений для всего семейства, могут относиться следующие:

- число макроячеек $N_{\text{мя}}$ (для CPLD) или логических элементов $N_{\text{лэ}}$ (для FPGA);

- суммарная емкость памяти $M_{\text{ит}}$ в схемах табличных функциональных преобразователей (блоков типа LUT);
- суммарная емкость встроенных блоков памяти $M_{\text{вбп}}$;
- число пользовательских вводов/выводов $N_{\text{вв}}$;
- стоимость микросхем, \$.

Число вентилях измеряется в тысячах, емкости блоков памяти — в битах. Диапазон значений $N_{\text{вв}}$ дается с учетом разных вариантов корпусов микросхем, иначе говоря, к различным цифрам могут относиться и разные типы корпусов, в которых выпускаются микросхемы данного семейства.

Стоимость микросхем приводится по американским данным 2001 года для лотов в 10 тыс. шт., для самых дешевых корпусов, схем минимального быстродействия и коммерческого диапазона температур.

В виде отдельных примечаний для некоторых семейств, обладающих ярко выраженными особенностями, они указываются текстовой надписью.

На рис. 1.34 приведены сведения об однократно конфигурируемых FPGA с программированием перемычек типа antifuse.

В этом секторе рынка доминируют фирмы Actel и QuickLogic. Полезные свойства ПЛИС с перемычками типа antifuse — невысокая стоимость, высокое быстродействие, высшая степень засекречивания проекта, повышенная стойкость к воздействию радиации. Недостаток — однократность программирования. Естественно, конфигурация схем сохраняется при снятии питания микросхемы.

Большинство выпускаемых FPGA с перемычками типа antifuse имеют уровень интеграции, оцениваемый числом вентилях от 3 до приблизительно 100 тыс. Исключение составляет семейство Eclipse фирмы QuickLogic с уровнем интеграции до 580 тыс. эквивалентных вентилях. Отличительными чертами FPGA семейства Eclipse, кроме высокого уровня интеграции, можно считать большие возможности входных и выходных буферов. В них введены триггеры для фиксации входных и выходных сигналов, а также сигналов управления третьими состояниями буферов, поддерживается множество стандартов выходных сигналов, в том числе и для дифференциальных передач. Введены 4 схемы PLL, реализован большой объем встроенных блоков двухпортовой памяти RAM.

В группе реконфигурируемых ПЛИС с электрическим программированием состояний транзисторов типа ЛИЗМОП представлена продукция 5 фирм (рис. 1.35).

Память конфигурации типов EEPROM и Flash характерна почти исключительно для CPLD, где эти типы памяти сменили применявшуюся ранее память типа EPROM. Пионер в разработке CPLD — фирма Altera, впервые

Фирма Actel
www.actel.com

Семейство **аХ**; 2,5 В

Nв	3	6	12
Nлз	192		
Nвв	49		
			\$ 2,30 ... 6,30
			Особенность - малое энергопотребление

Семейство **СХ**; 3,3 В
Семейство **СХ-А**; 2,5 В

Nв	12	24	48	108 (только СХ-А)
Nлз	768			6036
Nвв	84			329
				Fc = 260 МГц
				\$ 4,5 ... 23,3

Семейство **МХ**;
3,5 В и 5,5 В

Nв	3	6	14	24	36	54
Nлз	295					2438
Nвв	44					256
						Fc = 260 МГц
						\$ 4,40 ... 23,10

Особенность - наличие блока памяти 2560 бит

Фирма QuickLogic
www.quicklogic.com

Семейство **РАSIC2**;
3,3 В и 5,0 В

Nв	3	5	7	9
Nлз	192			
Nвв	84			
				672
				256
				\$ 14,9 ... 59,8

Семейство **РАSIC3**;
3,3 В

Nв	4	12	25	40	60
Nлз	96				
Nвв	68				
					1584
					456
					Fc = 400 МГц
					\$ 4,5 ... 36,9

Семейство **Eclipse**;
2,5 В

Nв	250	320	500	580
Nлз	960			
Nвв	208			
				4032
				672
				Fc = 400 МГц
				\$ 16 ... 80

Имеются блоки памяти от 46080 до 82944 бит

Рис. 1.34. Основные параметры однократно конфигурируемых FPGA фирм Actel и QuickLogic

Фирма Altera
www.altera.com

Семейство **МАХ3000А**;
3,3 В

Nвв	0,6	1,25	2,5	5
Nлз	32			
Nвв	44			
				256
				Fmax = 160 ... 190 МГц
				\$ 0,95 ... 10

