

Имплементация средств самотестирования модулей памяти DDR3 в ПЛИС Spartan3e

С.В. Волобуев, В.Г. Рябцев

Волгоградский государственный аграрный университет, akim.onoke@mail.ru

Аннотация — Приведены основные процедуры имплементации средств самотестирования модулей памяти DDR3 в ПЛИС семейства Spartan3e. Предложена структура мультипроцессорных средств MBIST, обладающих высоким быстродействием за счет одновременного формирования тестовых воздействий для нескольких смежных тактов за один период сигнала синхронизации.

Ключевые слова — средства самотестирования, модель счетчика, модули памяти, формат команд и микроопераций.

I. ВВЕДЕНИЕ

Почти во всех современных электронных устройствах применяются модули оперативной памяти. В настоящее время доминирующей технологией синхронной памяти, которую устанавливают в большинство ноутбуков, компьютеров и серверов является технология DDR3 SDRAM с удвоенной скоростью передачи данных. Она обеспечивает хорошее сочетание скорости и емкости, при более низком рабочем напряжении, сокращении потребляемой мощности и физических размеров при относительно низкой цене за бит.

Инженеры по производственным испытаниям предполагают, что ошибки в архитектуре микросхем памяти уже исправлены к тому времени, когда они поступит на сборку модулей. Следовательно, тесты памяти, которые будут выполняться во время производства, предназначены для определения соответствия модулей памяти требованиям пользователей [1, 2]. Тестовое диагностирование добавляется в процессы производства и сборки модулей памяти, выдает результат «годен / брак» и выполняется с заранее определенной скоростью системной шины.

Очевидно, что возможность тщательного тестирования, определения характеристик и диагностики модулей памяти является одной из самых актуальных проблем для разработчиков и производителей в данной отрасли [3].

Наиболее распространенной методом является алгоритмическое формирование тестов модулей памяти встроенными средствами самотестирования (Memory Built-In Self Test, MBIST). Подразумевают, что средства MBIST размещаются на кристалле и являются инструментом алгоритмического формирования тестов памяти [4]. Для повышения

надежности микросхем оперативной памяти возможно также применение встроенных средств саморемонта для восстановления работоспособности при многократных отказах [5-8].

Принципы построения структуры встроенных средств самотестирования для микросхем памяти статического типа приведены в [9-11], однако данная проблема для модулей памяти DDR3 типа раскрыта недостаточно подробно. Микросхемы памяти DDR3 имеют сложный интерфейс связи с центральным процессором, поэтому основная стратегия проектирования средств самотестирования заключается в сохранении неизменными логических элементов контроллера памяти и добавлении коммутаторов, которые в режиме самотестирования подключают к микросхемам памяти алгоритмический генератор (АЛГ), который по отношению к центральному процессору является дополнительным внешним устройством. Естественно, что тесты MBIST выполняются независимо от системных входов, поэтому применяются средства для изоляции схемы от системной шины во время выполнения теста.

Однако размещение встроенных средств самотестирования непосредственно в составе микросхем памяти DDR3 требует существенного изменения технологии изготовления и увеличения площадей кристаллов. Механизмы MBIST можно сконфигурировать для параллельного тестирования микросхем памяти, размещенных в модуле памяти, при этом компоненты средств самотестирования можно конструктивно выполнить в виде программируемой логической схемы (ПЛИС) или в виде заказной интегральной схемы (ASIC).

Целью работы является имплементация средств самотестирования модулей памяти в ПЛИС семейства Spartan3e для выполнения прототипного проектирования и проверки принципиальных проектных решений по составу и структуре, которые можно в будущем реализовать в готовое приложение высокого класса в виде заказной большой интегральной микросхемы при снижении затрат на ее разработку. Принципы построения архитектуры мультипроцессорного устройства тестового диагностирования полупроводниковой памяти, приведенные в [11, 12], можно взять за основу построения средств MBIST для модулей памяти DDR3.

II. АРХИТЕКТУРА СРЕДСТВ САМОТЕСТИРОВАНИЯ МОДУЛЕЙ ПАМЯТИ

На рис. 1 показана мультипроцессорная структурная схема средств MBIST, которая содержит программный счетчик Counter адреса команд и микроопераций, память инструкций Instruction Memory, контроллер Controller, генераторы рабочих операций Control Generator, генераторы кодов адреса Address Generator, генераторы кодов данных Data Generator, формирователи стробирования сопровождения кодов данных Sharper DQS, Sharper CER, регистры ответных реакций Data Memory, компараторы данных Data Comparator и мультиплексоры кодов операций Mx_COP, кодов адреса Mx_Addr, кодов данных Mx_Data и мультиплексоры Multiplexers для подключения модуля памяти к системной шине.

