

менных АИС, с одной стороны, и необходимость соблюдения действующих стандартов ГОСТ 34, с другой, в данной работе предпринята попытка определения базы для совместного использования моделей ЖЦ по ГОСТ 34 и MSF, или, иными словами, возможность использования модели MSF в контексте стандартов ГОСТ 34.

Проведенный анализ отражает важность правильного выбора модели ЖЦ, необходимость использования методов документационного управления при создании АИС и является приглашением к разговору о результатах *практического* применения модели проектирования MSF. По мнению авторов работы группа стандартов ГОСТ 34 не исчерпала свой потенциал и продолжает оставаться полезным нормативно-методическим материалом для создателей и заказчиков АИС. Заложенная разработчиками этих стандартов адаптивность позволяет "вписать" модель процесса проектирования MSF в рамки, определяемые стандартами ГОСТ 34, путем надлежащего определения этих рамок. Основная роль здесь отводится договорной документации и техническому заданию, то есть основополагающим документам на проектирование АИС. Для повышения эффективности подобной адаптации целесообразна разработка стандартов предприятий, отраслевых и межотраслевых документов, облегчающих выполнение адаптации для каждого конкретного проекта.

Успешность применения модели ЖЦ MSF должна быть поддержана использованием всех сопутствующих моделей, в том числе модели проектной группы, модели управления риском, модели модульного приложения, модели разработки решений, модели архитектуры масштаба предприятия, модели общей стоимости владения, а также надлежащим формированием сквозных процессов проектирования АИС, что является неотъемлемой частью производственной архитектуры MSF. Эти вопросы также должны найти надлежащее отражение в соответствующих нормативно-методических документах на проектирование АИС.

ЛИТЕРАТУРА

- [1]. Зиндер Е.З. Новое системное проектирование: информационные технологии и бизнес-реинжиниринг // Системы управления базами данных. 1995. № 4. - (www.osp.ru).
- [2]. Зиндер Е.З. Соотнесение и использование стандартов организации жизненных циклов систем. // Системы управления базами данных. 1997. № 3. С. 41-53.
- [3]. Зиндер Е.З. Революционные изменения базовых стандартов в области системного проектирования // Директору информационной службы. 2001. № 5. - (www.osp.ru).
- [4]. Липаев В.В. Документирование и управление конфигурацией программных средств. Методы и стандарты. - М.: СИНТЕГ, 1998. - 220 с.
- [5]. Липаев В.В. Стандарты, регламентирующие жизненный цикл сложных комплексов программ информационных систем // Информационные технологии. 1998. № 9. С. 2-8.
- [6]. Вендров А.М. Проектирование программного обеспечения экономических информационных систем: Учебник. - М.: Финансы и статистика, 2000. - 352 с.
- [7]. Ойхман Е.Г. Залог успеха - в быстрой разработке приложений / Интервью Э.Пройдакова, А.Ливеровского. 1998. - (<http://kis.pcweek.ru/kis/win/meto/argussoft.html>).
- [8]. Microsoft Corporation. Принципы проектирования и разработки программного обеспечения. Учебный курс MCSD / Пер. с англ. - М.: Издательско-торговый дом "Русская Редакция", 2000. - 608 с.
- [9]. Брандт Д. Architectures. Экзамен - экстерном (экзамен 70 - 100). - СПб.: Питер, 2001. - 432 с.
- [10]. Брукс Ф. Мифический человек-месяц или как создаются программные комплексы / Пер. с англ. - СПб.: Символ-Плюс, 1999. - 304 с.
- [11]. Гринберг А.С., Горбачев Н.Н. Документационные системы управления и документационные оболочки. // Новые информационные технологии = New Information Technologies: Тр. междунар. конф. Минск, 5-7 декабря 2000 г.: В 3-х кн. Кн. 2. - Мн.: БГЭУ, 2000.
- [12]. Спицнадель В.Н. Основы системного анализа: Учеб. пособие. - СПб.: Изд. дом "Бизнес-пресса", 2000. - 326 с. использование JTAG-технологий для встраиваемых систем

ИСПОЛЬЗОВАНИЕ JTAG-ТЕХНОЛОГИЙ ДЛЯ ВСТРАИВАЕМЫХ СИСТЕМ

А.А. Иванюк, Ю.В. Климец, Ю.В. Быков, И.А. Мурашко, В.Н. Ярмолик

Белорусский государственный университет информатики и радиоэлектроники, кафедра программного обеспечения информационных технологий, ул. П.Бровки, 6, Минск, 220600, БЕЛАРУСЬ, тел. +375 (17) 239-80-20, ivaniuk@bsuir.unibel.by

АННОТАЦИЯ

В данной статье рассматриваются современные аспекты применения технологии граничного сканирования JTAG для доступа к внутренним компонентам встраиваемых систем. Приводится описание программирования Flash-памяти с использованием JTAG.

