УДК 004.3, 004.41 DOI: 10.31114/2078-7707-2021-2-63-69

Динамическая модификация внутреннего программного обеспечения встраиваемых устройств для решения задач обратной разработки

А.А. Гладких, А.И. Власов, Д.А. Узеньков, Т.М. Фатхутдинов

МГТУ им. Н.Э. Баумана, г. Москва, vlasov@iu4.ru

***Аннотация* — Рассмотрены особенности хранения внутреннего программного обеспечения различных электронных устройств. Предложено использование динамической модификации данных при решении различных задач обратной разработки электронных устройств. Описанный способ имеет преимущество по скорости реализации и подстройке модификации хранимых в запоминающем компоненте устройства данных относительно методов статической модификации. Основное внимание уделено архитектуре устройства динамической модификации данных запоминающего устройства с параллельным асинхронным интерфейсом. Показаны возможности по применению программируемых логических интегральных схем. Основная проблема методов модификации внутреннего программного обеспечения электронных устройств состоит в том, что они не подразумевают изменения конфигурации модификации данных запоминающего компонента устройства в процессе его работы устройства, т.е. являются статическими. Это следствие использования подхода статической модификации данных. Авторами предложена стратегия «динамического подхода с применением программируемой логической интегральной схемы», позволяющая устранить вышеуказанную проблему. Структура системы с использованием предложенной стратегии динамической модификации данных системы на основе программируемой логической интегральной схемы позволяет модифицировать данные запоминающего устройства, передаваемые по интерфейсу связи от запоминающего к исполняющему компоненту, без физического вмешательства в состояние постоянно хранимых данных внутреннего программного обеспечения и, как следствие, использовать различные конфигурации модификации в течение одного цикла работы обратно разрабатываемого устройства. Предложенный метод динамической модификации данных внутреннего программного обеспечения реализован в аппаратно-программном комплексе, встраиваемом в разрыв между постоянным запоминающим и исполняющим компонентами устройства. Он позволяет передать заранее подготовленные данные взамен ответа запоминающего устройства. Целевая ответная посылка запоминающего устройства определяется при предварительной конфигурации в бинарном виде. Метод может быть использован для решения различных задач при отладке в динамическом режиме программных алгоритмов встраиваемого электронного устройства.**

***Ключевые слова* — модификация данных, динамическая отладка, обратная разработка, программируемые логические интегральные схемы, встраиваемые устройства, исследование встраиваемых устройств, динамическая модификация данных, модификация внутреннего программного обеспечения встраиваемых систем, модификация данных запоминающих устройств.**

I. Введение

С развитием информационных технологий и переходе промышленности к Индустрии 4.0 набирает обороты тенденция глобальной автоматизации – производственных линий, технологических и логистических процессов управления различной критической инфраструктурой. Прямо пропорционально росту числа автоматизированных систем увеличивается потребность в анализе таких устройств на безопасность [1] – поиск различных неявных ошибок внутреннего программного обеспечения (ВПО), защищенности устройства от воздействия недоброжелателя, а также недокументированных возможностей, тем или иным образом способных повлиять на работоспособность объектов критической инфраструктуры. Исследователи безопасности работают непосредственно с ВПО устройства в подавляющем большинстве случаев. Зачастую производитель устройства желает скрыть различные недоработки во избежание возможных репутационных потерь. Тогда вводятся дополнительные проверки целостности и подлинности ВПО, что сильно усложняет процессы отладки [2] при проведении анализа защищенности, обратной разработки устройства [3].

Таким образом, все чаще при проведении анализа защищенности встраиваемых систем автоматизации критической инфраструктуры возникает задача деактивации/обхода противоотладочных механизмов. Трудоемкость данной задачи довольно высока, то есть один из основных процессов при проведении обратной разработки устройства становится дороже во времени. Классически данная задача решается путем полного перебора возможных алгоритмов работы противоотладочных механизмов, статической «прямой» модификации внутреннего программного обеспечения устройства для их полного устранения. При этом каждая модификация зачастую подразумевает физическое извлечение микросхемы

памяти, что сильно ограничивает количество возможных модификаций информации, хранимой в запоминающем устройстве (ЗУ) системы до его полного/частичного отказа [4].