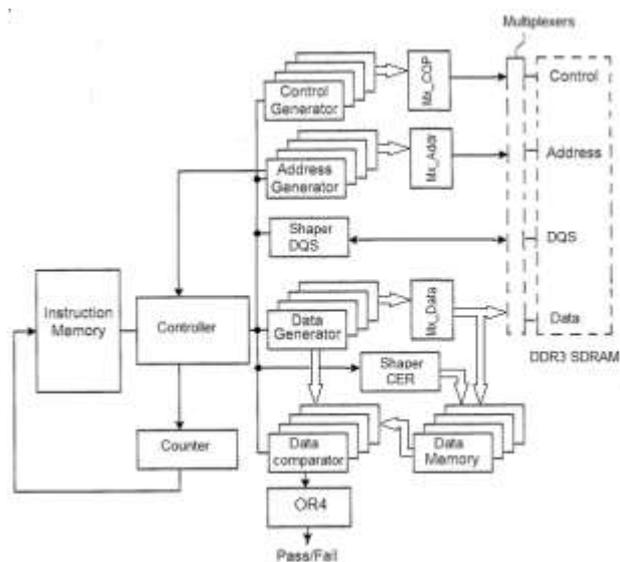


Рис. 1. Структурная схема средств MBIST

Тестируемая память DDR3 SDRAM и мультиплексоры Multiplexers в состав средств MBIST не входят, поэтому на рис. 1 они показаны пунктиром.

Средства MBIST имеют мультипроцессорную структуру, состоящую из управляющего процессора и многоканальных операционных процессоров, формирующих коды адреса, данных и осуществляющих сравнение считанных и эталонных данных. Для алгоритмического формирования программ тестов применяются программный счетчик Counter, память инструкций Instruction Memory и контроллер Controller. Сложность разработки средств MBIST заключается в том, что они одновременно должны формировать команды, задающие работу алгоритмического генератора и тестируемой памяти, формат которых для четырехканальных операционных процессоров приведен в табл. 1.

Разряды 3...0 поля команд применяются для кодирования команды, а разряды 15...4 позволяют формировать сигналы $We_n[3:0]$, $Cas_n[3:0]$, $Ras_n[3:0]$ для обращения к тестируемой микросхеме памяти в четырех смежных тактах. Коды F1 и F0 применяются для выбора флагов условных переходов, которые формируются четырехканальными операционными процессорами. Коды полей W1, Ri Ai, применяются для формирования вспомогательных сигналов при записи, считывании и сравнении данных в четырех смежных тактах обращения к микросхеме памяти DDR3 типа.

Генераторы кодов адреса Address Generator обеспечивают сохранение кода адреса ($A:=A$), инверсию кода адреса ($A:=notA$), увеличение кода адреса на 1 ($A:=A+1$), уменьшение кода адреса на 1 ($A:=A-1$), занесение кода начального адреса ($A:=G$), занесение кода конечного адреса ($A:=N$), увеличение кода на 4 ($A:=A+4$), уменьшение кода на 4 ($A:=A-4$).

Таблица 1

Формат команд и микроопераций средств MBIST

Разряды микропрограммной памяти											
23	22	21	20	19	18	17	16	15...12	11...8	7...4	3...0
A1	R1	W1	A0	R0	W0	F1	F0	Ras_n [3:0]	Cas_n [3:0]	We_n [3:0]	Код операции
Разряды микропрограммной памяти											
43, 42	41...39	38...36	35...33	32...30	29	28	27	26	25	24	
MD0	MA3	MA2	MA1	MA0	A3	R3	W3	A2	R2	W2	
Разряды микропрограммной памяти											
60	59	58	57...50					49, 48	47, 46	45, 44	
B2	B1	B0	Адрес перехода					MD3	MD2	MD1	

Генераторы кодов данных Data Generator обеспечивают хранение данных ($D:=D$), инверсию данных ($D:=notD$), увеличение кода данных на единицу ($D:=D+1$) уменьшение на единицу ($D:=D-1$).