1. ВВЕДЕНИЕ

В настоящее время встраиваемые системы (Embedded Systems) находят широкое применение в различных областях техники: системы телекоммуникаций, навигационные системы, системы цифровой обработки сигналов (DSP), системы автоматизированного управления, аппаратные комплексы для быстрого и эффективного проектирования электронных систем (Evaluation and Fast Prototyping Boards and Systems) и т.д.

Встраиваемые системы (ВС) характеризуются интенсивным использованием современных СБИС (микропроцессоры, микроконтроллеры, ПЛИС типа FPGA и CPLD, блоки памяти различных типов и организаций) и технологий производства печатных плат для их изготовления. Вследствие этого методы их отладки и тестирования, которые включают в себя проверку правильности функционирования отдельных компонентов системы, тестирование межсоединений, программирование устройств памяти, содержащих инициализирующий код всей системы, становятся все более сложными и дорогостоящими [1,2].

Одним из эффективных подходов для решения задач отладки и диагностики ВС является использование технологии Boundary Scan, которую также часто называют JTAG-технологией или стандартом IEEE 1149.1 [2]. В 1985 году была сформирована специальная группа, состоящая из ведущих производителей электроники - JTAG (Joint Test Action Group). Основной целью группы являлась разработка промышленного стандарта на метод тестирования печатных плат. Первая версия стандарта вышла в 1990 году и называлась IEEE 1149.1-1990 "Test Access Port and Boundary-Scan Architecture". В 1993 году вышла вторая версия стандарта, получившая номер 1149.1a, и содержащая наряду с исправлением ошибок предыдущей версии, описание двух новых команд. Кроме этого, в 1994 году вышло дополнение к стандарту, содержащее описание специального языка Boundary Scan Description Language (BSDL), позволяющего производителям СБИС

с поддержкой граничного сканирования описывать процедуру тестирования. В настоящее время повсеместно в качестве синонима стандарта используется аббревиатура группы, которая его разрабатывала (JTAG). Поэтому далее в статье мы также будем при ссылке на стандарт IEEE 1149 использовать аббревиатуру JTAG. В первые годы после своего появления JTAG-технология в основном использовалась для тестирования и диагностики, как и предполагалось ее разработчиками. Однако, в настоящее время ее мощные средства, которые, прежде всего, обеспечивают высокую степень управляемости и наблюдаемости в электронной системе, поддерживающей стандарт IEEE 1149.1, стали использоваться и для других целей.

В настоящей работе рассматриваются методы эффективного использования JTAG-технологии, применяемые при разработке и использовании ВС.

2. JTAG-ТЕХНОЛОГИИ И ОСНОВНЫЕ ОБЛАСТИ ПРИМЕНЕНИЯ

Необходимым условием для применения JTAG-технологий является использование в системе электронных компонентов, поддерживающих стандарт IEEE 1149.1 (Рис. 1).

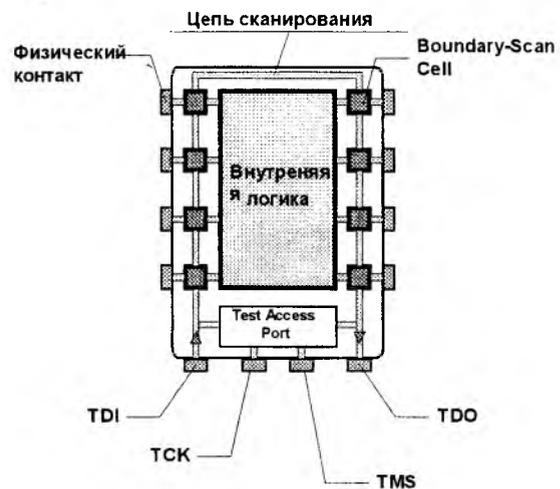


Рисунок 1. JTAG-компонент

Каждый контакт JTAG-компонента (СБИС) может быть изолирован от внутренней логики посредством Boundary Scan Cell – ячейки для подачи в цепь сканирования (Boundary Scan Chain) логических сигналов, которые могут быть использованы для различных целей (тестирование, программирование). Для управления цепью сканирования и выполнения опера-

ций, поддерживаемых стандартом IEEE 1149.1, используются дополнительные контакты TDI (Test Data In), TDO (Test Data Out), TMS (Test Mode Select) и TCK (Test Clock).

