of new FPGAs offers opportunities for partial reconfigurations. A change only of the part (that is represented by the particular SM-component) of a control program requires data exchange only in a small part of the considered FPGA. The reconfiguration is carried out with the small differential configuration bit-stream.

References: **1.** *M.Adamski*, "Parallel Controller Implementation using Standard PLD Software", in: W.R.Moore, W.Luk (Eds), *FPGAs*, *The Oxford 1991 International Workshop on Field Programmable Logic and Applications*, Abingdon EE&CS, Abingdon (UK), 1991, pp.296-304. **2.** *M.Adamski, M.Wegrzyn,* "Hierarchically Structured Coloured Petri Net Specification and Validation of Concurrent Controllers", *Proceedings of the 39th International Scientific Colloquium, IWK'94*, Ilmenau (Germany), 27-30.09.1994, Band 1, pp.517-522. **3.** *M.Adamski, M.Wegrzyn*, "Field Programmable Implementation of Concurrent State Machine", *The Third International Conference on Computer-Aided Design of Discrete Devices, CAD DD'99*, 10-12.11.1999, Minsk, Belarus, Vol.1, pp.4-12. **4.** *A.Barkalov, M.Wegrzyn*, *Design of Control Units with Programmable Logic*, University of Zielona Gyra Press, Zielona Gyra, 2006. **5.** *H.Belhadj, L.Gerbaux, M.-C.Bertrand, G.Saucier,* "Specification and Synthesis of Communicating Finite State machines", *Synthesis for Control Dominated Circuits*, (A-22), Elsevier Science Publishers B.V. (North Holland), 1993, pp.91-101. **6.** S.Chmielewski, M.Wкgrzyn, "Modelling and synthesis of automata in HDLs", *Proceedings of SPIE (XVIII IEEE-SPIE Symposium on Photonics, Electronics and Web Engineering),* Wilga (Poland), 29.05-04.06.2006, (to be published). **7.** *R.David, H.Alla, Petri Nets and Grafcet*, Prentice Hall, New York, 1992. **8.** *A.Wegrzyn, Symbolic Analysis of Logical Control Devices using Selected Methods of Petri Net Analysis*, Ph.D. Thesis, Warsaw Univ. of Technology, 2003, (in Polish). **9.** *A.Wegrzyn, M.Wegrzyn*, "Petri Net Decomposition Algorithm based on Finding Deadlocks and Traps", *IEEE East-West Design & Test International Workshop*, Sochi (Russia), 15-19.09.2006. **10.** *M.Wegrzyn, P.Wolacski, M.A.Adamski, J.L.Monteiro*, "Field Programmable Device as a Logic Controller", *Proceedings of the 2nd Conference on Automatic Control - Control'96*, Oporto (Portugal), 11-13.09.1996, Vol.2, pp.715-720. **11.** *M.Wegrzyn*, *Hierarchical implementation of Logic controllers by means of Petri nets and FPGAs*, Ph.D. Thesis, Warsaw Univ. of Technology, 1999, (in Polish).

Поступила в редколлегию 27.07.2006

Рецензент: д-р техн. наук, проф. Хаханов В.И.

Wegrzyn Agnieszka, PhD., lecturer at University of Zielona Gora. Interested in: formal analysis, Petri net, databases, DBMS, Internet applications. Address: Computer Eng. and Electronics Institute, University of Zielona Gora, ul. Podgorna 50, 65-246 Zielona Gora, Poland, E-mail: A.Wegrzyn@iie.uz.zgora.pl Ph.: (+48 68) 3282 484.

Marek Wegrzyn, PhD., lecturer at University of Zielona Gora. Interested in: formal analysis, Petri net, databases, DBMS, Internet applications. Address: Computer Eng. and Electronics Institute, University of Zielona Gora, ul. Podgorna 50, 65-246 Zielona Gora, Poland, E-mail: A.Wegrzyn@iie.uz.zgora.pl Ph.: (+48 68) 3282 484.