Выбор выполняемых микроопераций обеспечивается подачей соответствующих кодов микроопераций на входы генераторов. Для управления адресом команд программы теста применяются флаги условных

переходов $A \neq G$, $A \neq N$, а также код номера такта диагностирования, заданный в полях 17, 16 кода команды. Команды занесения данных STGA, STNA, STRA, STDR обеспечивают установку в заданное состояние регистров GA, NA, A и D, установленных в генераторах кодов адреса Address Generator и данных Data Generator.

III. МЕТОДЫ ПРОЕКТИРОВАНИЯ СРЕДСТВ MBIST

С учетом большого количества связей между компонентами сначала было разработано схемотехническое представление проекта средств MBIST, затем модель проекта и все его компоненты были скомпилированы на язык VHDL. Для имплементации средств самотестирования модулей памяти в ПЛИС spartan3e при помощи Навигатора проектов фирмы Xilinx потребовалось дополнить проект моделями таких логических элементов: ADD1, BUF16, BUF8, BUF8E, CB2CE, COMP16, INV16, INV8, FD8CE, FD16CE, M2_1, M4_1E, M8_1E.

Особые проблемы вызвала разработка модели двухразрядного двоичного счетчика CB2CE. На схеме были представлены два отдельных выхода Q0 и Q1, а в существующей модели его выходы были определены в виде двухразрядного вектора, поэтому пришлось доработать схему. Но все равно при компиляции проекта Навигатор проектов фирмы Xilinx выдал следующее сообщение:

Signal Q cannot be synthesized, bad synchronous description. (Сигнал Q не может быть синтезирован, плохое синхронное описание).

Процедуры функционирования счетчика CB2CE были доработаны и описаны в следующем виде:

```
begin
  process(C, CE, CLR)
  begin
    if (CLR = '1') then
      Q <= "00";
      R <= "00";
    elsif rising_edge(C) then
      if (CE='1') then
        P<= std2int(R)+1;
        Q<= std_logic_vector(to_unsigned(P, 2));
        P<= (P+1)mod 4;
      end if;
    end if;
  end process;
end CB2CE;
```

Временная диаграмма сигналов счетчика CB2CE приведена на рис. 2.

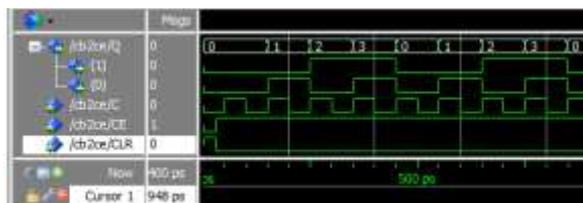


Рис. 2. Временная диаграмма счетчика CB2CE

Потребовалась доработка модели 8-ми разрядного счетчика CC8CLE, который может увеличивать свое содержимое на единицу, если сигнал $L='0'$, или параллельно заносить код, поступающий на его входы при $L='1'$. Для того чтобы к вектору добавить единицу, нужно сначала данный вектор преобразовать в целое число, прибавить к нему единицу, а полученный результат снова преобразовать в вектор, т.е. необходимо выполнить следующие преобразования:

$$Q \leq \text{int2std}(\text{std2int}(Q)+1, 8).$$

Однако при компиляции модели счетчика CC8CLE Навигатор проектов фирмы Xilinx выдал следующее сообщение:

int2std has two possible definitions in this scope (int2std имеет два возможных определения в данной области).

Для получения корректной модели была применена новая библиотека ieee.numeric_std.all, а процедуры функционирования счетчика CC8CLE были представлены в следующем виде:

```
begin
  process(C, CLR)begin
    if (CLR = '1') then
      Q <= (others => '0');
      R <= (others => '0');
    elsif (C'event and C='1') then
      if (L = '1') then
        Q <= D;
        P<= std2int(D)+1;
      elsif (CE='1' and L = '0') then
        R <= D;
        P<= std2int(R)+1;
        R <= D;
      end if;
      Q<= std_logic_vector(to_unsigned(P, 8));
      P<= (P+1)mod 256;
    end if;
  end process;
end CC8CLE;
```

Временная диаграмма сигналов счетчика CC8CLE приведена на рис. 3.



