

УДК 681.326.7

Г.П. АКСЁНОВА

*Институт проблем управления им. В.А. Трапезникова РАН, Москва, Россия***ОБЕСПЕЧЕНИЕ КОНТРОЛЕПРИГОДНОСТИ В МАТРИЦЕ ТИПА *FPGA***

Цель данной работы – выработать рекомендации по преобразованию матрицы *FPGA* так, чтобы повысить ее контролепригодность, что, в свою очередь, облегчит организацию тестирования устройства или системы, «зашиваемых» в неё, так как всё необходимое для тестирования будет выполняться автоматически.

контролепригодность, самотестирование, матрица *FPGA***Введение**

В данной работе речь идет не о проверке матрицы *FPGA* (*field-programmable gate array*) при ее изготовлении, а о проверке «защитой» в нее схемы, что используется на стадии эксплуатации кристалла. Предлагается, чтобы услуги для такого вида тестирования предоставлял изготовитель кристалла. Он оборудует кристалл и процесс размещения в нем схемы так, чтобы любая «зашиваемая» схема была бы самотестируемой. В результате таких шагов снизится время проектирования устройств и выхода их на рынок, так как проектировщику не надо будет думать о том, как проверять то, что создаешь. Это особенно важно при разработке систем на кристалле (*SoC*). Понятно, что повышенная контролепригодность матрицы облегчит и проверку ее самой в процессе производства.

Чтобы сразу ответить на возражения, что изготовитель этого делать не будет, напомним, что когда-то и слышать не хотели о том, чтобы в проектируемое устройство надо добавить дополнительно аппаратуры в целях ее контроля. А сейчас это стало правилом и даже утверждены стандарты, например *JTAG*.

Более того, раньше не думали и о широком использовании матриц *FPGA*, так как реализация в них сопровождается большой избыточностью. В настоящее время, благодаря успехам нанотехнологий, можно пойти навстречу требованиям снижения стоимости и времени проектирования аппаратуры тестирования для разрабатываемых устройств и систем.

Для того чтобы выпускать конкурентноспособную продукцию, надо смотреть вперед и создавать новые качества в ней. В предлагаемой работе дается один из подходов по созданию контролепригодных матриц *FPGA*. Это первая заявка по данному вопросу. Главное – начать об этом говорить, спорить, развивать идеи.

Структура матрицы *FPGA*

Рассматривается программируемая матрица вентилей типа *FPGA*. В отечественной литературе пользуются аббревиатурой ПЛИС – программируемая логическая интегральная схема. Однако здесь будем употреблять термин *FPGA*, подразумевая под этим определенное структурное устройство матрицы.

Структура матрицы представлена на рис. 1. Из рисунка видно, что матрица содержит большое число одинаковых конфигурируемых (программируемых)

Работа поддержана Российским фондом фундаментальных исследований (грант 05-08-01388) и Программой №16 ОЭММПУ РАН.

мых) логических блоков (КЛБ) 1. Они распределены по всему кристаллу среди программируемых соединений – коммутационного поля 2, а вся матрица окружена программируемыми блоками ввода-вывода 3. КЛБ в микросхемах различных семейств отличаются друг от друга, однако все они содержат комбинационную часть, реализованную на ПЗУ (*look-up table - LUT*), триггер и мультиплексоры.

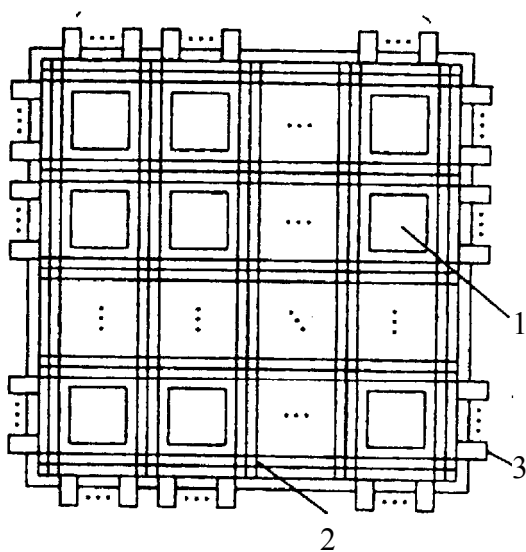


Рис. 1. Структура матрицы *FPGA*

Таким образом, матрица *FPGA* – это однородная структура. В матрице легко организовать различные связи. В ней не так критично отношение к дополнительным затратам площади кристалла при введении схем контроля.