Семейство **МАХ7000АВ,С**;
А - 3,3 В; В - 2,5 В; С - 5,0 В

Nв	0,6	1,25	2,5	3,2	3,75	5
Nлз	32					
Nвв	44					
						512
						208
						Fmax = 130 ... 175 МГц
						Fmax = 285 МГц (для В)
						\$ 1,40 ... 34

Семейство **МАХ9000**;
5 В

Nв	6	8	10	12
Nлз	320			
Nвв	59			
				560
				216
				Fmax = 125 МГц
				\$ 30 ... 65

Фирма Xilinx
www.xilinx.com

Семейства **XC9500, XC9500XL**;
5 В; 3,3 В; 2,5 В

Nв	0,8	2,4	4,8	6,4	9,6
Nлз	36				
Nвв	44				
					288
					352
					Fc = 60 ... 100 МГц
					\$ 2,85 ... 25,95
					\$ 0,99 ... 15,50 (XL)

Семейство **CoolRunner**;
от 2,7 до 3,6 В

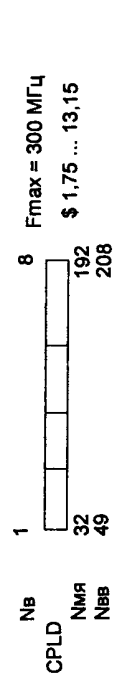
Nв	32	48	584	280
Nлз	48			
Nвв	48			
				584
				280
				Fc = 60 ... 100 МГц
				\$ 2,85 ... 25,95
				\$ 0,99 ... 15,50 (XL)

Особенность - чрезвычайно малая потребляемая мощность

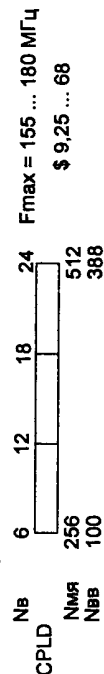
Рис. 1.35. Основные параметры реконфигурируемых ИСПС с электрическим программированием зарядов в плавающих затворах ЛИЗМОП-транзисторов (ИСПС с теневой памятью типов EEPROM и Flash) (см. продолжение)

Фирма Lattice Semiconductor
www.latticesemi.com

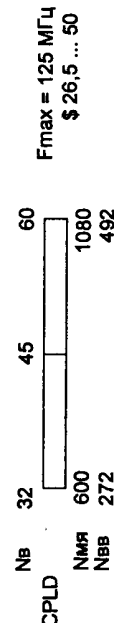
Семейство l9pLSI2000VE, 3,3 В



Семейство l9pLSI5000VE, 3,3 В

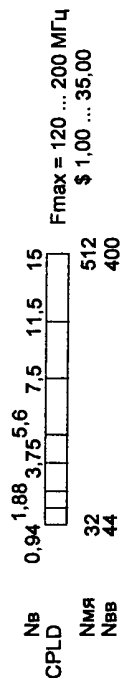


Семейство l9pLSI 8000V, 3,3 В



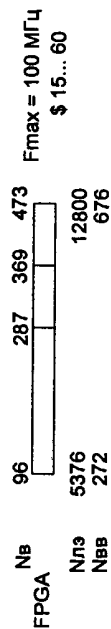
Фирма Cypress Semiconductor
www.cypress.com

Семейство Ultra 37000, 3 В; 5 В



Actel
www.actel.com

Семейство ProASIC500K, 2,5 В



Имеется память 13824 ... 46080 бит

Рис. 1.35. Окончание

реализовавшая их в конце 1980 гг. Основой большинства современных CPLD этой фирмы явилась архитектура семейства MAX7000, вслед за которой появились модификации, развивавшиеся в двух направлениях.

В рамках первого направления возросшие технологические возможности использовались для увеличения уровня интеграции микросхем, числа их макроячеек и быстродействия при одновременном снижении напряжений питания, потребляемой микросхемами мощности и стоимости в расчете на одну макроячейку. Начиная с семейства MAX7000S, микросхемы снабжаются средствами программирования в системе (ISP). Отличительной чертой микросхем становится также большое разнообразие в выборе стандартов для сигналов ввода/вывода, уровни которых могут быть как ниже, так и выше рабочего напряжения ядра (Multivolt I/O Technology).

Семейство MAX9000 также ориентировано на повышение уровня интеграции. Его старший представитель имеет 12 тыс. макроячеек против 5 тыс. старшего представителя семейства MAX7000. Однако при этом пришлось перейти к иерархической системе межсоединений, что привело к утере ценного свойства CPLD классической структуры — предсказуемости и постоянства задержек сигналов независимо от их конкретного пути.