Ускорение решения задачи модификации ВПО устройства может быть произведено при помощи применения подхода динамической, а не статической модификации. Подразумевается автоматизация процесса изменения ВПО устройства до такого уровня, что становится возможным контролируемое получение процессором различной информации в течение одного рабочего цикла из тех частей образа программного обеспечения, которые при применении подхода статической модификации не могли изменяться подобным образом.

При динамическом подходе к модификации ВПО появляется возможность передачи в процессор данных одних и тех же ячеек памяти в неизменном и модифицированном виде в зависимости от различных метрик – состояния на шине данных в течение времени, времени от старта и так далее. Тогда возможно передать в исполняющий компонент системы в момент проверки подлинности реальные данные, а в момент исполнения модифицированные.

Автоматизация подобных операций способна существенно сократить необходимые для разработки методов отладки ВПО встраиваемых систем критической инфраструктуры усилия, что критически снижает временную стоимость исследований на безопасность встраиваемых устройств автоматизации критической инфраструктуры.

В настоящее время на рынке не представлено аппаратно-программных комплексов, реализующих подобную функциональность. Существуют различные аппаратные эмуляторы микросхем памяти, при использовании которых появляется возможность статического модифицирования ЗУ системы программным образом – эмулируя ЗУ, такое устройство предоставляет возможность сократить время статической модификации, не перепаивая каждый раз микросхему, сокращает время изменения состояния ВПО. То есть не обладает возможностью модификации данных ЗУ исследуемой встраиваемой системы автоматизации, а только ускоряет процесс [5], с чем лучше справляется динамический подход.

В данной работе целью является повышение эффективности отладки внутреннего программного обеспечения устройств за счет применения динамического подхода к модификации данных ЗУ исследуемой системы.

Для решения поставленной задачи использованы методы анализа высокоскоростных интерфейсов передачи данных [6], реализации практически применяемых устройств на базе программируемых логических интегральных схем (ПЛИС), встраивания устройств съема в существующие линии передач и динамической модификации данных высокоскоростного интерфейса передачи.

II. Методы динамической модификации ВПО на базе ПЛИС

A. Аппаратный уровень реализации

Внутреннее программное обеспечение любого электронного устройства представляет собой машинный код, описанный с помощью набора байт. Данный набор передается в исполняющий компонент устройства, и задача динамической модификации состоит в том, чтобы изменять байты в различные моменты рабочего цикла. Для того, чтобы гарантировать изменение при условии бесконфликтной работы разрабатываемого устройства с целевой встраиваемой системой, необходимо контролировать канал передачи данных между запоминающим и исполняющим компонентами (рисунок 1).

Рис. 1. Концептуальная структурная схема целевой для встраивания системы

Возникает необходимость встроиться в интерфейс связи (рисунок 2), иначе реализовать подобный «шлюз» крайне трудоемко в силу законов физики в части электроники [7].

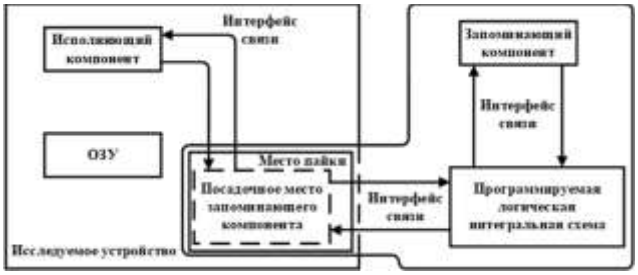

Рис. 2. Концептуальная структурная схема встраивания устройства в целевой интерфейс связи

При этом задержка, вносимая встраиваемым устройством, должна быть максимально мала, ведь частота работы современных высокоскоростных интерфейсов – от 50MHz, то есть период около 20наносекунд [8-10]. В то же время, программируемые логические интегральные схемы (ПЛИС) показывают свою эффективность при решении подобных задач за счет возможности произвольного конфигурирования передающего тракта, что невозможно сделать на

готовой интегральной микросхеме. Исключая дополнительные обвязочные цепи, распределительные тракты, то есть не используемые в конкретной узконаправленной задаче элементы, возможно добиться лучшего показателя вносимых задержек.

