

## Список литературы

1. Автоматное управление асинхронными процессами в ЭВМ и дискретных системах / Под ред. В.И. Варшавского. — М.: Наука, 1986. — 398 с.
2. *Филин А.В., Степченко Ю.А.* Компьютеры без синхронизации // Системы и средства информатики. Вып. 9. — М.: Наука, 1999. — С. 247–261.
3. *Hauck S.* Asynchronous design methodologies: An overview // Proc. of the IEEE. — 1995. — V. 83, № 1. — P. 69–93.
4. *Степченко Ю.А., Дьяченко Ю.Г., Петрухин В.С., Филин А.В.* Цена реализации уникальных свойств самосинхронных схем // Системы и средства информатики. Вып. 9. — М.: Наука, 1999. — С. 261–292.
5. *Рождественский Ю.В., Морозов Н.В., Степченко Ю.А., Рождественскене А.В.* Универсальная подсистема анализа самосинхронных схем // Наст. сборник.
6. Annual report 2002. TIME Laboratory. — Grenoble: Techniques of informatics and microelectronics for computer architecture, 2003. — P. 39–53. (<http://time.imag.fr>.)
7. *Woods J.V., Day P., Furder S.B. et al.* AMULET1: an asynchronous ARM microprocessor // IEEE Transactions Computers. — 1997. — V. 46, № 4. — P. 385–397.
8. *Степченко Ю.А., Денисов А.Н., Дьяченко Ю.Г., Гринфельд Ф.И., Филимоненко О.П., Фомин Ю.П.* Библиотека элементов БМК для критических областей применения // Системы и средства информатики. Вып. 14 — М.: Наука, 2004. — С. 318–361.
9. *Степченко Ю.А., Петрухин В.С., Дьяченко Ю.Г.* Опыт разработки самосинхронного ядра микроконтроллера на базовом матричном кристалле // Тр. конференции «Проблемы разработки перспективных микроэлектронных систем — 2005» / Под общ. ред. А.Л. Стемпковского — М.: ИППМ РАН, 2005. — С. 235–242.
10. Исследование программируемости, алгоритмических и схемотехнических проблем проектирования рекуррентных компьютеров. Шифр: «Парсек». № г.р. 01.2.00.412412. Отчёт о НИР (промежуточный). — М.: ИПИ РАН, 2005. — 250 с.

УДК 621.3.049.77+004.312

## СИСТЕМА ТЕСТИРОВАНИЯ САМОСИНХРОННЫХ МИКРОСХЕМ<sup>1</sup>

*В.С. Петрухин, Ю.А. Степченко, Н.В. Морозов,  
Д.Ю. Степченко*

Рассмотрены основные проблемы сопряжения контрольно-измерительного оборудования с самосинхронными микросхемами. Для проведения тестирования и сравнительных испытаний синхронного и самосинхронного образцов микросхем в лабораторных условиях рассмотрена структура аппаратных и программных средств системы автоматизированного тестирования опытных кристаллов (САТОК). Детально представлен интерфейс взаимодействия пользователя с системой тестирования.

### 1. Введение

С ростом степени интеграции традиционные системы синхронизации становятся всё более громоздкими и малоэффективными. Используемая в них глобальная синхронизация вызывает ряд проблем: недоиспользование производительности (из-за необходимости ориентации на худший случай); повышение вероятности сбоя вследствие роста числа «точек» арбитража; отрицательное воздействие параметрической неустойчивости элементов; трудность выполнения требований к точности и стабильности частоты синхронизации и ряд других [1]. Это заставляет разработчиков обращаться к альтернативным принципам координации взаимодействия компонентов системы: асинхронному, квази-самосинхронному и самосинхронному (СС). Многообещающим представляется использование СС-подхода (построение СС-схем, работа которых не зависит от задержек их элементов), особенно для разработки гарантоспособной аппаратуры [2].

Для подтверждения декларируемых достоинств СС-схемотехники ИПИ РАН разработал и совместно с НПК «Технологиче-

<sup>1</sup> Работа выполнена при финансовой поддержке по Государственному контракту № 1.4/03 (регистрация РАН: № 10002-251/ОИТВС-04/103-098/260503-201).