Второе направление, представленное семейством MAX3000A, связано со стремлением к удешевлению микросхем. Семейство MAX3000A отличается от базового семейства MAX7000 снижением требований к некоторым параметрам схемы, прежде всего, к потребляемой ими мощности. Микросхемы семейства MAX3000A тестируются по пониженным нормам, уменьшен выбор типов их корпусов, реализованы не все возможности базового семейства MAX7000. Следствием стало снижение цен почти вдвое.

Фирма Cypress Semiconductor в секторе продукции CPLD с памятью конфигурации EEPROM представлена семейством микросхем Ultra37000. Это семейство реализовано на основе архитектуры семейства MAX7000 фирмы Altera, но, как указывает фирма Cypress Semiconductor, имеет ряд усовершенствований в сравнении с базовым вариантом. Изменения касаются временных параметров схем, у них отсутствует ощутимое влияние коэффициента разветвления на задержку распространения сигналов, снижены задержки, вносимые логическими расширителями, отсутствует рост задержек сигналов с увеличением числа термов макроячеек вплоть до 16 и др.

Фирма Atmel известна как поставщик популярных PAL типов 16V8, 20V8, 22V10 (в этой главе из-за широкой известности эти PAL подробно не рассматривались). В области CPLD высокого уровня интеграции фирма производит семейство ATF1500. Семейство ATF1500, как и семейство Ultra37000 фирмы Cypress Semiconductor, сходно с семейством MAX7000 фирмы Altera и конкурирует с ним. В микросхемах фирмы Atmel увеличено число входов у логических блоков и ресурсы трассировки. Фирма Atmel считает, что в ее микросхемах можно размещать более сложные проекты, чем в микросхемах

семейства MAX7000 со сравнимым числом макроячеек. В микросхемах семейства ATF1500 используется память конфигурации типа Flash.

Фирма Xilinx подключилась к производству CPLD лишь во втором поколении своих ПЛИС и в ограниченном масштабе. Ею выпущено семейство XC9500 с памятью конфигурации типа Flash. Схемотехника микросхем семейства XC9500 рассматривалась выше в качестве примеров решений, характерных для CPLD.

Фирма Lattice Semiconductor стала в области ИСПС также и преемником фирмы AMD в связи с приобретением у нее подразделения Vantis, с давних пор занимавшегося производством PAL и CPLD. Эта фирма известна, в частности, тем, что впервые применила программирование в системе (ISP) для ПЛИС невысокого уровня интеграции. С тех пор активность фирмы в области ISP подчеркивается даже в названиях ее микросхем. В сфере CPLD продукция фирмы делится на секторы: особо быстродействующей (SuperFast), особо "широкой" (SuperWide) и особо сложной (SuperBig).

Особо быстродействующие микросхемы второго поколения семейства ispLSI2000VE имеют время распространения сигнала от входа к выходу около 3 нс и максимальную частоту 300 МГц, особо "широкие" микросхемы семейства ispLSI5000VE имеют по 68 входов на каждый логический блок (однако в один логический блок этих микросхем входят 32 макроячейки, так что отношение "число входов блока/число макроячеек в блоке" не является столь уж большим и даже уступает этому показателю некоторых других CPLD). В особо сложных микросхемах семейства ispLSI8000 введена двухуровневая система межсоединений. Средства локальной трассировки объединяют по 20 макроячеек в так называемые мегаблоки, а 6 мегаблоков связываются между собой средствами глобальной трассировки. Микросхемы имеют встроенные шины с тремя состояниями и 108 линиями. Буферы ввода/вывода имеют триггеры для возможности выбора комбинационного или регистрового варианта для всех сигналов входов, выходов и управления двунаправленностью.

Фирма Lattice Semiconductor выпускает также микросхему специальной архитектуры ispG, в которой мало логических элементов и почти все ресурсы кристалла отданы под программируемые средства коммутации. Напомним, что фирма Lattice Semiconductor принадлежит также к пока малому числу производителей аналоговых схем с программируемыми структурами (ПАИС).

Фирма Actel, наиболее известная как производитель ПЛИС с перемычками типа antifuse, приняла участие и в выпуске реконфигурируемых микросхем, что связано с покупкой ею фирмы Gatefield Corporation с ее производством семейства ProASIC500K. По своей архитектуре микросхемы ProASIC500K относятся к FPGA (это единственный представитель класса FPGA среди микросхем с памятью конфигурации типа EEPROM или Flash). Производст-

вом микросхем ProASIC500K с Flash-памятью конфигурации фирма Actel поддерживает концепцию тесной преемственности FPGA и БМК, заботясь об их подобии. Микросхемы ProASIC500K имеют крайне мелкозернистые логические ячейки, что позволяет в значительной мере сохранить для этого семейства методику и инструменты проектирования, освоенные ранее для БМК.