УДК 638.235.231

СУБЪЕКТНО-ОБЪЕКТНАЯ МОДЕЛЬ НЕСАНКЦИОНИРОВАННОГО ДОСТУПА С ИСПОЛЬЗОВАНИЕМ АППАРАТНЫХ ЗАКЛАДОК К КОМПЬЮТЕРНОЙ ИНФОРМАЦИИ

ГОРБАЧЕВ В.А., СТЕПАНЕНКО В.В., ИВАНИСЕНКО И.Н.

Использован субъектно-объектного подхода при разработке моделей функционирования аппаратной закладки. Полученные результати могут быть использованы при построении формальных моделей политики безопасности КС.

Целью работы является анализ угроз информации, исходящих от аппаратных ресурсов компьютерных систем. Для достижения этой цели решаются следующие *задачи*: разработка математической модели несанкционированного доступа, который реализуется аппаратными закладками, а также определение функции управления доступом к аппаратным ресурсам.

Для построения модели взаимодействия электронных компонентов компьютерной системы (КС), среди которых имеются аппаратные закладки, применяется субъектно-объектный подход. В рамках данного подхода все множество электронных компонентов представляется в виде двух подмножеств: субъектов и объектов. Разделение на соответствующие подмножества осуществляется по признаку активности каждого компонента. Полученные результаты могут быть использованы при построении формальных моделей политики безопасности КС.

Анализ литературы по защите информации в КС [1-5] позволяет утверждать, что в настоящее время сам компьютер, его аппаратные ресурсы (электронные компоненты) могут содержать различные источники угроз безопасности информации, расположенные как в отдельном компьютере, так и в КС в целом. Это обстоятельство обусловлено рядом причин. Во-первых, современная технология позволяет разместить на достаточно малых площадях, которые составляют 5-10% от размеров всего кристалла интегральной схемы, устройства, способные выполнять широкий спектр не специфицированных функций. Во-вторых, информация в КС рано или поздно появляется в открытом виде, и аппаратные ресурсы могут получить доступ к этой информации. В-третьих, высокая степень интеграции электронных компонентов и технологии, применяемые для их производства, усложняют работу по их сертификации, а также по наблюдению за их функционированием.

В настоящее время активно исследуется проблема несанкционированного доступа, который реализуется программными закладками (ПЗ) [1,2,4, 5]. На практике можно внедрить ПЗ в ядро операционной системы или в любое приложение таким образом, что она будет как контролировать и модифицировать всю деятельность системы безопасности, так и осуществлять несан-

кционированный доступ. Программы такого типа можно найти в Интернете. Они называются руткиты [4].

Литература, в которой рассматривались бы угрозы безопасности информации в КС от электронных компонентов, внедрённых в структуру КС и не использующих каналы ПЭМИН, практически отсутствует. Как следствие, отсутствуют методы защиты информации от данного типа угроз. Это позволяет утверждать, что проблема угроз, реализуемых аппаратными ресурсами КС, является актуальной.

В данной работе проблема угроз информации, исходящих от аппаратных ресурсов, будет рассмотрена для сложных электронных систем, которыми являются КС, современные коммуникационные системы специального назначения и другие.

Определим аппаратную закладку (АЗ) как электронный компонент в интегральном исполнении, встроенный в интегральную схему при её проектировании и производстве, или выполненный в виде отдельной интегральной схемы и создающий угрозу безопасности информации своими не специфицированными функциями.

В основу разработки формальной модели несанкционированного доступа, реализуемого АЗ, могут быть положены, во-первых, понятия: объект, субъект и операция доступа для пары «объект-субъект»; во-вторых, аксиома [3]: все вопросы безопасности информации описываются доступами субъектов к объектам.

Упомянутым выше абстрактным понятиям поставим в соответствие такие физические представления в среде КС:

Объект (O_j) – часть ресурсов системы, находящаяся в дискретный момент времени в пассивном состоянии относительно информации, а также других аппаратных элементов этой системы.