Все ячейки граничного сканирования объединяются в один длинный сдвиговый регистр с параллельной загрузкой/выгрузкой. Операция параллельной загрузки (CAPTURE) параллельно загружает все логические значения, поступающие на вход СБИС, во внутреннюю память каждой ячейки граничного сканирования. Кроме того, выходные сигналы, поступающие из внутренней логики СБИС, также записываются в память соответствующих ячеек. Операция параллельной выгрузки (UPDATE), напротив, параллельно выгружает все значения из памяти ячеек граничного сканирования на выходы СБИС или на входы внутренней логики. Кроме того, в режиме последовательного сдвига, вся цепь сканирования функционирует как последовательный регистр, входные данные на который поступают со входа TDI, а выходные данные выдвигаются через выход TDO. Объединяя выход TDO одной СБИС с входом TDI другой, можно создать одну длинную цепь сканирования, охватывающую все интегральные схемы на плате. Таким образом, для организации граничного сканирования требуется всего 4 дополнительных контакта на каждой ИС и 4 контакта на печатной плате, объединенных в TAP-порт. Назначение контактов JTAG-порта представлено в Табл. 1.

Таблица 1. Описание контактов JTAG-порта

Контакт	Имя	Описание
TDI	Test Data Input	Последовательный вход цепи сканирования.
TDO	Test Data Output	Последовательный выход цепи сканирования.
TMS	Test Mode Select	Вход управления TAP-контроллера.
TCK	Test Clock	Вход синхронизации
TRST	Test Reset	Асинхронный сброс TAP-контроллера. (необязательный контакт)

Структурная схема средств граничного сканирования состоит из следующих элементов:

- четырех обязательных и одного необязательного контакта, объединенных в порт TAP (Test Access Port);
- набора ячеек граничного сканирования на каждом входе и выходе СБИС, последовательно объединенных в сдвиговый регистр;

- контроллера TAP, представляющего собой конечный автомат, управляемый сигналами TCK и TMS;
- n-разрядного ($n > 2$) регистра команд (IR - Instruction Register), используемого для хранения текущей команды контроллера;
- 1-разрядного регистра обхода данных (Bypass);
- необязательного 32-разрядного регистра идентификации (Identification Register), хранящего идентификационный код ИС.

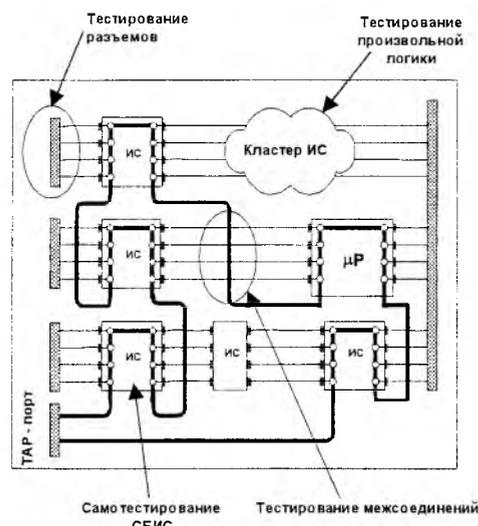


Рисунок 2. Использование JTAG-технологии для тестирования печатных плат и электронных компонентов

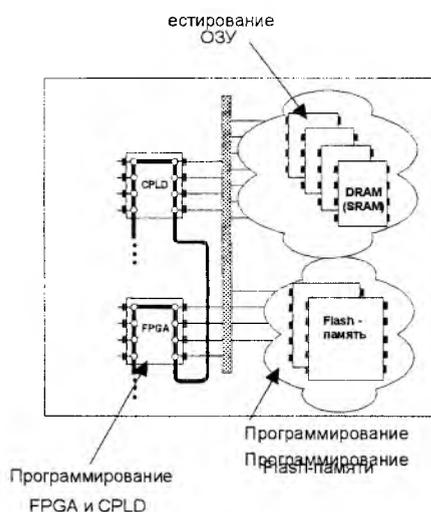


Рисунок 3. Использование JTAG-технологии для конфигурирования ПЛИС и программирования FLASH-памяти

Стандарт IEEE 1149 описывает ряд команд, которые могут выполняться средствами граничного сканирования (Табл. 2).