В группу оперативно репрограммируемых ИСПС входит большое число микросхем разной архитектуры и разных уровней интеграции (рис. 1.36).

Фирма Xilinx
www.xilinx.com

Семейство Spartan II;
2,5 В

FPGA	№	15	30	50	100	150	200	F _c = 200 МГц \$ 5,25 ... 19,45
	Нлэ							
	КЛБ							
	Мвбп							
	Нвв							
		432					5292	
		96					1176	
	16384						57344	
	84						256	

Семейство Virtex E/EM;
1,8 В

FPGA	№	1569	2189	2542	3264	4074	F _c = 200 МГц \$ 20 ... 3100
	Нлэ	1728				73008	
	Млут	24576				1038336	
	Мвбп	16384				1146880	
	Нвв	144				1152	

Семейство Virtex II;
1,5 В

№	40	1000	6000	10000	Fc = 200 МГц \$ 13,25 ... 1834
FPGA					
Nn3	576			138240	
Mlut	8192			1966080	
Mb6n	75728			3538944	
NvB	144			1152	

Рис. 1.36. Основные параметры оперативно репрограммируемых ИСПС с триггерной памятью конфигурации (см. продолжение)

Фирма Altera
www.altera.com

Семейство ACEX1K;
2,5 В

На, тип	10	30	50	100	
КА					
Нля	576			4992	
Мвбп	12288			49152	
Нвв	100			672	

F_{CNT} = 200 МГц
\$ 3 ... 13

Семейство APEX20K/E;
2,5 В

На, тип	30	100	200	400	1000	1500	
SOPC							
	60	160	300	600			
Нлз	1200					51840	
Нмя	192					3456	
М, бит	24576					442368	

F_c = 200 МГц
\$ 44 ... 440

Семейство APEXII;
1,5 В

На, тип	600	900	1500	3000	4000	
SOPC						
Нлз	16640				89260	
М	465964				1523712	
Каналов LVSD	92				134	
Нвв	672				1508	

F_c = 270 МГц
\$ 45 ... 820

Семейство Mercury;
1,8 В

На	120		350	
SOPC				
Нлз	4800		14400	
М	49152		114688	
Каналов HSDI	8		16	
Нвв	484		780	

F_{CNT} = 333 МГц
\$ 120 ... 350

Рис. 1.36. Продолжение

Фирма Atmel
www.atmel.com

Семейство AT6000;
3 ... 5,5 В

На	6	9	15	30	
FPGA					
Нлз	1024			6400	
Нвв	84			240	

F_c = 100 МГц
\$ 6 ... 40

Особенность - реализуемость
кэш-логики

Семейство AT40K;
3 ... 5,5 В

На	10	20	30	50	
FPGA					
Нлз	256			6400	
Нвв	84			352	

F_c = 100 МГц
\$ 5 ... 50

Особенность - реализуемость
кэш-логики

Имеется память объемом
2048 ... 18432 бит

Фирма Cypress
www.cypress.com

Семейство Delta 39K;
3,3 В; 2,5 В; 1,8 В

На	15	50	165	250	350	
CPLD						
Нв	30	100	200			
Нмя	256				5376	
Нвв	208				676	

F_c = 250 МГц
\$ 18 ... 145

Имеется память объемом
40960 ... 860160 бит

Рис. 1.36. Продолжение

Фирма Agere Systems
www.agere.com

Семейство ORCA3;
5 В; 3,3 В; 2,5 В

FPGA	Nbc	244	340	FCNT = 176 МГц \$ 9,6 ...206
				
	Nlut	8192	11552	
	Mlut	131000	185000	
	Nвв	208	680	

Семейство ORCA4;
1,5 В

	Nbc	470	720	970	1350	1700	Fc = 250 МГц
FPGA							
	Nlut	4992				39960	\$ 41 ... 697
	Nbp	73728				221184	
	Nbv	400				1068	

Особенность семейств - наличие встроенного интерфейса для процессора

Рис. 1.36. Окончание

Среди ИСПС с триггерной памятью конфигурации преобладают архитектуры FPGA и комбинированные архитектуры, а по уровню интеграции эти схемы располагаются в диапазоне от нескольких тысяч до 10 млн. эквивалентных вентилей. Последняя цифра относится к микросхемам типа "системы на кристалле".