Рис. 3. Временная диаграмма счетчика CC8CLE

После устранения ошибок в проекте выполнено распределение входов/выходов средств MBIST на краевые выводы ПЛИС XC3S500E, которая размещена в корпусе PQ208 и имеет большое количество краевых контактов. Некоторые входы/выходы имеют специальное назначение. При выборе контактов ПЛИС необходимо учитывать графические и цветовые различия назначения контактов, приведенные на рис. 4. При некорректном выборе контактов Навигатор проектов выдает предупреждающее сообщение. После

завершения основных процедур имплементации проекта формируется файл программирования ПЛИС с расширением .bit. Структура проекта средств MBIST, главные процедуры Навигатора проектов ISE,

выполненные в процессе имплементации в ПЛИС, и сообщение об успешном завершении программирования приведены на рис. 5.

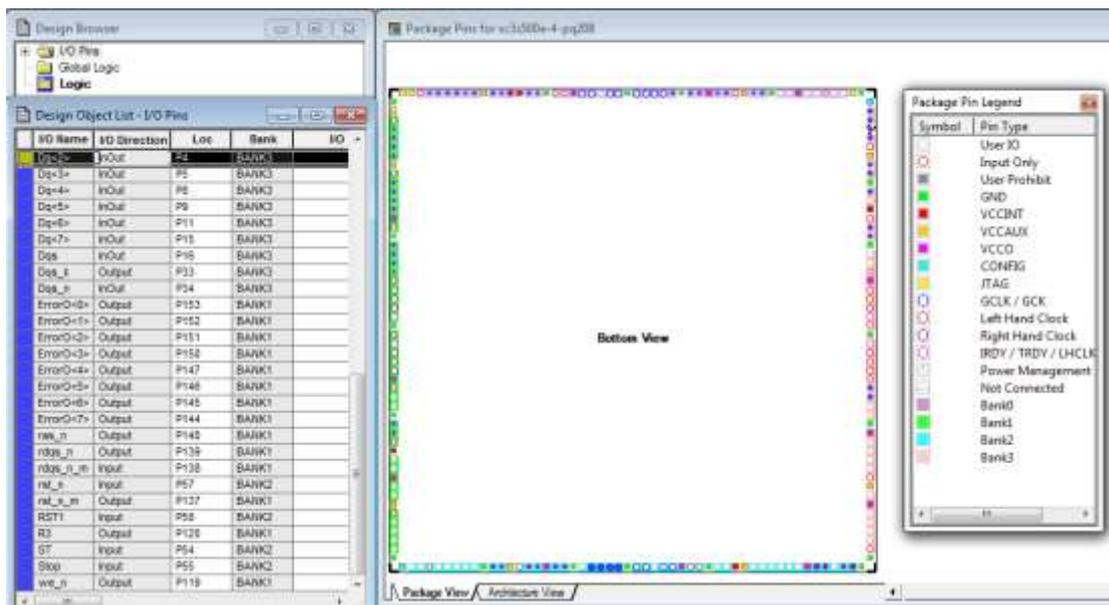


Рис. 4. Процесс распределения входов/выходов проекта

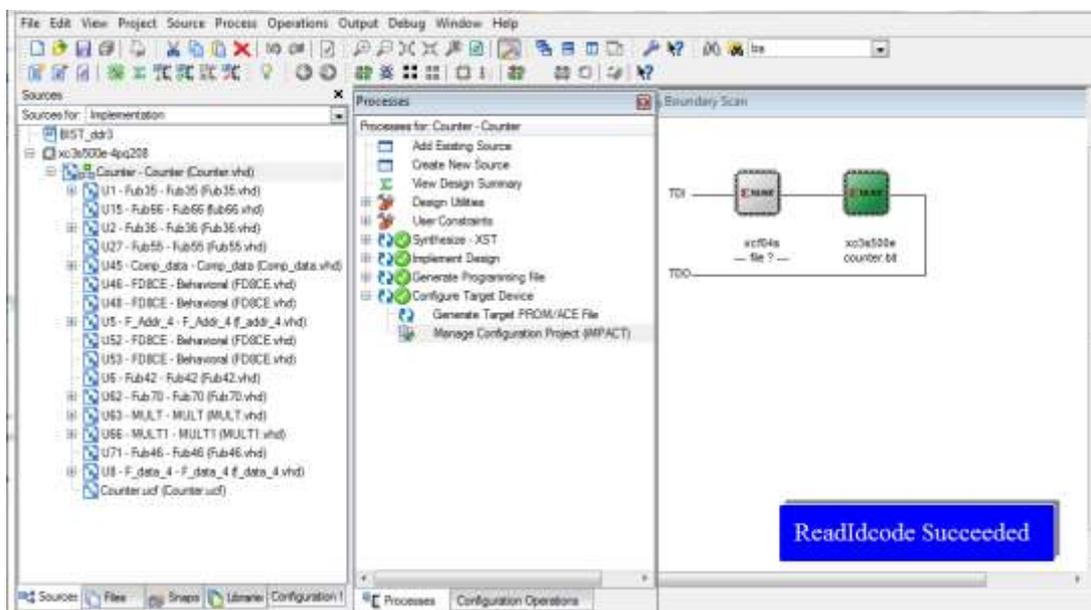


Рис. 5. Структура проекта средств MBIST и сообщение о завершении программирования

IV. РЕЗУЛЬТАТЫ ВЕРИФИКАЦИИ СРЕДСТВ MBIST

В среде Active-HDL была выполнена верификация VHDL-модели проекта средств MBIST. Для верификации проекта применялась VHDL-модель микросхемы памяти DDR3 типа фирмы Micron Technology, Inc. емкостью 1Gb 8-ми разрядных слов с параметром TCK_MIN, который равен 2,5 ns. Результаты верификации проекта в режиме записи

пакета данных длиной 4 слова приведены на рис. 6. Генераторы кодов данных Data Generator обеспечивают передачу на информационные входы модуля памяти «шахматного» кода. Формирователь Sharper DQS обеспечивает выдачу сопроводительного строба, фронты которого совпадают со серединой передаваемых данных, что обеспечивает запись данных в запоминающие ячейки модуля памяти.

На рис. 7 приведены результаты верификации проекта в режиме считывания пакета данных длиной 4 слова. Считанные из модуля памяти данные сопровождаются стробом Dqs_k, фронты которого совпадают с фронтами передаваемых данных. Применить строб Dqs_k, формируемый микросхемой памяти, не удалось, т.к. для его преобразования в серию стробирующих сигналов обычно применяют управляемые напряжением линии задержки, модели которых недоступны. Для фиксирования считанных

данных из последовательностных логических элементов был разработан формировательSharper CER, обеспечивающий генерацию стробов CER0-CER3 для приема данных в заданные моменты времени. Разработанные средства формирования двунаправленного синхросигнала Dqs_k для стробирования записываемых в микросхемы памяти DDR3 данных позволяют обрабатывать пакеты данных длиной 4 слова.