Сигналы TAP-порта TMS и TCK (а также необязательный сигнал асинхронного сброса TRST) управляют работой конечного автомата TAP-контроллера, который в свою очередь, генерирует управляющие сигналы для внутренней логики средств граничного сканирования.

Посредством JTAG может быть осуществлен доступ к внутренним полюсам платы, что позволяет решать следующие задачи:

- тестирование СБИС;
- тестирование межсоединений;
- тестирование произвольной логики;
- тестирование ОЗУ;
- программирование CPLD и FPGA;
- отладка и эмуляция аппаратных средств ВС.

Таблица 2. Обязательные команды JTAG

Команда	Описание	Код команды
Bypass	Используется для соединения входа TDI с выходом TDO через регистр обхода Bypass	"11...1" - все единицы (количество "1" зависит от конкретной реализации TAP)
Sample/Preload	Позволяет загрузить новые и одновременно считать старые данные из цепи сканирования	Код для этой команды стандартом не описывается и может произвольно назначаться производителем
Extest	Переводит цепь сканирования в режим тестирования внешних соединений	"00...0" - все нули (количество "0" зависит от конкретной реализации TAP)

В настоящей работе основное внимание уделяется использованию JTAG-технологий для программирования Flash-памяти. Данная задача является одной из основных при проектировании и использовании ВС.

3. ПРОГРАММИРОВАНИЕ FLASH-ПАМЯТИ С ИСПОЛЬЗОВАНИЕМ JTAG-ТЕХНОЛОГИЙ

Под программированием Flash-памяти обычно понимают действия, связанные с чтением/записью данных, стиранием сектора, получением идентификационного номера устройства, верификацией данных и т.п [3].

Как правило, при производстве встраиваемых систем, входящая в их состав Flash-память не содержит в себе какой-либо информации, а процедура записи инициализирующего кода в память целиком ложится на разработчиков и пользователей. Для разработчиков эта процедура необходима при отладке и диагностировании

системы. Пользователю же предоставляется возможность записывать свой собственный инициализирующий код. ВС, как правило, не содержат дополнительные физические контакты, посредством которых возможно было бы прямое подключение устройств Flash-памяти к внешним программаторам. Однако, существует возможность косвенного доступа к выводам Flash-памяти при наличии IEEE-1149.1 - совместимых электронных компонентов, входящих в состав ВС. Такой доступ может быть организован при помощи внешних физических контактов TAP-порта и физических выводов интегральных схем (ИС), соединенных между собой общей цепью сканирования (Рис. 4).

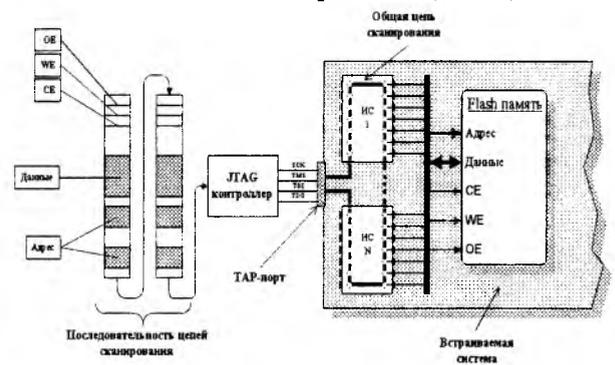


Рисунок 4. Доступ к Flash-памяти через TAP-порт

Формируя в определенной последовательности значения цепей сканирования и подавая их на TDI вход TAP-порта, можно добиться появления конкретных значений на физических выводах микросхемы Flash-памяти, соответствующих, например, операции записи одного слова. Аналогичным образом можно организовать чтение данных из Flash-памяти, подавая определенные последовательности на TDI-вход и захватывая требуемые данные на TDO-выходе.

Для осуществления доступа к Flash-памяти через TAP-порт необходимо наличие информации о следующих конфигурациях:

- 1) конфигурация общей цепи сканирования (количество ИС и типы встроенных в них TAP-контроллеров);
- 2) конфигурация соединений выводов программируемой микросхемы Flash-памяти с выводами ИС, образующих общую цепь сканирования;
- 3) конфигурация программируемой микросхемы Flash-памяти (разрядность шин адреса и данных, количество и назначения управляющих сигналов, алгоритмы записи/чтения

слова, стирания сектора, количество секторов и их адресные диапазоны).