Оперативное репрограммирование появилось в разработках фирмы Xilinx — создателя FPGA с триггерной памятью конфигурации (SRAM-based FPGA). Перечень выпущенных фирмой Xilinx FPGA с оперативным репрограммированием насчитывает несколько семейств. К современным микросхемам можно отнести семейства, начиная с XC4000E/XL, в состав которого входит довольно много представителей и архитектура которого послужила базой для последующего семейства Spartan. В дальнейшем технологический прогресс позволил усложнить схемы, увеличить в них число логических блоков и ресурсы трассировки, улучшить технико-экономические показатели. На новую ступень развития архитектуры и схемотехники продукция фирмы

Xilinx перешла при создании семейств Virtex и их модификаций, достигнутый уровень был затем использован при разработке семейства Spartan II.

Архитектура логических элементов у всех микросхем фирмы Xilinx традиционна, они содержат табличные функциональные преобразователи, триггеры и программируемые мультиплексоры. Память в течение довольно длительного времени аккумулировалась в ресурсах LUT-блоков, но в семействе Virtex и последующих появились и встроенные блоки памяти.

При переходе к семействам Virtex и Virtex E были изменены параметры LUT-блоков и характер схем памяти. Распределенная память дополнилась 4-Кбитными двухпортовыми блоками памяти SRAM, для блоков верхнего уровня памяти были применены высокоскоростные буферы ввода/вывода с целью реализации скоростного интерфейса как между этими блоками, так и для обмена между ними и внешними статическими и динамическими ЗУ. Число блоков DLL было доведено до 8, был введен протокол обмена информацией с производительностью более чем 300 Мбит/с на контакт (SelectLink). В семействе Virtex EM два верхних слоя межсоединений выполнены на медных проводниках для распределения по кристаллу мощных сигналов тактирования с большой скоростью.

Семейство Spartan II основано на технических новшествах, разработанных при создании микросхем Virtex, но его микросхемы проще и дешевле. Семейство Spartan II родственно по архитектуре семейству Virtex, но в нем сокращен состав функциональных блоков (нет параллельного интерфейса для одного из режимов конфигурирования, нет средств создания декодеров методами монтажной логики и т. д.), используются только дешевые корпуса. Последняя разработка фирмы в рамках данного направления — семейство Spartan IIE с цепями передачи данных типа LVDS (малосигнальными дифференциальными линиями связей), ориентированное на реализацию быстрых алгоритмов ЦОС. Отмечается, что микросхемы семейства Spartan IIE сочетают такие ценные свойства, как уровень интеграции, приближающийся к возможностям ASIC, малое время выхода продукции на рынок и репрограммируемость при хорошем соотношении цена/вентиль. Сложность микросхем семейства лежит в пределах от 50 до 300 тыс. вентилей.

Последнее на сегодня мегавентильное семейство микросхем фирмы Xilinx — Virtex II. В его архитектуре продолжена линия усложнения логических блоков, увеличены объемы памяти и ресурсы трассировки. Введена специализированная схема быстрого умножителя 18×18. Появились внутрикристалльные резисторы-терминаторы для линий передачи сигналов с возможностью выбора их сопротивлений, которые автоматически устанавливаются соответственно внешним условиям.

На будущее планируется выпуск семейства Virtex IIPro с интегрированным ядром процессора Power PC фирмы IBM.

Если фирма Xilinx имеет приоритет в применении триггерной памяти конфигурации, то лидером в развитии ПЛИС комбинированной архитектуры

можно считать фирму Altera. Первым семейством микросхем с комбинированной архитектурой были микросхемы семейства FLEX8000, затем появились семейства FLEX10K/KE и FLEX6000. По типу логических блоков эти семейства можно отнести к FPGA, но по типу системы межсоединений, напротив, они имеют близость к CPLD, т. к. фирма Altera избегает сегментированных линий связи, стремясь по возможности сохранить предсказуемость задержек в связях.

Структура и схемотехника микросхем семейства FLEX10K/KE рассмотрена ранее в этой главе. Семейства FLEX6000 и FLEX6000A структурно упрощены в сравнении с семейством FLEX10K/KE, но одновременно с этим выполнены на более новом и совершенном технологическом базисе, имеют улучшенные показатели по плотности и быстродействию схем. Эти схемы расцениваются как удачное решение с точки зрения сочетания параметров "цена-производительность" для проектов, не требующих наличия памяти на кристалле и не слишком сложных. В сравнении с семейством FLEX10K/KE в микросхемах FLEX6000 и FLEX6000A исключены блоки PLL, сокращены ресурсы глобальной трассировки и усилены ресурсы местных связей, ограничен выбор типов корпусов.

Семейство ACEX1K — вариант микросхем FLEX10K/KE, реализованный с уменьшенными топологическими нормами, но значительно более дешевый, чем его мегаэвентильные "соседи" того же технологического поколения. Микросхемы семейства ACEX1K имеют сравнительно небольшой уровень интеграции, меньший выбор типов корпусов. Эти микросхемы популярны у отечественных разработчиков вследствие невысокой стоимости при хороших технических параметрах.