ский центр» МИЭТ (г. Зеленоград) изготовил синхронный и самосинхронный варианты тестового БМК-кристалла «Микроядро», который реализует функции вычислительного ядра 8-рядного микроконтроллера PIC18CXX, широко используемого в отечественных разработках [2]. СС-схемы сохраняют работоспособность (устойчивую работу без сбоев) в предельно широких условиях эксплуатации, ограничиваемых только физическим сохранением переключательных свойств элементной базы. Тестирование и проведение испытаний таких схем предъявляет к контрольно-испытательному оборудованию определённые требования.

Для проверки синхронных больших интегральных схем (БИС) разработана широкая номенклатура контрольно-испытательного оборудования [3, 4]. Взаимодействие тестового оборудования и проверяемой БИС осуществляется синхронно по сигналам от тестового оборудования.

В отличие от синхронных схем, взаимодействие самосинхронных схем (СС-схем) с внешней средой и другими СС-схемами осуществляется по асинхронному (запрос-ответному) принципу [2] с фиксацией действительного окончания любого инициированного переходного процесса. Функционирование самих СС-схем происходит без участия каких-либо синхросигналов; генераторы могут быть использованы только для сугубо второстепенных целей, например для подсчёта астрономического времени.

Тестирование СС-схем с помощью типового (синхронного) оборудования не позволяет точно зафиксировать момент появления ответной реакции на тестовое воздействие, а значит, тестировать СС-схемы с максимальным быстродействием.

В некоторых измерительных системах, например HP82000, предусмотрена возможность смены вектора тестирования при появлении определённого сигнала от испытуемой микросхемы. Такое взаимодействие тестового оборудования и испытуемой микросхемы можно оценить как асинхронное. Этот режим обеспечивает возможность проверки БИС только в небольшом диапазоне изменения параметров.

Чтобы решить проблему тестирования микросхем с максимальным быстродействием, необходимо реализовать самосинхронную процедуру взаимодействия СС-схем и тестового оборудования, а это требует существенной доработки тестового оборудования. Кроме того, необходимо обеспечить мобильность тестовой

вой системы для демонстрации работы микросхем на площадках заказчика и в лабораторных условиях.

Было принято решение спроектировать контрольно-испытательное оборудование для проведения экспериментов с самосинхронными схемами.

Процедура тестирования схем была разделена на две задачи:

1) отбраковка и тестирование СС-схем на предприятии-изготовителе;

2) тестирование и испытания СС-схем у заказчика.

Как правило, отбраковка микросхем предназначена для выявления дефектов микросхем, вызванных технологическими нарушениями, и осуществляется на типовом тестовом оборудовании. При этом нет необходимости тестировать СС-схемы на максимальном быстродействии: достаточно подавать тестовые воздействия с частотой, существенно отличающейся от максимальной, с гарантией появления необходимых реакций на выводах СС-схем для последующего сравнения с эталоном на тестовом оборудовании.

Тесты для БИС «Микроядро» были разработаны в системе проектирования вентиляционных матриц «Ковчег». Они создавались ручным способом, и с помощью подсистемы моделирования автоматически генерировались реакции микросхемы на входные тестовые воздействия.

СС-схемы отбраковывались на заводе-изготовителе на измерительной системе HP82000-D50.

Как указано выше и в публикации [4], после отбраковки СС-схем необходимо проводить испытания синхронного и самосинхронного вариантов реализации БИС «Микроядро», в том числе и на максимально возможном быстродействии самосинхронных микросхем. Для этого разработано средство тестирования и испытаний САТОК (система автоматического тестирования опытных кристаллов).

## 2. Аппаратные средства

Для реализации самосинхронного взаимодействия испытательного оборудования и БИС «Микроядро» средства генерации тестовых воздействий непосредственно введены в состав БИС (рис. 1). На внешние контакты БИС подаётся только сигнал пуска, и снимается с них сигнал об окончании переходных процессов (окончание вычислений). В качестве генераторов тестов ис-

пользуются самосинхронные счётчики IDCL (младшая тетрада), IDCH (старшая тетрада). В каждом цикле выполнения команд значение счётчиков увеличивается на единицу, и тем самым обеспечивается перебор всех возможных сочетаний операндов на входе вычислителя.