B. Программный уровень реализации

На программном уровне алгоритм работы данного устройства должен быть реализован в виде модели типа «конечный автомат» для исключения паразитных операций [11-13]. Алгоритм работы, на основе которого была описана данная модель, представлен на рисунке 3.

В соответствие концептуальным схемам, представленным на рисунках 2 и 3, было разработано и практически применено устройство на базе модуля программируемой логической интегральной схемы Intel Cyclone IV.

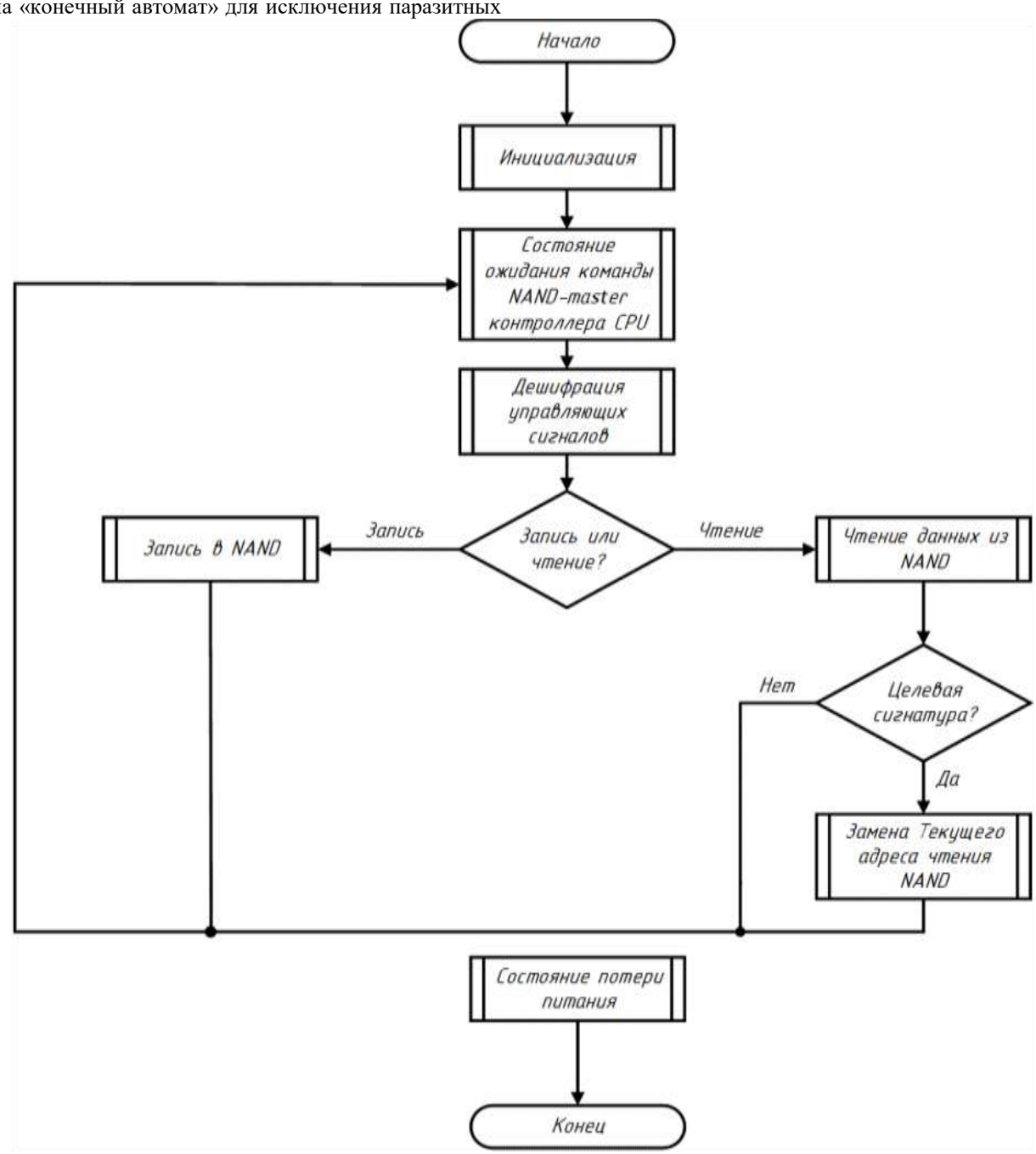

Рис. 3. Концептуальная схема алгоритма работы устройства

III. Практическое применение устройства динамической модификации ВПО на базе ПЛИС

A. Требования к ПЛИС и коммутационной плате

Для получения устройства на базе программируемой логической интегральной микросхемы, содержащей минимальное число паразитных компонентов после описания и синтеза проекта, необходимо использовать ПЛИС FPGA, а не CPLD. Также невозможно в точности рассчитать заранее, сколько понадобится логических ячеек ПЛИС, поэтому стоит начинать подобное проектирование на микросхемах, как минимум, среднего по количеству логических ячеек сегмента [14]. «Пересаживание» микросхемы памяти на макетную плату и реализация электрических линий интерфейса навесным монтажом крайне негативно скажется на передаче сигналов, в силу чего для установки ПЛИС в разрыв между извлеченной из устройства микросхемой памяти и исполняющим компонентом должна быть использована коммутационная плата. Простейший вариант коммутационной платы – из тонкого фольгированного стеклотекстолита, на нижней стороне которого топология, соответствующая посадочному месту микросхемы памяти [15], а на верхней – установлена ПЛИС с самой микросхемой. Плата припаивается нижней стороной к посадочному месту запоминающего компонента, что обеспечивает надежное соединение с минимальными паразитными показателями. При создании топологии платы необходимо учесть требования к минимизации разности длин проводников интерфейса передачи данных [16].

В ходе исследования была разработана печатная коммутационная плата 0,5 мм, а также описан и синтезирован проект динамической модификации данных интерфейса связи на базе модуля ПЛИС FPGA.

B. Описание проекта ПЛИС

Структурное описание проекта на уровне блоков верхнего уровня представлено на рисунке 4. Структура разработанного проекта в виде конечного автомата представлена на рисунке 5.