Субъект (S_i) – электронный компонент, находящийся в дискретный момент времени в активном состоянии, другими словами, способный осуществить доступ к объекту. Будем рассматривать такой компонент, как пару: ресурс и доступ.

Доступ – категория субъектно-объектной модели, описывающая процесс выполнения операций субъектов над объектами. Множество возможных операций субъектов над объектами в системе будем обозначать через **X**. Очевидно, если операция доступа реализуется АЗ, то она должна иметь аппаратно-программную поддержку в виде некоторого канала.

Следует отметить, что практически все аппаратные компоненты КС в различные моменты времени могут представлять собой либо субъекты, либо объекты. Данная особенность обусловлена тем фактом, что практически все элементы КС представляют собой микропроцессорные устройства, функциями которых является получение, хранение, обработка и передача информации. В первых двух случаях устройство будет выполнять роль объекта, в остальных случаях то же самое устройство будет выполнять роль субъекта.

Рассматривая АЗ как тип субъекта, который способен осуществить НСД, отметим следующие ее особенности.

Во-первых, АЗ, являясь, программно-аппаратными ресурсами КС, имеет одну и ту же среду существования, что и объекты. Более того, в пассивном состоянии АЗ может быть частью некоторого объекта.

Во-вторых, угрозы информации в КС могут исходить только от активного субъекта - АЗ, который в текущий момент времени владеет функцией управления ресурсами и, в общем случае, способный изменять состояния объектов.

В-третьих, субъекты могут влиять друг на друга через изменяемые ими объекты. Одним из результатов воздействия субъекта на объект могут быть порожденные в КС другие субъекты (или состояния системы), которые представляют угрозу для безопасности информации. В данном случае речь идет об инициализации АЗ и переводе ее в активное состояние.

Пары (S_i, O_j) связываются множеством разрешенных, с точки зрения системы защиты информации (ЗИ), операций $\mathbf{X}_R$. Это множество определяется политикой безопасности (ПБ) и является подмножеством всего множества **X** возможных операций для каждой пары. В то же время пары (S_i, O_j) могут связываться множеством запрещенных, с точки зрения ПБ, операций $\mathbf{X}_B$. Очевидно, что $\mathbf{X} = \mathbf{X}_R \bigcup \mathbf{X}_B$. Задачей ПБ является контроль и блокирование выполнения операций из множества $\mathbf{X}_B$.

Если учесть предположение о том, что АЗ может воспользоваться штатным каналом КС, т.е. каналом, который поддерживает операцию доступа из множества X_R, то $X_R \bigcap X_B = \varnothing$. Таким образом, становится очевидным, что разрешения либо запрещения самого факта доступа для обеспечения заданной политики безопасности – недостаточно.

Рассмотрим понятие пользователя (злоумышленника) в рамках объектно-субъектного подхода, сформулируем важное свойство пользователя (злоумышленника) в виде следующей аксиомы.

Аксиома. Пользователь (злоумышленник) воспринимает объекты и получает информацию о состоянии КС через субъекты (АЗ), которые он может активизировать. Таким образом, злоумышленник для осуществления несанкционированного доступа к информации должен перевести некоторый объект системы в активное состояние.

Обобщим понятие субъекта КС, определенное выше, на пользователя (злоумышленника) следующим образом. Пользователь – физическое лицо, аутентифицируемое некоторой информацией и управляющее субъектами КС через органы управления КС. Пользователь (злоумышленник) КС является, таким обра-

зом, внешним субъектом или субъектом внешней среды КС.

Анализ свойств субъектов, таких как АЗ и злоумышленника, позволяет сделать следующие выводы, связанные с реализацией модели безопасности КС: необходимость учета возможностей субъектов изменять свойства и архитектуру КС; в любой дискретный момент времени должна быть известна и фиксирована декомпозиция КС на субъекты и объекты; в любой дискретный момент времени множество субъектов КС не пусто (в противном случае соответствующие моменты времени исключаются из рассмотрения и рассматриваются отрезки с ненулевой мощностью множества субъектов).