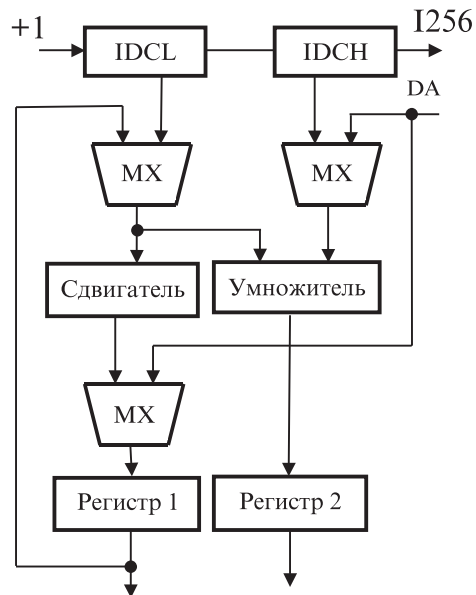


Рис. 1. Структура вычислителя

Время выполнения команды в СС-схемах зависит от сочетания и повторения входных операндов на входе вычислителя БИС «Микроядро». Чтобы получить усреднённое значение времени выполнения команды, введён сигнал I256, индицирующий выполнение 256 циклов команд.

Для минимизации финансовых и интеллектуальных затрат идеальным решением является использование готовых аппаратно-программных средств. Стоимость аппаратуры зависит от масштабов производства.

Специализированная аппаратура, выпускаемая малыми тиражами, обладает высокой стоимостью. Вполне логично строить такие системы на базе персональных компьютеров, выпускаемых большим тиражом. Поэтому мы построили комплекс аппаратно-программных средств САТОК на основе персонального компью-

тера (ПК) (рис. 2). Однако сам по себе персональный компьютер не имеет большого количества управляемых однобитных портов.

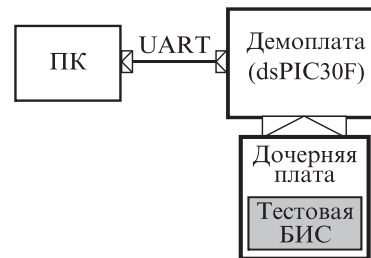


Рис. 2. Структура аппаратных средств

В нашем случае необходимо иметь как минимум 60 таких портов. Для ПК с этой целью изготавливаются специализированные дорогостоящие платы. Поскольку в отдельных случаях необходимо обеспечить проверку микросхем в автономных условиях (без ПК), для организации непосредственного взаимодействия с микросхемой нами использована демонстрационная

плата dsPIC30F, содержащая 68 однобитных портов и позволяющая быстро и эффективно реализовать контрольно-испытательное оборудование.

Непосредственным элементом сопряжения демонстрационной платы с микросхемой является специальная дочерняя плата с тестовой БИС, изготавливаемой на производстве ГУ НПК «Технологический центр» МИЭТ и служащей для непосредственной распайки микросхемы. Таким образом, для реализации аппаратных средств САТОК необходимо изготовить только дочернюю плату.

### 3. Программные средства

Для тестирования БИС «Микроядро» в составе тестового комплекса САТОК разработаны программные средства, включающие в себя:

- программу «Комплекс тестовый» (КТ), реализующую интерфейс с пользователем и выполненную для ПК на платформе Windows;

- программу «Комплекс управления» (КУ), осуществляющую управление тестовой БИС через демонстрационную плату dsPICDEM 1.1.

Обмен данными между ПК (программа КТ) и dsPICDEM 1.1 (программа КУ) происходит с помощью порта UART.

От ПК к демонстрационной плате пересылаются управляющие воздействия для передачи на тестируемую БИС «Микроядро». Воздействия формируются оператором с помощью про-

граммы КТ. В режиме автоматического тестирования на демонстрационную плату передаётся заранее подготовленный набор тестовых воздействий.

В режиме работы по шагам возможна пооператорная отладка.