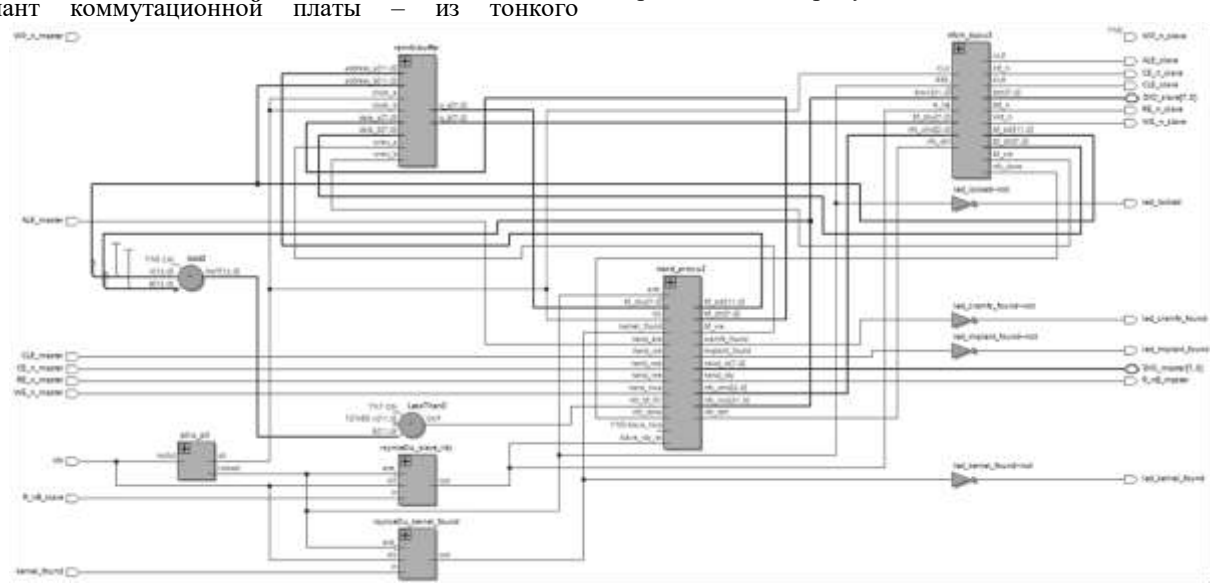

Рис. 4. Структурное описание проекта

Рис. 5. Представление структуры разработанного проекта в виде конечного автомата

Успешное проведение динамической модификации данных обуславливается незначительностью задержки, возникающей при работе программируемой логической интегральной микросхемы. На рисунке 6 показана величина задержки, возникающая при динамической модификации передаваемых по интерфейсу связи данных.

Динамическая модификация производилась в условиях изначальной частоты работы параллельного асинхронного интерфейса передачи данных NAND-микросхемы памяти (ONFI) 20МГц (период 50нс), и т.к. компоненты системы обладают запасом в скорости исполнения, то возникшая от встраивания в интерфейс связи задержка в 9нс не нарушила её работоспособности. В результате можно утверждать, что устройства динамической модификации внутреннего программного обеспечения на базе программируемой логической интегральной микросхемы способны производить операции подмены на шине данных ONFI [17] частотой работы 20МГц,

внося задержку около 9нс, при этом стабильность работы системы тем выше, чем больший запас по увеличению тактовой частоты интерфейса передачи данных между запоминающим и исполняющим компонентом имеет система.

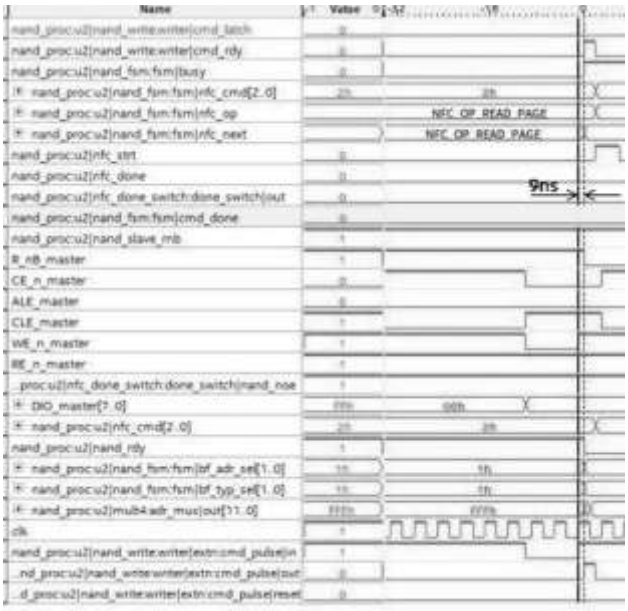

Рис. 6. Показатели задержек на временных диаграммах интерфейса связи при динамической модификации данных на базе ПЛИС FPGA

IV. Заключение

Исходя из полученных в процессе разработки и практического использования устройства: опыта встраивания в исследуемое устройство, данных о вносимой задержке и осциллограмм передач, а также успешной конфигурируемой модификации передаваемых от запоминающего компонента данных, можно сделать вывод о том, что динамический подход к модификации является приоритетным в случаях, когда необходимо модифицировать данные запоминающего компонента не на постоянной основе, «статически», а только в некоторые, заранее известные моменты рабочего цикла исследуемого устройства. Это достигается за счет возможности предварительной конфигурации модификации, со срабатыванием по целевому состоянию шины данных, чего не предусматривает классический подход. Разработанное устройство вносит задержку при передаче данных, что является ограничением при применении к более высокоскоростным интерфейсам, что необходимо оценивать заранее и выбирать более высокоскоростную программируемую логическую интегральную микросхему. В описанном применении задержка не является критической, и не приводит к распознаванию исполняющим компонентом состояния ошибки на шине. Стоит обратить внимание, что встраивание модифицирующего устройства производилось посредством переходной коммутирующей печатной платы материалом основы FR4. Очевидно, для удобства и работы с более высокоскоростными интерфейсами необходимо уменьшать длину проводников и толщину платы, что может быть достигнуто за счет использования гибко-жестких печатных плат.