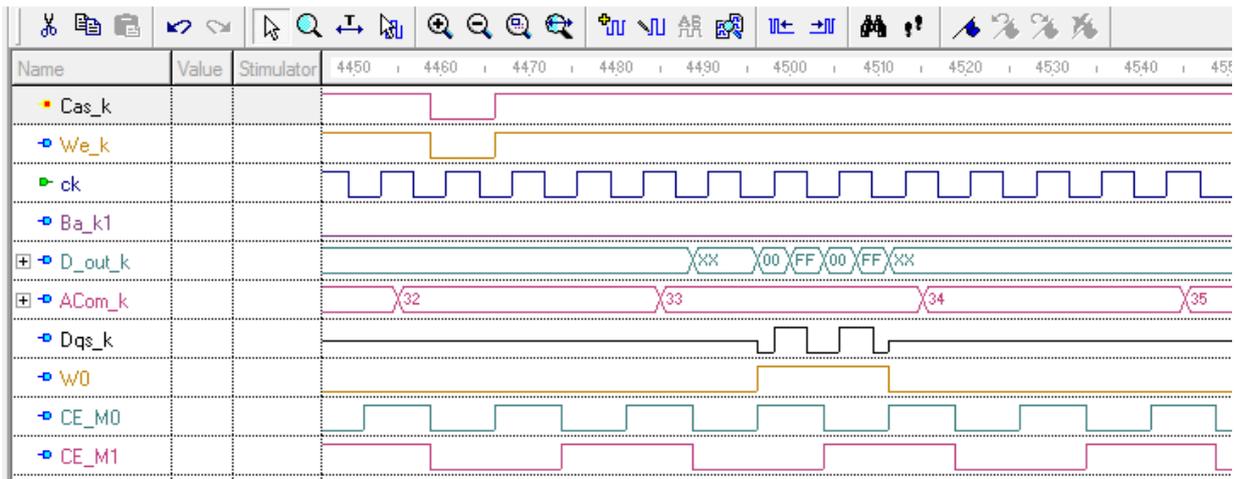


Рис. 6. Временная диаграмма средств MBIST в режиме записи пакета данных

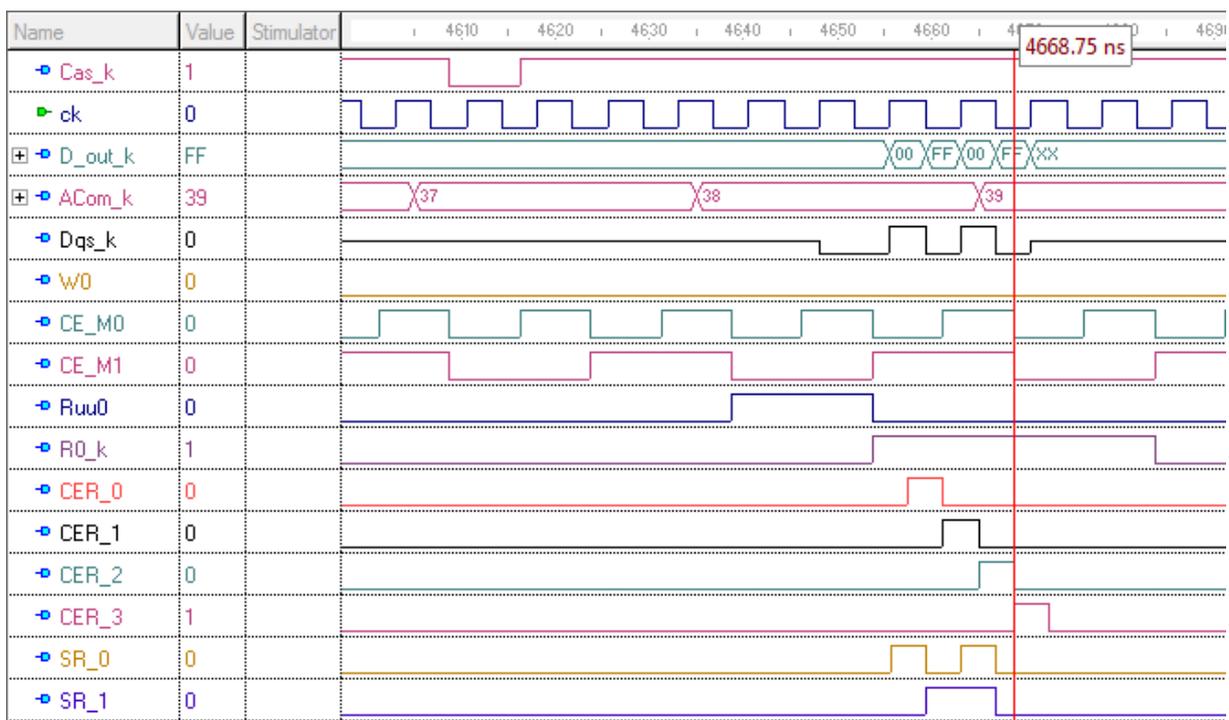


Рис. 7. Временная диаграмма средств MBIST в режиме считывания пакета данных

V. Выводы

В разработанных мультипроцессорных средствах самотестирования модулей памяти DDR3 частота формирования тестов ограничивается только временем переключения коммутаторов и выходных регистров. Разработанная структура средств MBIST позволяет формировать тестовые последовательности при передаче и приеме пакетов данных длиной 4 слова, а для выполнения тестов с применением пакетов данных большей длины количество каналов генераторов кодов адреса и данных, а также компараторов данных нужно увеличить. Получены следующие параметры средств MBIST, размещенных в ПЛИС XC3S500E: количество входов/выходов: 57 из 158 (36%); потребляемая мощность: 159 Mw; емкость памяти конфигурации проекта: 188 МВ.

Реализация встроенного самотестирования значительно снижает стоимость модулей памяти и повышает вероятность обнаружения неисправностей, так как тестирование производится на рабочих частотах и не требуется дорогостоящее внешнее диагностическое оборудование.