От демонстрационной платы к ПК возвращается информация о текущем состоянии контактов БИС «Микроядро» и реакциях на подаваемые воздействия. Эта информация снимается программой КУ непосредственно с выводов БИС «Микроядро».

Главное окно программы КУ (рис. 3) содержит четыре блока кнопок.

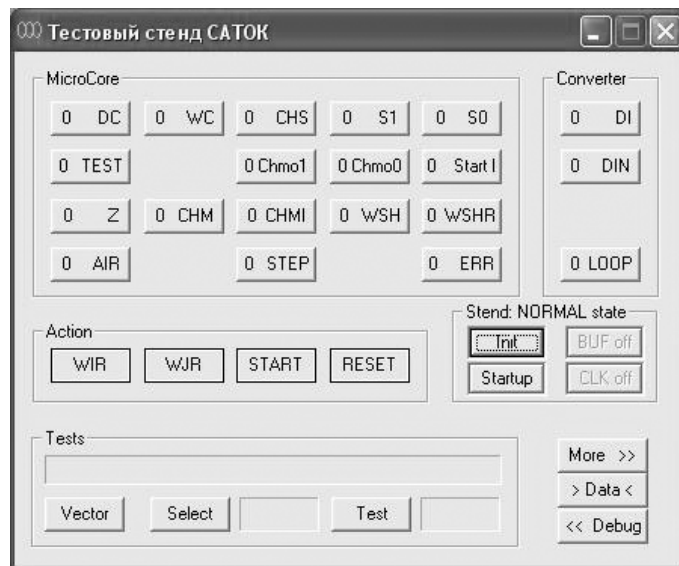


Рис. 3. Главное окно программы КУ

Блок MicroCore содержит кнопки сигналов для микроядра, блок Converter — кнопки сигналов для преобразователя. Справа на кнопках сигналов приведено обозначение сигнала, а слева — его текущее значение. Нажатие на такую кнопку приводит к изменению значения соответствующего сигнала на противоположное.

В блоке Action собраны все управляющие сигналы. Нажатие на кнопку посылает в микросхему соответствующий управляющий сигнал.

Блок Tests предназначен для работы с тестами. С помощью кнопки *Select* выбирается файл, содержащий тестовый набор

данных; имя загруженного файла отображается в верхней строке блока. В поле справа от кнопки *Select* отражается число строк в загруженном тестовом наборе данных. При нажатии на кнопку *Test* начинается исполнение теста. Строка, обрабатываемая в текущий момент, отражается в поле справа от кнопки *Test*.

Главное окно программы КУ содержит также три кнопки, по которым открываются дополнительные окна программы. По кнопке *More >>* справа открывается блок Control, состоящий из трёх блоков: View, Signal и Clock (рис. 4).

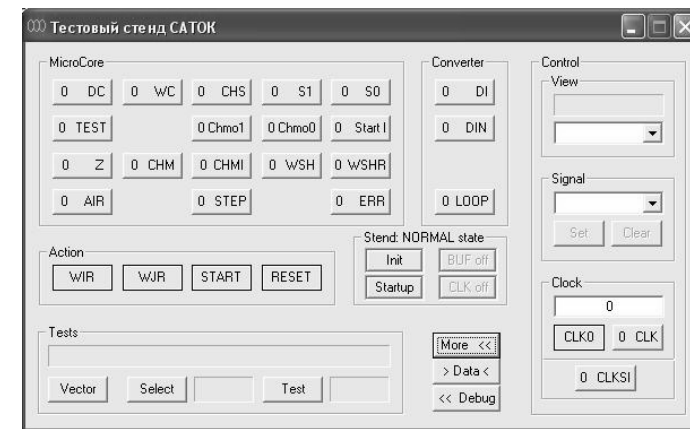


Рис. 4. Главное окно программы КУ с блоком Control

В блоке View в раскрывающемся списке можно выбрать сигнал, и в верхнем поле высветится его текущее значение.

В блоке Signal в раскрывающемся списке можно выбрать сигнал и кнопкой *Set* установить его в значение 1, а кнопкой *Clear* — в значение 0.

Блок Clock работает с тактовыми сигналами.

Ещё одна дополнительная кнопка *>Data<* главного окна программы КУ открывает окно Data, отражающее текущее состояние сигналов-данных микросхемы.