Благодарности

Авторы статьи выражают благодарность ООО «ИНФОРИОН» за предоставленную при проведении исследования техническую базу.

Поддержка

Ряд результатов настоящего проекта получен при финансовой поддержке Министерства науки и высшего образования Российской Федерации по проекту № 0705-2020-0041 «Фундаментальные исследования методов цифровой трансформации компонентной базы микро- и наносистем».

Литература

[1] Skorobogatov S., Kuhn M. The bumpy road towards iPhone 5c NAND mirroring - Cambridge, University of Cambridge, Computer Laboratory. Available at: https://arxiv.org/abs/1609.04327 (дата обращения: 08.01.2021).

[2] Степченков Д.Ю., Петрухин В.С., Морозов Н.В. Средства системной отладки рекуррентного вычислителя // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС).2014. Сборник трудов / под общ. ред. академика РАН А.Л. Стемпковского. М.: ИППМ РАН, 2014. Часть 2. С. 39-44.

[3] Ненашев О.В. Методы встраивания средств тестирования в устройства с использованием средств автоматизации реинжиниринга // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2014. Сборник трудов / под общ. ред. академика РАН А.Л. Стемпковского. М.: ИППМ РАН, 2014. Часть 2. С. 101-106.

[4] Руткевич А.В., Поляков Е.А., Сысоев И.Ю. СФ-блок контроллера массива NAND Flash-памяти // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2014. Сборник трудов / под общ. ред. академика РАН А.Л. Стемпковского. М.: ИППМ РАН, 2014. Часть 4. С. 7-12.

[5] Потовин Ю.М., Соин С. Разработка быстродействующего блока памяти с ассоциативной выборкой // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2014. Сборник трудов / под общ. ред. академика РАН А.Л. Стемпковского. М.: ИППМ РАН, 2014. Часть 4. С. 29-32.

[6] Власов А.И., Семенцов С.Г., Поляков Ю.А. Микропроцессорные микросистемы активной акустической индивидуальной защиты // Микросистемная техника. 2000. № 2. С. 1-5.

[7] Костиков В.Г., Парфенов Е.М., Шахнов В.А. Источники электропитания электронных средств // Научно-техническое изд-во "Радио и связь". Москва, 1998. 344 с. Схемотехника и конструирование. Учебник для вузов.

[8] URL: https://www.alldatasheet.com/datasheet-pdf/pdf/1132540/HYNIX/H27UCG8T2BTR-BC.html (дата обращения: 08.01.2021)

[9] URL: http://www.onfi.org/specifications (дата обращения: 10.01.2021)