К недостатку средств MBIST следует отнести проблему гибкости, поскольку программа тестирования жестко запрограммирована в виде кодов постоянного запоминающего устройства, поэтому можно применять только ограниченное количество тестов, которые фиксируются при проектировании, кроме того ограничен набор микросхем DDR3, которые можно протестировать спроектированными средствами MBIST.

ЛИТЕРАТУРА

[1] Almadi M.K., Moamar D.N., Ryabtsev V.G. Methodology of Algorithms Synthesis of Memory Test Diagnosing // Proceedings of IEEE East-West Design & Test Symposium 2010 (EWDTS'10). St. Petersburg, 17-20 September 2010. – Pp. 366-370.

[2] Ryabtsev V., Evseev K., Almadi M. The Concept of Memory Device Diagnosis Algorithm Design// Journal of Multidisciplinary Engineering Science and Technology (JMEST), Vol. 3, Issue 10, October. 2016. – Pp. 5771-5774.

[3] Андриенко В.А., Рябцев В.Г., Уткина Т.Ю. Архитектура средств встроенного самотестирования микросхем

памяти // Проблемы разработки перспективных микро- и наноэлектронных систем - 2010. Сборник трудов / под общ. ред. А.Л. Стемповского. М.:ИППМ РАН, 2010. – С. 386-389.

- [4] Aswin A.M., Ganesh S. Sankar. Implementation and Validation of Memory Built in Self Test (MBIST) – Survey // International Journal of Mechanical Engineering and Technology, 10(3), 2019. – Pp. 153-160.
- [5] Ryabtsev V.G., Volobuev S.V. Implementation of Memory in a System on a Chip with Built-In Self-Testing and Self-Healing // Russian Microelectronics, 2020. Vol. 49. No. 7. – Pp. 527-531.
- [6] Волобуев С.В., Рябцев В.Г. Имплементация памяти в систему на кристалле со встроенными средствами самотестирования и восстановления // Изв. вузов. Электроника. – 2019. – Т.24. – №3. – С. 239-247.
- [7] Ryabtsev V.G., Volobuev S.V., Shubovich A.A. Fault-tolerant Architecture of Storage device for on-board spacecraft control systems // Russian Aeronautics. 2019. Т. 62. № 1. – Pp. 106-112.
- [8] Волобуев С.В., Евдокимов А.П., Рябцев В.Г. Принципы проектирования отказоустойчивых оперативных запоминающих устройств для космического применения // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2020. Вып. 2. С. 103-109.
- [9] Араkelов А.А., Сидоров Е.А., Бобков С.Г. Архитектура контроллера интерфейса памяти SRAM для обеспечения максимальной производительности коммутирующего устройства // Проблемы разработки перспективных микроэлектронных систем - 2005. Сборник научных трудов / под общ. ред. А.Л.Стемповского. М.:ИППМ РАН, 2005. – С. 229-234.
- [10] Волобуев С.В., Рябцев В.Г. Архитектура встроенных средств самотестирования микросхем памяти статического типа // Электронная техника. Серия 3. Микроэлектроника. 2019. Выпуск 3(175). – С. 15-20.
- [11] Евдокимов А.П., Рябцев В.Г., Меликов А.В. Принципы проектирования устройств тестового диагностирования быстродействующих микросхем и модулей полупроводниковой памяти // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2018. Вып. 2. С. 23-30.
- [12] Волобуев С.В., Евдокимов А.П., Меликов А.В., Рябцев В.Г., Шубович А.А. Методы и средства повышения надежности модулей памяти компьютеров: монография/ – Волгоград: ФГБОУ ВО Волгоградский ГАУ, 2018. – 284 с.

Implementation of Self-Testing Tools for DDR3 Memory Modules in Spartan3e FPGA

S.V. Volobuev, V.G. Ryabtsev

Volgograd Agrarian State University, akim.onoke@mail.ru

Abstract — Designing self-testing tools for memory modules is a method of developing additional tools that eliminate the need for external expensive diagnostic systems, the cost of which is many times higher than the cost of the memory modules themselves. The implementation of self-testing tools increases the percentage of fault coverage, since testing is

performed at operating frequencies, and no external test equipment is required.

The paper describes the architecture of self-testing tools for DDR3 memory modules and features of the procedures for their implementation in programmable logic circuits. The project of self-testing tools for DDR3 memory modules is

implemented in the XC3S500E FPGA of the Spartan3e family.