Организация самотестирования в матрице *FPGA*

Будем называть «зашиваемую» в матрицу схему объектом. Проверку объекта при помощи средств тестированием, тоже «зашитых» в матрицу, будем называть самотестированием. Т.е. самотестирование осуществляется практически без использования внешнего тестового оборудования; с внешней стороны необходимо только инициировать процесс самотестирования, а по окончании считать резуль-

тат. Будем придерживаться следующих принципов самотестирования объекта, изложенных в [1].

- 1) Объект проверяется в декомпозированном виде, т.е. разбивается на части, удобные для тестирования, и каждая часть проверяется самостоятельно.
- 2) Для декомпозиции нужны средства разрыва.
- 3) Части проверяются на исчерпывающем тесте (тесте полного перебора).
- 4) Тест вырабатывает встроенный генератор тестов (ГТ), выходную реакцию собирает сигнатурный анализатор (СА).

5) Результирующая сигнатура сравнивается с эталонной, хранящейся внутри кристалла.

Посмотрим, как эти принципы тестирования можно осуществить в *FPGA*.

В качестве частей возьмем КЛБ. Таким образом, при проектировании объект собирался из КЛБ, а при тестировании его «рассыпаем» опять на эти части. Здесь напомним, что хотя в матрице все КЛБ одинаковые, в «зашитом» объекте каждый КЛБ запрограммирован на свою функцию. КЛБ имеет небольшое число информационных входов n (у фирм Xilinx и Altera $n < 10$). Соответственно столько же разрядов должен иметь и ГТ. Следовательно, время прогонки теста для КЛБ $t = 2^n$ будет меньше 1000 тактов, что на частоте 1 ГГц составляет 1 микросекунду.

Благодаря малому объему аппаратуры ГТ для экономии линий связи в матрице можно иметь несколько ГТ и территориально расположить их в разных местах матрицы. В этом случае каждый ГТ обслуживает свои ближайшие КЛБ.

Чтобы «рассыпать» объект для тестирования, требуются средства разрыва, которые отсоединяют КЛБ друг от друга и присоединяют их к ГТ и СА. В качестве средств разрыва можно взять оконечные (выходные) мультиплексоры (Мх) в КЛБ и несколько их модернизировать, а именно: добавить вход (назовем его диагностическим), куда будут подаваться сигналы от ГТ, и добавить управляющий

вход, который будет управлять переключением Мх из штатного режима в диагностический и наоборот. Здесь штатный режим – это режим нормального функционирования, а диагностический – когда штатная связь разрывается и на выход Мх проходят сигналы с диагностического входа.

При тестировании проверяемые части матрицы будут иметь вид, показанный (в качестве примера) на рис. 2, где связи с предшествующими КЛБ включены в состав проверяемого КЛБ.

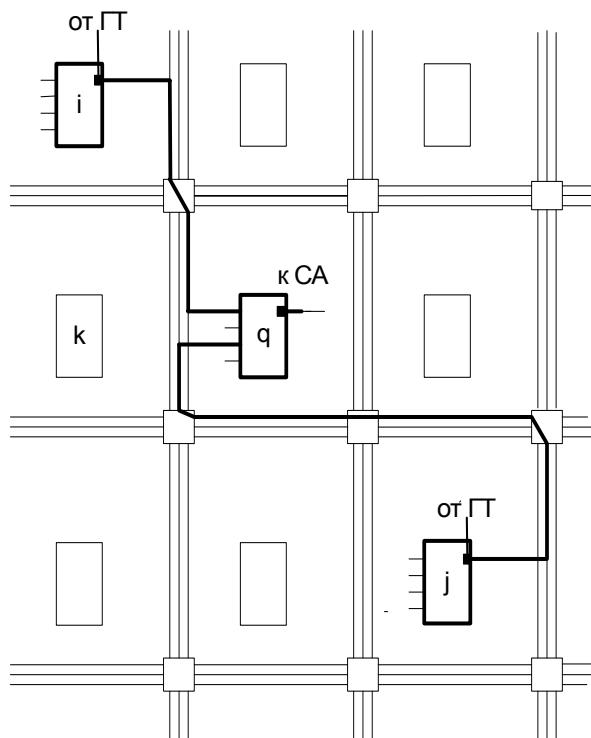


Рис. 2. Вид проверяемой части

На входах и выходе проверяемой части стоят средства разрыва, причем входные Мх находятся в диагностическом режиме, подавая сигналы с ГТ, а выходные Мх – в штатном, посылая выходную реакцию на СА. Так как входной Мх q -го КЛБ является одновременно выходным Мх i -го КЛБ, то понятно, что при одновременной проверке этих двух КЛБ Мх должен работать сразу в двух взаимоисключающих режимах. Отсюда вывод: КЛБ не может проверяться одновременно со своими непосредственными предшественниками и последователями.