[10] URL: https://www.iwavejapan.co.jp/product/iW%20pdf%20files/iW-nand-flash-controller-datasheet-R1.0.pdf (дата обращения: 10.01.2021)
[11] Шахнов В.А., Власов А.И., Поляков Ю.А., Кузнецов А.С. Нейрокомпьютеры: архитектура и схемотехника // Изд-во Машиностроение. Москва, 2000. 64 с. Сер. Приложение к журналу "Информационные технологии". №9.
[12] Шахнов В.А., Власов А.И., Кузнецов А.С., Поляков Ю.А. Нейрокомпьютеры - Архитектура и Реализация. Ч. 3.1. Аппаратная реализация нейровычислителей // Chip news: Инженерная микроэлектроника. 2000. № 8. С. 12-18.
[13] Соловьев В.В. Проектирование на программируемых логических интегральных схемах быстрых конечных автоматов // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2016. Вып. 1. С. 24-31.
[14] Sabirov T.R., Vlasov A.I., Mirzamagomedov Y.I. Microstrip antenna array for the onboard radio system of small spacecraft // Pleiades Publishing Ltd, Solar system research. 2013. №7, pp. 569-576.
[15] Штучный А.М., Курейчик В.М. Обратная разработка печатных плат с использованием алгоритма обезьян // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2020. Вып. 3. С. 120-125. doi:10.31114/2078-7707-2020-3-120-125
[16] Романов Ф.И., Шахнов В.А. Конструкционные системы микро-и персональных ЭВМ // Изд-во "Высшая Школа". Москва, 1991. 272 с. Практическое пособие.
[17] Руткевич А.В., Воронков Д.И., Сысоев И.Ю., Хайло Н.Н., Вейков А.А. Опыт разработки радиационно-стойкого контроллера накопителя для бортовой космической аппаратуры // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2018. Вып. 2. С. 162-169. doi:10.31114/2078-7707-2018-2-162-169

Dynamic modification of embedded devices internal firmware for solve reverse engineering problems

A.A. Gladkikh, A.I. Vlasov, D.A. Uzenkov, T.M. Fatkhutdinov

BMSTU, Moscow, vlasov@iu4.ru

***Abstract* — The paper discusses embedded devices firmware dynamic modification in reverse engineering. The features of storing embedded devices internal firmware are briefly considered. Usage of dynamic data modification in solving various problems of electronic devices reverse engineering is proposed. The described method has advantage over static modification methods in terms of implementation speed and modification data correction, which is stored in the device ROM. The main attention is paid to the dynamic data modification device architecture with parallel asynchronous interface. Possibilities of using programmable logic integrated circuits for dynamic modification of data storage devices and solving problems of reverse engineering are shown. The main problem of the methods for modifying electronic devices internal firmware is that they do not suppose data changes in the device storage component during its operation, i.e. are static. This is a consequence of using approach of "direct" data modification, which implies real change in the amount of energy stored in memory cell. The authors proposed a strategy of "dynamic approach using programmable logic integrated circuit", which allows to eliminate the above problem. The structure of the system using the proposed strategy of dynamic modification of the system data based on the programmable logic integrated circuit allows modifying the data of the storage device transmitted via the communication interface from the storage device to the executing component without physically interfering with the state of the permanently stored data of the internal software and, as a consequence, using various modification configurations during one cycle of operation of the back-developed device. The proposed method for dynamic modification of internal software data is implemented in a hardware-software complex embedded in the gap between the read-only memory and the executing components of the device. It allows you to transfer pre-prepared data in lieu of the actual response of the storage device. The target response message of the storage device is determined in binary form during preliminary configuration. The method can be used to solve various problems when debugging software algorithms of embedded electronic devices in dynamic mode.**

***Keywords* — data modification, dynamic data modification, dynamic debug, reverse engineering, programmable logic integration circuit, embedded devices, embedded systems, embedded systems research, embedded devices internal firmware modifications, embedded devices internal firmware patch, memory modifications in non-volatile memory devices.**

ASSISTANCE

Some results of the project were obtained with the financial support of the Ministry of Science and Higher Education of the Russian Federation for the project No. 0705-2020-0041 “Fundamental research of methods of digital transformation of the component base of micro- and nano-systems”.

REFERENCES

[1] Skorobogatov S., Kuhn M. The bumpy road towards iPhone 5c NAND mirroring - Cambridge, University of Cambridge, Computer Laboratory. Available at: https://arxiv.org/abs/1609.04327 (accessed 08.01.2021).
[2] Stepchenkov D.Yu., Petrukhin V.S., Morozov N.V. System Debugging Tools for Recurrent Computing Device // Problems of Perspective Micro- and Nanoelectronic