The applied hardware-firmware method of forming tests combines high performance with low hardware costs, while reducing the memory capacity for storing test programs of the Marsh family. The structure of the system of commands for self-testing tools has been developed, which provides parallel encoding of working operations for accessing the tested memory and generating address and data codes. The diagnostic properties of the generated tests are improved due to their execution without skipping the clock cycles of accessing the memory module under test.

The features of the design route are revealed, which defines the stages of design procedures used at all stages of development, from the development and formalization of the idea to the testing of finished samples. The project of self-testing tools for DDR3 memory modules uses libraries of logic elements that adapt to the schematic representation of the components.

The results of prototype design with the implementation of the project on the FPGA allow you to check the correctness of design decisions on the composition and structure, which can be implemented in the future in a ready-made high-class application in the form of a custom large integrated circuit.

Keywords — self-testing tools, the counter model, memory modules, the format of commands and micro-operations.

REFERENCES

- [1] Almadi M.K., Moamar D.N., Ryabtsev V.G. Methodology of Algorithms Synthesis of Memory Test Diagnosing // Proceedings of IEEE East-West Design & Test Symposium 2010 (EWDTS'10), St. Petersburg, 17-20 September 2010. – Pp. 366-370.
- [2] Ryabtsev V., Evseev K., Almadi M. The Concept of Memory Device Diagnosis Algorithm Design// Journal of Multidisciplinary Engineering Science and Technology (JMEST), Vol. 3, Issue 10, October. 2016. – Pp. 5771-5774.
- [3] Andrienko V. A., Ryabtsev V. G., Utkina T. Yu. Architecture of means of built-in self-testing of memory chips // Problems of development of perspective micro-and nanoelectronic systems-2010. Collection of works / under the general editorship of A.L. Stempkovsky. Moscow: IPPM RAS, 2010. – Pp. 386-389.
- [4] Aswin A.M., Ganesh S. Sankar. Implementation and Validation of Memory Built in Self Test (MBIST) – Survey // International Journal of Mechanical Engineering and Technology, 10(3), 2019. – Pp. 153-160.
- [5] Ryabtsev V.G., Volobuev S.V. Implementation of Memory in a System on a Chip with Built-In Self-Testing and Self-Healing // Russian Microelectronics, 2020. Vol. 49. No. 7. – Pp. 527-531.
- [6] Volobuev S. V., Ryabtsev V. G. Implementation of memory in a system on a crystal with built-in means of self-testing and recovery. Izv. vuzov. Electronics. – 2019. – Vol. 24. – No. 3. - Pp. 239-247.
- [7] Ryabtsev V.G., Volobuev S.V., Shubovich A.A. Fault-tolerant Architecture of Storage device for on-board spacecraft control systems // Russian Aeronautics. 2019. T. 62. № 1. – Pp. 106-112.
- [8] Volobuev S.V., Evdokimov A.P., Ryabtsev V.G. Design principles for fault-tolerant random access memory for space applications // Problems of Perspective Micro- and Nanoelectronic Systems Development - 2020. Issue 2. P. 103-109. doi:10.31114/2078-7707-2020-2-103-109
- [9] Arakelov A. A., Sidorov E. A., Bobkov S. G. Architecture of the SRAM memory interface controller for ensuring the maximum performance of a switching device // Problems of developing promising microelectronic systems-2005. Collection of scientific papers / under the general editorship of A. L. Stempkovsky. Moscow: IPPM RAS, 2005. Pp. 229-234.
- [10] Volobuev S. V., Ryabtsev V. G. Architecture of the built-in means of self-testing of memory chips of static type. Series 3. Microelectronics. 2019. Issue 3(175). – Pp. 15-20.
- [11] Evdokimov A.P., Ryabtsev V.G., Melikov A.V. Principles of Designing Devices for Test Diagnosing of High-speed Microchips and Semiconductor Memory // Problems of Perspective Micro- and Nanoelectronic Systems Development. 2018. Issue 2. P. 23-30. doi:10.31114/2078-7707-2018-2-23-30
- [12] Volobuev S. V., Evdokimov A. P., Melikov A.V., Ryabtsev V. G., Shubovich A. A. Methods and means of improving the reliability of computer memory modules: monograph / - Volgograd: FGBOU VO Volgogradsky GAU, 2018. – 284 p.