Первую конфигурацию можно составить на основе так называемого BSDL-описания (распространяется фирмами-производителями ИС) электронных компонент ВС. BSDL (Boundary Scan Description Language) представляет собой подмножество языка описания аппаратуры VHDL и позволяет описывать структурные элементы механизма граничного сканирования на уровне разрядностей регистров управления TAP-контроллеров и форматов регистров сканирования. На основе BSDL-описаний устройств, входящих в состав ВС, решается задача нахождения длины общей цепи сканирования и форматов управляющих команд для всех TAP-контроллеров.

Для получения информации о конфигурации 2 зачастую пользуются картой межсоединений выводов всех электронных компонент, входящих в состав ВС, так называемый netlist. Карта межсоединений формируется на стадии разработки при трассировке и разводке всех соединений схемотехнического описания системы, осуществляемой пакетами автоматизированного проектирования. Такое решение возможно лишь в случае, когда выводы микросхемы Flash-памяти имеют непосредственные соединения с выводами ИС общей цепи сканирования (Рис. 4). В иных вариантах, при которых непосредственное соединение невозможно (Рис. 5), требуется наличие описания функционирования транзитных компонент, составляющих кластер ИС, не входящих в общую цепь сканирования. При этом разработчикам и пользователям ВС необходимо вручную указывать соответствие сигналов на выводах ИС сигналам на выводах Flash-памяти.

На основе информации о конфигурации микросхем Flash-памяти определяется последовательность сигналов на шине адреса и шине данных совместно со значениями управляющих сигналов, которые необходимо формировать для осуществления успешного программирования через JTAG порт. Описание алгоритмов задается разработчиками и пользователями ВС. Существует несколько подходов для задания алгоритмов программирования. Рассмотрим один из основных подходов, который заключается в использовании специализированных языков описания механизмов граничного сканирования. В настоящее время среди таких языков нашли широкое применение язык STAPL (Standard Test and Programming Language) и JAM, разработанный фирмой Altera

[4,5]. В семантику обоих языков, по сравнению с другим, входят средства автоматизации, такие как: вызовы процедур, циклы, переменные, механизмы загрузки и чтения регистров граничного сканирования, возможности переводить TAP-контроллер в определенное состояние, а так же анализ получаемых данных. Основным достоинством такого подхода является простота описания значений цепей сканирования, необходимых для осуществления программирования. Практически все современные фирмы-разработчики в области тестирования и отладки аппаратуры поставляют JTAG контроллеры, поддерживающие управляющие языки STAPL и JAM.

Одну из важных ролей при использовании JTAG-технологий для встроенных систем играют JTAG-контроллеры, от которых зависит эффективность доступа к внутренним компонентам ВС. JTAG-контроллеры представляют собой системы управления, отвечающие за передачу данных для регистров сканирования и переключение в различные режимы подключенных TAP-контроллеров. JTAG-контроллеры могут быть выполнены в виде программно-аппаратных или чисто аппаратных комплексов.

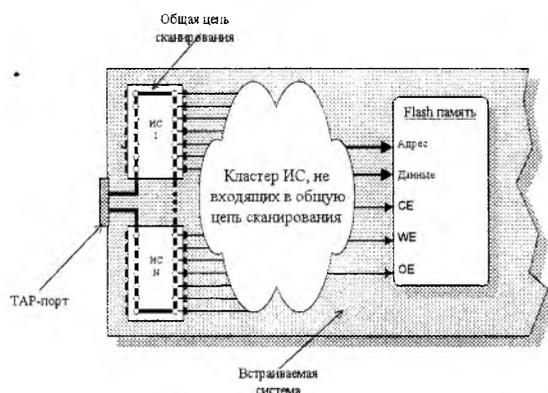


Рисунок 5. Пример ВС с кластером ИС, не входящих в общую цепь сканирования

Рассмотрим основные варианты реализации JTAG-контроллеров, которые могут быть применены для программирования Flash-памяти:

- 1) использование стандартных портов (USB, LPT, COM) персональных компьютеров и рабочих станций для подключения к TAP-портам ВС;
- 2) реализация JTAG-контроллера в качестве платы расширения системной шины (ISA, PCI);
- 3) реализация JTAG-контроллера в качестве автономного устройства.