Блок Common предназначен для отображения общих сигналов-данных. Нажатие на кнопку блока обновляет значение соответствующего сигнала.

Безымянный блок содержит одну кнопку *xTEST*. Нажатие на неё позволяет изменить значение сигнала TEST (режим тестирования схемы).

Блок Core отображает состояние сигналов I256 (количество циклов вычислений равно или больше 256) и END (окончание переходного процесса).

Блок Converter содержит три кнопки, отображающих состояние сигналов-данных, и одну (ERR), задающую источник ошибок.

Третья дополнительная кнопка главного окна программы KU открывает поле вспомогательной информации DEBUG (рис. 5). В поле Command расшифровывается команда, посланная мик-

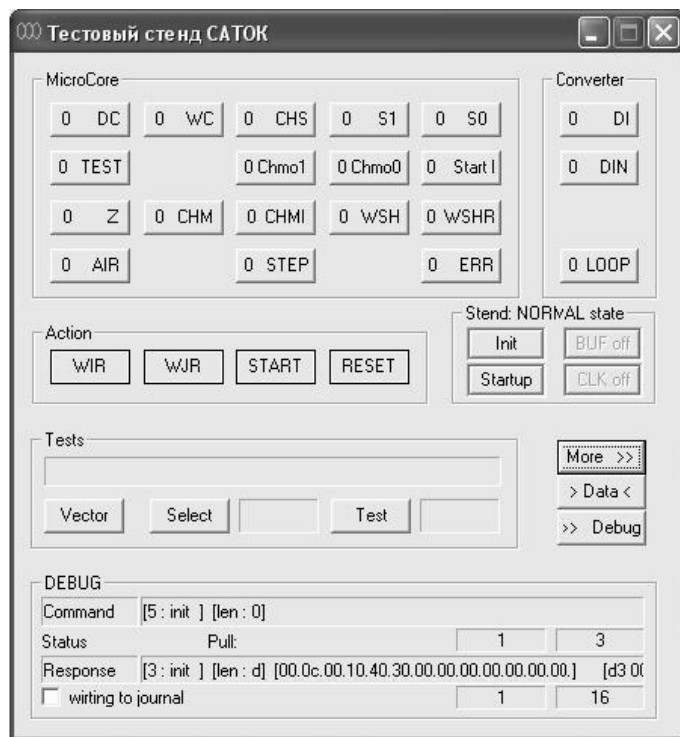


Рис. 5. Главное окно программы KU с блоком DEBUG

росхеме, а в поле Response отражается ответ. При установке галочки в пункте «writing to journal» все команды и ответы фиксируются в журнале тестирования.

Тестовые воздействия, подготовленные с помощью САПР «Ковчег», могут быть использованы для тестирования микрокристалла программой KU. Для этого предназначен блок Tests главного окна программы. Файл с тестовыми наборами данных

выбирается командой Select; появляется диалог выбора тестового набора. В раскрывающемся списке необходимо выбрать требуемый тестовый набор и подтвердить выбор нажатием кнопки ОК.

Имя загруженного файла отображается в верхней строке блока Tests. В поле справа от кнопки Select отражается число строк в загруженном тестовом наборе данных (рис. 6).