Следовательно, все КЛБ, принадлежащие объекту, надо разбить на группы так, чтобы в одну группу попали подсистемы, которые можно проверять одновременно, а сами группы проверять последовательно. Очевидно, можно найти много вариантов такого разбиения, но если искать минимальное решение, то эта задача сводится к известной задаче о «раскраске» вершин графа.

Как организовать проверку группы? Для этого надо входы проверяемой части развести по выходам ГТ так, чтобы никакая пара входов не была подсоединена к одному и тому же разряду генератора (иначе на входах не сложится исчерпывающий тест). В [1] разработан алгоритм такой разводки. КЛБ разводится на ближайший к нему ГТ. При проверке все ГТ запускаются одновременно. Такая параллельная работа генераторов эквивалентна работе одного ГТ. Выходная реакция со всех проверяемых КЛБ записывается в СА. Ранее в литературе было доказано, что при таком «замешивании» в СА реакций со всех КЛБ неправильная реакция от неисправного КЛБ с очень большой вероятностью оставляет «свой след» в СА.

Из сказанного ясно, что в матрице должны быть реализованы связи диагностических входов Мх КЛБ с ГТ и выходов Мх с СА. Это можно сделать двумя способами.

1) Связи могут набираться на коммутационном поле матрицы. При этом тратится коммутационный ресурс матрицы.

2) Чтобы не занимать коммутационный ресурс, изготовитель может сделать эти связи как постоянные.

Разводка связей на ГТ и СА осуществляется при помощи подпрограммы, включенной в программу конфигурации. Подпрограмма делает следующее:

1) по результатам размещения объекта в кристалле (когда стало известно, кто чей сосед) осуществляет разбиение всех КЛБ на группы, которые могут проверяться одновременно;

2) реализует все необходимые соединения с ГТ, СА и узлом управления самотестированием. Таким образом, в описанном подходе

– благодаря декомпозиции преодолена проблема размерности проверяемой схемы;

– благодаря использованию исчерпывающих тестов снята проблема построения детерминированного теста и достигнуто 100%-е обнаружение любых логических неисправностей в КЛБ;

– благодаря включению связей между КЛБ в состав проверяемых частей осуществляется проверка всех соединений коммутационного поля, используемых в «зашитом» объекте;

– благодаря декомпозиции и параллельному тестированию КЛБ в группе время выполнения теста равно: $t = K 2^n$, где $n \leq 10$ – число входов КЛБ, а K – число групп;

– благодаря тому, что все триггеры в КЛБ управляются и наблюдаются, сняты «автоматные трудности» тестирования. Однако чтобы находящийся в составе КЛБ триггер гарантированно получил свой тест (т.е. переключился из одного своего состояния в другое и потом назад), необходимо исчерпывающий тест подать последовательно два раза. Это увеличивает время тестирования вдвое: $t = 2 K 2^n$;

– благодаря инвертированию исчерпывающего теста во время второй его прогонки внесенные средства разрыва получают свой проверяющий тест.

Заключение

Впервые предлагается для решения задачи тестирования разрабатываемых устройств и систем

обеспечить контролепригодность самих матриц *FPGA*, в которых реализуются устройства. Это значительно сократит время проектирования новых устройств, так как проектировщику не надо будет думать о том, как проверять создаваемый объект.

Отличие изложенного подхода от различных существующих подходов по тестированию интегральных схем заключается в том, что в последних средства тестирования объекта проектируются и реализуются в матрице точно так же, как и сам объект, не затрагивая структуры матрицы. Данный подход частично вторгается в структуру матрицы.

В [2] проведено сравнение по затратам предложенного подхода с распространенным методом сканирования *LSSD*.

Литература

1. Аксенова Г.П., Халчев В.Ф. Метод параллельно-последовательного самотестирования СБИС на основе их декомпозиции // *АиТ*. – 1991. – № 4. – С. 147-156.

2. Аксенова Г.П., Халчев В.Ф. Метод параллельно-последовательного самотестирования в интегральных схемах типа *FPGA* // *АиТ*. – 2007. – № 1. – С. 163-170.

Поступила в редакцию 3.02.2007

Рецензент: д-р техн. наук, проф. В.С. Харченко, Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ», Харьков.