Первый вариант представляет собой самый экономичный способ реализации контроллера. При этом основные операции обмена данными с TAP-портом возлагаются на программную часть. Она представляет собой набор из системного драйвера стандартного порта и пользовательского приложения, которое может быть написано как на языках типа STAPL, так и на языках программирования высокого уровня (C, PASCAL). К недостаткам такой реализации следует отнести низкую скорость обмена данными, программное формирование тактируемого сигнала ТСК и программное управление переключением состояний TAP-контроллеров.

Вариант реализации JTAG-контроллера в качестве платы расширения является наиболее эффективным с точки зрения быстродействия. Такой подход позволяет аппаратно реализовывать на больших частотах формирование управляющих сигналов для TAP-контроллеров, осуществлять быстрое формирование значений цепей сканирования по заданным алгоритмам программирования Flash-памяти. Передача данных между программными модулями и контроллером осуществляется посредством 16/32-разрядной системной шины, используя различные режимы работы, такие как BURST и DMA.

Третий вариант реализации контроллера отличается от предыдущего лишь тем, что запись инициализирующего или иного кода осуществляется предварительно в модули памяти автономного JTAG-контроллера. Такой подход наиболее эффективен при производстве ВС, при котором необходимо осуществлять массовое программирование Flash-памяти.

Последние два из рассмотренных вариантов реализации JTAG-контроллеров обладают общим недостатком - относительно высокой стоимостью аппаратуры.

В заключении рассмотрим основные факторы, влияющие на скорость программирования Flash-памяти. В общем случае время программирования через TAP-порт может быть вычислено по формуле 1:

$$T_{prog} = (1/V_{tck}) * D_{bs} = D_{bs}/V_{tck}, \quad (1)$$

где

T_{prog} - время программирования, V_{tck} - частота сигнала ТСК, D_{bs} - количество информационных бит, входящих в состав цепей сканирования, необходимых для осуществления заданной операции. В свою очередь параметр D_{bs} зависит от ряда значений (формулу 2):

$$D_{bs} = (L_{bs} + N_{tap}) * N_{flash} * N_{oper}, \quad (2)$$

где

L_{bs} - дина общей цепи сканирования,

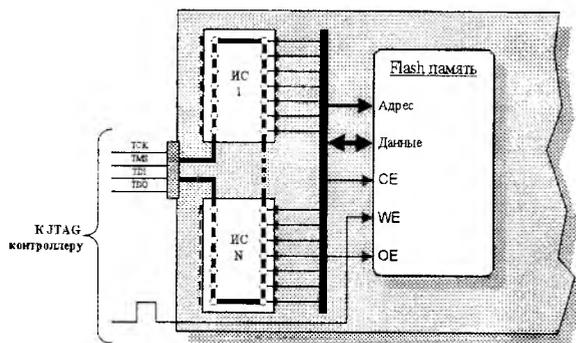


Рисунок 6. Технология AutoWrite

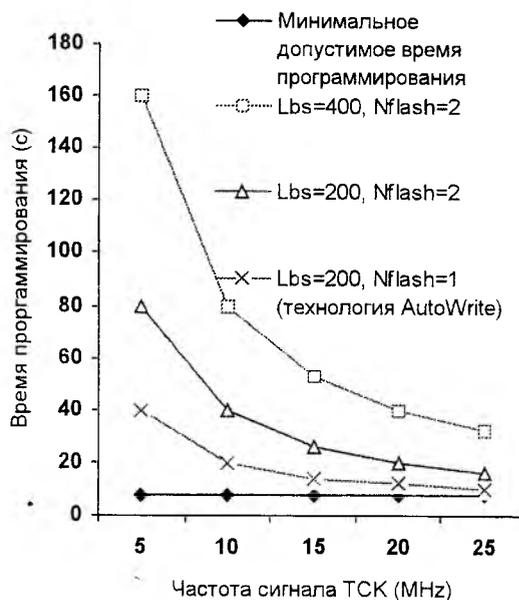


Рисунок 7. Время программирования микросхемы Flash-памяти серии ATMEL AT49F1604 с организацией 1Mx16

N_{tap} - количество управляющих бит, необходимых для переключения состояний TAP контроллера (как правило значение $N_{tap} \ll L_{bs}$ и им можно пренебречь),

N_{flash} - количество последовательных цепей сканирования для осуществления одной операции с Flash-памятью,

N_{oper} - количество операций. Так, например, для осуществления $N_{oper}=100'000$ операций записи для Flash-памяти ($N_{flash}=3$) через цепь сканирования длиной $L_{bs}=200$ разрядов, на частоте сигнала ТСК $V_{tck}=10\text{MHz}$, понадобится $T_{prog}=(200*3*100'000)/10*7 = 6$ секунд.