Systems Development - 2014. Proceedings / edited by A. Stempkovsky, Moscow, IPPM RAS, 2014. Part 2. P. 39-44.
[3] Nenashev O.V. An approach to hardware test point insertion automation based on hardware reengineering tools // Problems of Perspective Micro- and Nanoelectronic Systems Development - 2014. Proceedings / edited by A. Stempkovsky, Moscow, IPPM RAS, 2014. Part 2. P. 101-106.
[4] Rutkevich A.V., Polyakov E.A., Sysoev I.Y. NAND Flash memory controller IP-core // Problems of Perspective Micro- and Nanoelectronic Systems Development - 2014. Proceedings / edited by A. Stempkovsky, Moscow, IPPM RAS, 2014. Part 4. P. 7-12.
[5] Potovin Y.M., Soin S. High-speed content addressable memory block design // Problems of Perspective Micro- and Nanoelectronic Systems Development - 2014. Proceedings / edited by A. Stempkovsky, Moscow, IPPM RAS, 2014. Part 4. P. 29-32.
[6] Vlasov A.I., Sementsov S.G., Polyakov Yu.A. Microprocessor-based microsystems for active acoustic personal protection. Mikrosistemnaya tehnika, 2000, no.2, pp. 1-5 (in Russian).
[7] Kostikov V.G., Parfenov E.M., Shahnov V.A. Istochniki elektropitaniya elektronnyih sredstv. Shemotehnika i konstruirovanie. Uchebnik dlya vuzov. - Sources of power supply for electronic devices. Circuitry and design. Textbook for universities., Moscow, "Radio i svyaz" Publ., 1998. 344 p. (In Russian).
[8] URL: https://www.alldatasheet.com/datasheet-pdf/pdf/1132540/HYNIX/H27UCG8T2BTR-BC.html (accessed 08.01.2021)
[9] URL: http://www.onfi.org/specifications (accessed 10.01.2021)
[10] URL: https://www.iwavejapan.co.jp/product/iW%20pdf%20files/iW-nand-flash-controller-datasheet-R1.0.pdf (accessed 10.01.2021)
[11] Shahnov V.A., Vlasov A.I., Polyakov Yu.A., Kuznetsov A.S. Neyrokompyuteryi: arhitektura i shemotehnika - Neurocomputers: architecture and circuitry, Moscow, Informatsionnyie tehnologii №9, Mashinostroenie Publ., 2000. 64 p. (In Russian).
[12] Shahnov V.A., Vlasov A.I., Kuznecov A.S., Polyakov Yu.A. Neurocomputers - Architecture and Implementation. P. 3.1. Hardware implementation of neurocomputers. Chip news: Inzhenernaya mikroelektronika, 2000, no.8, pp. 12-18 (in Russian).
[13] Salauyou V.V. Designing on FPGA of high-speed finite state machines // Problems of Perspective Micro- and Nanoelectronic Systems Development - 2016. Proceedings / edited by A. Stempkovsky, Moscow, IPPM RAS, 2016. Part 1. P. 24-31.
[14] Sabirov T.R., Vlasov A.I., Mirzamagomedov Y.I. Microstrip antenna array for the onboard radio system of small spacecraft // Pleiades Publishing Ltd, Solar system research. 2013. №7, pp. 569-576.
[15] Shtuchnyy A.M., Kurejchik V.M. Reverse PCB development using the monkey algorithm // Problems of Perspective Micro- and Nanoelectronic Systems Development - 2020. Issue 3. P. 120-125. doi:10.31114/2078-7707-2020-3-120-125
[16] Romanov F.I., Shahnov V.A. Konstruktsionnyie sistemyi mikro-i personalnyih EVM. Prakticheskoe posobie. - Structural systems of micro and personal computers. A practical guide., Moscow, "Vyisshaya Shkola" Publ., 1991. 272 p. (In Russian).
[17] Rutkevich A.V., Voronkov D.I., Sysoev I.Y., Khaylo N.N., Veykov A.A. The Experience of Universal Controller SSD Development for Space Application // Problems of Perspective Micro- and Nanoelectronic Systems Development - 2018. Issue 2. P. 162-169. doi:10.31114/2078-7707-2018-2-162-169