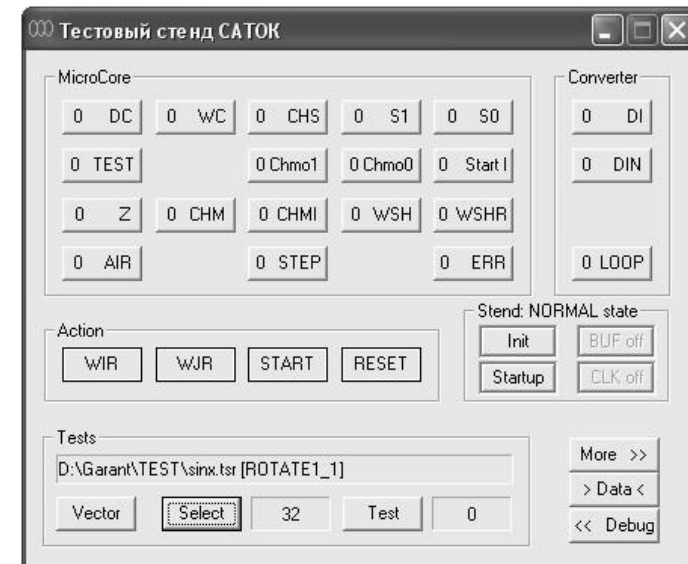


Рис. 6. Выбор тестового файла и тестового набора

При нажатии на кнопку Test начинается исполнение загруженного теста. Строка, обрабатываемая в текущий момент, отражается в поле справа от кнопки Test. По окончании прохождения теста программа сообщит о результате тестирования.

Как видно из описания, программа жёстко привязана к тестируемой БИС по сигналам и функциям; это значительно ограничивает программу, но имеет объективный характер.

#### 4. Заключение

Разработанные и изготовленные средства тестирования самосинхронных микросхем САТОК позволили осуществить не только проверку опытных образцов микросхем в условиях лаборатории, но и провести сравнительные испытания синхронных и самосинхронных образцов микросхем.

Для серийного производства самосинхронных микросхем необходимо разработать специальное контрольно-испытательное оборудование и более гибкие программные средства, которые можно будет настраивать на новые схемы.

#### Список литературы

1. *Филин А.В., Степченко Ю.А.* Компьютеры без синхронизации // Системы и средства информатики. Вып. 9. — М.: Наука, 1999. — С. 247–260.
2. *Степченко Ю.А., Петрухин В.С., Дьяченко Ю.Г.* Опыт разработки самосинхронного ядра микроконтроллера на базовом матричном кристалле: Сб. науч. трудов // Тез. конференции «Проблемы разработки перспективных микроэлектронных систем — 2005» / Под общ. ред. А.Л. Стемпковского. — М.: ИППМ РАН, 2005. — С. 235–242.
3. *Szekely V., Rencz M., Torok S., Courtois B.* IDDQ Testing of Submicron CMOS by Cooling. — [http://tima.imag.fr/publications/files/rr/its\\_52.pdf](http://tima.imag.fr/publications/files/rr/its_52.pdf).
4. *Mahlstedt U., Heinitz M., Alt J.* Test Generation for I Testing and Leakage Fault Detection DDQ in CMOS Circuits. — [www.tet.uni-hannover.de/papers/1992/92umah\\_1.pdf](http://www.tet.uni-hannover.de/papers/1992/92umah_1.pdf).

УДК 681.321.80

## ПОВЫШЕНИЕ ПРОИЗВОДИТЕЛЬНОСТИ МИКРОПРОЦЕССОРНЫХ СИСТЕМ ПУТЁМ ЭФФЕКТИВНОГО ИСПОЛЬЗОВАНИЯ КЭШ

*Б.З. Шмейлин, Е.Я. Попкова*

Рассмотрены различные методы повышения производительности микропроцессоров путём ускорения обращения к памяти. Особое внимание уделено методам повышения эффективности использования кэш-памяти при работе с различного вида прикладными программами.

### 1. Введение

В современных микропроцессорных системах значительное время, необходимое для обращения к памяти, является главным препятствием на пути повышения их производительности. При этом наблюдается тенденция к увеличению разрыва между производительностью центрального процессора и скоростью обращения к главной памяти. Так, за последние 10 лет производительность процессора выросла более чем в 12 раз, тогда как быстродействие памяти лишь удвоилось.

Для уменьшения времени обращения к памяти используется иерархическая структура памяти. Эффективность такой структуры в большой степени определяется производительностью встроенной в кристалл кэш-памяти.

Оптимизация использования кэш осуществляется по трём основным направлениям:

- 1) уменьшение времени обращения к кэш.
- 2) снижение потерь, вызванных промахом в кэш.
- 3) уменьшение относительного числа промахов в кэш

Существует два вида кэш: ассоциативный и прямого отображения.

Для уменьшения времени обращения к любому кэш следует уменьшить объём ассоциативного кэш, либо изменить его ассоциативность (увеличить число наборов частично-ассоциативной кэш-памяти при снижении числа строк в каждом наборе), либо