Последнее семейство микросхем фирмы Altera — Mercury (Меркурий) оптимизировано по критерию быстродействия и ориентировано на использование в высокопроизводительных системах ЦОС и сложных коммуникационных системах. Имеет высокоскоростную систему межсоединений с использованием приоритетностей и конвейерный умножитель, до 18 каналов поддержки CDR (Clock Data Recovery), и способно поддерживать протоколы Gigabit Ethernet и другие скоростные протоколы передачи данных.

Семейства микросхем APEX20K/KE и APEX II типа "система на кристалле" рассмотрены выше в разд. 1.6.

Фирма Agere Systems (ранее Lucent Technology и еще ранее AT&T Micro-electronic) в соответствии с изначальной проблемной ориентацией фирмы AT&T концентрирует свои усилия на производстве микросхем для систем цифровой связи. Эта фирма на современном этапе выпускает семейства ORCA2, ORCA3, ORCA4 и несколько специализированных ПЛИС на их основе. Указанные семейства модифицируются с течением времени и переводятся на производство по новым технологическим процессам, что сразу же отражается на уровнях питающих и сигнальных напряжений. Логические

элементы серий ORCA основаны на использовании LUT-блоков и триггеров, четыре таких комбинации группируются в блок PFU (Programmable Function Block), память функциональных блоков может быть использована как RAM или ROM. Отличием семейств, начиная с ORCA3, является наличие интерфейса для сопряжения с процессором, допускающее параллельное программирование конфигурации микросхем и подключение процессоров 960 или Power PC. При переходе к семейству ORCA4 акцент был сделан на совершенствовании сигнальной трассировки, задержки сигналов в которой все более определяют общее быстродействие микросхем.

На основе семейств ORCA выпущены схемы FPSC (Field Programmable System Circuits), в том числе OR3TP12, OR3LP26B, ORT4622, ORT8850, ORT82G5, ORLI10G. В этих микросхемах матрицы типа FPGA сочетаются с ядрами на элементах ASIC, причем вначале применялись ядра PCI, а позднее эти ядра стали заменяться на схемы дуплексных синхронных интерфейсов с высокими скоростями передач и встроенными средствами CDR (Clock Duty Recover). В микросхемах на основе серии ORCA4 достигнуты наивысшие скорости передач: в микросхеме 4622 реализованы 4 канала с пропускной способностью 622 Мбит/с, в микросхеме 8850 — 8 каналов и 50 Мбит/с, в микросхеме 82G5 получены скорости 1,25, 2,5 и 3,125 Гбит/с, в микросхеме 10G — 10 Гбит/с.

Фирмой Atmel выпускаются сходные семейства FPGA: AT6000 и AT40K, причем семейство AT40K новее и больше приспособлено к широкому применению в универсальных проектах. Компания делает акцент на возможностях частичного динамического реконfigurирования своих FPGA (возможностях так называемого reconfigurable computing). Подробнее о микросхемах AT40K сказано ранее при описании SOPC типа FPSLIC. К сведениям о микросхемах FPSLIC можно добавить, что в последнем нововведении фирмы (архитектуре AT49S) приняты меры по повышению засекреченности разработанных проектов.

Фирма Cypress Semiconductor использует триггерную память конфигурации в своих микросхемах Delta39K, имеющих сходство с описанными ранее микросхемами Ultra37000. Переход от памяти конфигурации типа EEPROM Ultra37000 к триггерной памяти в Delta39K сопровождался значительным увеличением емкости пользовательской памяти. Вариантом пониженной стоимости является семейство Quantum38K, отличающееся от семейства Delta39K рядом упрощений. На основе микросхем Delta39K разработаны кристаллы PSI (Programmable Serial Interface) с сочетанием областей БМК и программируемой логики, в которых достигнуты скорости передачи сигналов от 200 Мбит/с до 2,5 Гбит/с на контакт и общая пропускная способность тракта в 12 Гбит/с. При этом поддерживаются стандарты PCI, SONET, Gigabit, Ethernet, Fibre Channel и InfiniBand. В отличие от других архитектур с триггерной памятью конфигурации, схемы Delta39K относятся к CPLD.