Следует отметить, что приведенные формулы 1 и 2 справедливы в случае непрерывной подачи данных на информационный вход TDI в соответствии с изменениями сигнала ТСК.

Пользователи ВС для ускорения процесса программирования Flash-памяти с использованием JTAG-контроллера могут прибегать к увеличению частоты сигнала ТСК, не превышая предельно допустимого значения (как правило, такую информацию предоставляют фирмы-разработчики ИС). Однако, существуют некоторые нововведения, например предложенные фирмой JTAG Technologies [6], позволяющие существенно ускорить процесс программирования. Одна из идей этой фирмы заключается в аппаратной реализации управления контрольным сигналом записи WE для микросхем Flash-памяти (технология AutoWrite (TM)) (Рис. 6.). При этом на запись одного слова памяти тратится не 2-3 полных цепи сканирования, а всего лишь одна.

На Рис.7 приведен график зависимости времени программирования Flash-памяти от частоты сигнала ТСК и длины общей цепи сканирования.

4. ЗАКЛЮЧЕНИЕ

В данной статье были рассмотрены особенности проектирования встраиваемых систем, которые находят все большее применение в настоящее время. Были показаны особенности

использования механизма граничного сканирования JTAG при тестировании и отладке. В статье были приведены современные аспекты программирования Flash-памяти с использованием JTAG-технологий.

ЛИТЕРАТУРА

- [1]. Ярмолик В.Н., Калоша Е.П., Быков Ю.В., Климец Ю.В., Иванюк А.А. Проектирование самотестируемых СБИС // науч. Монография, в 2 т. – Мн.: БГУИР, 2001.
- [2]. Использование интерфейса JTAG для отладки встраиваемых систем / Ключев А.О., Коровьякова Т.А., Платунов А.Е. // Изв. ВУЗов. Приборостроение. 1998. Т41, №5 С. 45-50.
- [3]. Угрюмов Е.П. Цифровая схемотехника // СПб.: ВHV - Санкт-Петербург, 2000. - 528 с.
- [4]. Standard Test and Programming Language (STAPL), Electronic Industries Alliance JEDEC Solid State Technology Association // URL: <http://www.jedec.org>.
- [5]. JAM Programming and Test Language Specification / Altera Corporation, 1997 // URL: <http://www.altera.com>.
- [6]. JTAG Boundary-scan, test and in-system programming solutions (IEEE 1149.1) / JTAG Technologies Inc. // URL : <http://www.jtag.com>.

КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ МНОГОМЕРНОГО КОНТРОЛЯ КАЧЕСТВА*

В.Н. Клячкин

Кафедра прикладной математики и информатики, Ульяновский государственный технический университет, Сев.Венец, 32, Ульяновск, 432027, Россия, e-mail: kl@ulstu.ru

АННОТАЦИЯ

В статье рассматриваются компьютерные технологии статистического контроля технологического процесса изготовления изделия, качество которого характеризуется совокупностью показателей, коррелированных между собой.

Для принятия решения по управлению процессом предлагается использование как совокупности одномерных контрольных карт по каждому показателю, так и специальных инструментов многомерного контроля – карт Хотеллинга, многомерных карт кумулятивных сумм и экспоненциально взвешенных скользящих средних.

Анализируются возможности универсального программного обеспечения (электронных таблиц и

статистических пакетов) для проведения одномерного и многомерного статистического контроля.

Приведено описание и технология работы пользователя специального пакета, разработанного для многомерного контроля технологического процесса.

1. ВВЕДЕНИЕ

Одна из важнейших задач управления предприятием – обеспечение качества выпускаемой продукции. Концепция всеобщего менеджмента качества, в основе которой – стандарты ИСО серии 9000, требует сертификации систем качества предприятия с целью удовлетворения потребностей заказчика.

* Работа выполнена в рамках научно-технической программы Министерства образования РФ «Научные исследования высшей школы по приоритетным направлениям науки и техники» (проект 02.01.002)