В производстве встроенных FPGA-ядер для схем на основе ASIC участвует пока что небольшое число фирм. Фирма Actel выпускает микросхемы Varicore с напряжением питания 1,8 В, числом логических ячеек от 1024 до 8192 и встроенной памятью в диапазоне от 36 864 до 73 728 бит. Фирма Adaptive Silicon производит семейство MSA2500 с напряжениями питания 1,8; 1,3 и 1,0 В и числом логических ячеек от 64 до 1024 (схемы памяти отсутствуют). Фирма Agere Systems поставляет микросхемы eORCA (embedded ORCA) с напряжением питания 1,5 В, числом логических ячеек 800 и памятью в составе LUT-блоков емкостью 102 400 бит. Фирма Atmel разработала микросхемы под названием Embedded FPGA с напряжением питания от 1,8 до 3,3 В, числом логических ячеек от 256 до 6400 и встроенной памятью емкостью от 2048 до 51 200 бит.

Сведения о блочных "системах на кристалле", содержащих процессорные ядра

Принятая выше форма представления параметров более или менее удовлетворительна применительно к микросхемам, не имеющим разнообразного состава аппаратных ядер. Для блочных систем указание числа эквивалентных вентилях, объема памяти и т. п. недостаточно характеризует микросхему, т. к. первоочередной интерес для таких систем представляет состав реализованных ядер, их типы, количество и характеристики. Для блочных систем с процессорными ядрами ниже дается краткое словесное описание основных представителей современной продукции ведущих фирм. Более подробные сведения о некоторых блочных SOPC, были даны ранее в этой главе.

В современной номенклатуре 32-разрядных *hard-ядер процессоров* имеется несколько вариантов, которые могут быть эффективно встроены в кристаллы со схемами программируемой логики вследствие своей настроенности на процессы изготовления FPGA. Такие *hard-ядра* компактны и могут занимать всего лишь 1,5–2 мм² (без кэшей). По архитектуре это, в сущности, стандартные RISC-процессоры. Производителями FPGA применяются *hard-ядра* с процессорами ARM9T, MIPS4KC, PowerPC405C. В маломощных портативных системах доминируют процессоры ARM, следом идут архитектуры MIPS и Power PC. В этих процессорах достигаются рабочие частоты до 200 МГц и более, в них применяются 5–6 ступенчатые конвейеры, выполняются скалярные операции, достигнута малая мощность потребления.

Hard-ядро ARM922T фирмы ARM имеет площадь 11,8 мм² (при топологических нормах 0,18 мкм), рабочие частоты до 200 МГц и используется фирмами Altera и QuickLogic. С этим ядром связано применение шинной системы AMBA с шинами AHB, ASB и APB, в обозначениях которых средние буквы означают соответственно High-Speed, System и Peripheral. Имеются кэши данных и команд емкостью по 16 Кбайт, 6-ступенчатый конвейер, аппаратный умножитель, рассеиваемая мощность 160 мВт.

Ядро ARM740T, используемое фирмой Triscend, имеет площадь 2,5 мм², рабочую частоту до 60 МГц, смешанный кэш команд/данных емкостью 8 Кбайт, 4 канала прямого доступа к памяти и контроллеры/интерфейсы для работы с внешней памятью.

Ядро MIPS (Microprocessor without Interlocked Pipeline Stages, что, в сущности, означает "микропроцессор без задержек ожидания конвейера") фирмы MIPS Technologies сейчас лицензируется многими крупными производителями электронной аппаратуры. Ядро MIPS4C занимает площадь 3 мм². Используется фирмой QuickLogic, а вариант MIPS4KC также и фирмой Altera. Имеет кэши команд и данных по 16 Кбайт, 5-ступенчатый конвейер, аппаратные средства для выполнения операций умножения, деления, умножения с накоплением. Рабочая частота ядра 200 МГц, рассеиваемая мощность 400 мВт. Используется с шинной системой AMBA.

Ядро Power PC405C фирмы IBM Microelectronics имеет площадь 3 мм², используется фирмой Xilinx с шинной системой CoreConnect фирмы IBM. Кэши команд и данных имеют объемы по 32 Кбайта, применяется 5-ступенчатый конвейер, аппаратные схемы умножения и деления, мощность рассеяния составляет 400 мВт.

Soft-ядра Nios фирмы Altera и Microblaze фирмы Xilinx кратко рассмотрены выше (см. разд. 1.5 и 1.6 данной главы).

Информация об однородных системах на кристалле приведена на рисунках в разделах ПЛИС с триггерной памятью конфигурации и в тексте главы (семейства Virtex, APEX и др.). Блочные "системы на кристалле" перечислены ниже с указанием фирм-производителей.

□ Фирма Triscend выпускает CSOC (Configurable SOC) семейств E5 и A7, в которых объединяются процессор (8-разрядное ядро 80C51 для семейства E5 с производительностью 10 MIPS на 40 МГц или 32-разрядное ядро ARMA7 для семейства A7 с производительностью 54 MIPS на 60 МГц), скоростные шины на кристалле и FPGA с триггерной памятью конфигурации. Работа процессора поддерживается наличием кэша емкостью 8 и 16 Кбайт статической памяти. На кристалле имеется 4-канальный контроллер прямого доступа к памяти и схемы интерфейса для работы с внешней памятью и стандартной периферией, а также обычная для контроллеров периферия (2 таймера, 2 UART, сторожевой таймер). FPGA дает до 40К вентилях (до 3200 логических элементов, до 51 200 бит памяти в LUT-блоках и до 521 288 бит встроенной памяти). Имеется блок внутрисхемной отладки. Стоимость микросхем лежит в диапазоне 5–19 долларов для семейства E5 и составляет приблизительно 20 долларов для пока единственного представителя семейства A7 (микросхемы A7520). Несмотря на достаточно универсальный характер блоков CSOC, фирма отмечает ее проблемную ориентацию — для систем коммуникации. Triscend ведет совместную работу с фирмой Hitachi с целью выпуска

- CSOC на основе ее процессора SuperH. Предполагается и дальнейшее расширение перечня процессорных ядер в CSOC фирмы Triscend.
- Фирма Atmel выпускает микросхемы FPSLIC с 8-разрядным процессорным ядром AVR, имеющим производительность 30 MIPS на 40 МГц. Процессорное ядро объединено с FPGA емкостью до 40K вентилей (до 6400 логических элементов) и собственной памятью RAM емкостью 18 Кбит. На кристалле размещены также память команд и память данных с общей емкостью 32 Кбайт, блок регистров размерностью 32×8, аппаратный умножитель размерностью 8×8 и стандартная для микроконтроллеров периферия (2 UART, 3 таймера, двухпроводной стандартный последовательный интерфейс, сторожевой таймер). Особенностью FPSLIC являются возможности динамической частичной реконфигурации системы, обеспечивающие реализуемость решения задач методом, называемым reconfigurable computing. Стоимость микросхем находится в пределах 7—58 долларов.
 - Фирма Altera производит SOPC семейство Excalibur, в состав которого обычно включают как SOPC с soft-ядром процессора Nios, размещаемым на кристалле семейств APEX, так и SOPC с hard-ядрами типов ARM или MIPS. Имея в качестве основы ПЛИС APEX высокой сложности, SOPC Excalibur занимают позиции в верхней части спектра SOPC. Типичное число эквивалентных вентилей доходит у них до 1 млн., число логических элементов до 38 400, 32-разрядное процессорное ядро ARM922T работает на частотах до 166 МГц, используется высокоскоростной вариант шинной системы AMBA (шина АНВ). Имеются отдельные кэши по 8 Кбайт для команд и данных, несколько видов оперативной памяти, модуль для отладки программного обеспечения, гибкий контроллер прерываний, UART, таймеры (общего назначения и сторожевой). Стоимость микросхем находится в пределах 265—925 долларов.
 - Фирма Cypress Semiconductor выпускает кристаллы PSOC, архитектурно представляющие собою микроконтроллер, дополненный программируемыми цифровыми и аналоговыми блоками, ориентированными на создание специального набора периферийных функций. Для цифровой части это таймеры, последовательные порты, генераторы, для аналоговой — цифроаналоговые и аналого-цифровые преобразователи и фильтры (низких частот, высоких частот, полосовые). Имеется умножитель 8×8 с 32-разрядным аккумулятором, до 16 Кбайт Flash-памяти и 256 Кбайт статического ОЗУ. Микросхема работает на частотах до 24 МГц.
 - Фирма Semiconductor Design Solution выпускает "системы на кристалле" FIPSOC с процессорным ядром 8051 и цифровыми и аналоговыми подсистемами программируемых периферийных устройств. Цифровая часть содержит функциональные блоки типа LUT + RG, аналоговая часть может программироваться на такие заказные функции, как усиление диф-

ференциальных сигналов, сравнение и преобразование данных, т. е. ориентирована на так называемое front-end design.

- Фирма Motorola намеревалась войти в сферу производства блочных SOPC с микросхемой MPACF250, которая расценивалась многими специалистами как весьма перспективная, поскольку планировавшееся для нее процессорное ядро имело преимущества перед существовавшими. Однако по каким-то причинам имевшиеся планы пока не реализуются.
- Фирма LSI Logic совместно с фирмой Adaptive Silicon разработала SOPC, в которой объединены традиционные схемы, реализованные по методу стандартных ячеек, ядро процессора ARM и программируемая логика для реконфигураций, адаптирующих проект к новым требованиям.
- Фирма Agere Systems объявила о выпуске SOC Ultra Mapper, содержащей до 8 млн. вентилей, для реализации функций, характерных для систем передачи речи и другой информации в проводных системах.