Понятие доступа является одним из основополагающих в теории защиты информации, поскольку разрешение или запрет доступа для заданных множеств субъектов и объектов в конечном итоге определяет безопасность КС.

Формализуем операцию доступа субъекта к объектам. В первую очередь специфицируем механизм доступа в целях порождения новых субъектов. Работу данного механизма можно продемонстрировать на следующем примере. Допустим, некоторому процессу, протекающему в КС, необходимо прочитать определённые данные с накопителя на жёстком диске (HDD). Для этого процесс в некоторый момент времени t обращается к контролеру жесткого диска (IDE controller) с соответствующим запросом. В этот момент времени процесс, инициирующий получение данных, является субъектом, а контроллер жёсткого диска – объектом. При этом субъект изменяет содержимое внутренних регистров объекта, тем самым изменяя его свойства и порождая новый субъект в КС. В следующий момент времени (t+1) порождённый субъект обращается к следующему объекту (непосредственно к контроллеру, управляющему механикой жёсткого диска) и передаёт ему соответствующие запросы, изменяя его структуру и тем самым порождая новый субъект и т.д.

Для формализации механизма порождения новых субъектов воспользуемся следующими определениями [3].

Определение 1. Объект O_i называется источником для субъекта S_k, если существует субъект S_j, в результате воздействия которого на объект O_i в компьютерной системе возникает субъект S_k.

Определение 2. Субъект S_j, порождающий новый субъект из объекта O_i, в свою очередь, называется активизирующим субъектом для субъекта S_k (S_k назовем порожденным объектом).

Теперь введем обозначение: $\mathrm{Create}(S_j,O_i,P') \rightarrow S_k$, которое означает, что из объекта O_i порожден субъект S_k при воздействии субъекта S_j с помощью активизирующего потока P'. *Create* назовем операцией порождения субъектов.

Операция *Create* задает отображение декартова произведения множеств субъектов и объектов на объединение множества субъектов с пустым множеством. Интерпретация операции *Create* на физическом уровне означает, что субъект S_j, обладая управлением и ресурсами, может передать S_k часть ресурсов и функцию управления. Заметим также, что в КС действует дискретное время, и фактически новый субъект S_k порождается в момент времени t+1 относительно момента t, в который произошло воздействие порождающего субъекта на объект - источник. Очевидно, что операция порождения субъектов зависит как от свойств активизирующего субъекта, так и от содержания объекта-источника.

Считаем, что если $\mathrm{Create}(S_j,O_i,P') \rightarrow \mathrm{NULL}$ (конструкция NULL далее обозначает пустое множество), то порождение нового субъекта из объекта O_i при активизирующем воздействии S_j невозможно.

Теперь формализуем механизм доступа в целях чтения или записи данных в объект, т.е. в целях изменения объекта. Свойство АЗ быть активной не только в выполнении действий над объектом для создания нового субъекта, но и для чтения или записи данных в объект, реализует такой механизм несанкционированного доступа.

В общем случае необходимо отметить, что поскольку объекты КС по определению являются пассивными, то для выполнения аппаратной закладкой всех описанных выше действий необходимо существование потоков информации от субъекта к объекту (в противном случае невозможно говорить об изменении объектов). Так как данный поток инициируется и реализуется субъектом - АЗ, это означает, что операция порождения потока локализована в субъекте - АЗ и отображается состоянием функционально ассоциированных с ним объектов. Таким образом, поток информации между объектом O_m и объектом O_j всегда должен быть связан с некоторой операцией над объектом O_m, реализуемой субъектом S_j и зависящей от O_j.

Порождение потока информации P'' от объекта O_m к объекту O_j обозначим как $\mathrm{Stream}(S_i, O_m,P'') \rightarrow O_j$. При этом будем выделять объект-источник O_m и объект-получатель (приемник) потока O_j. Из данного определения также следует, что поток всегда инициируется (порождается) субъектом, а также что в частном случае операция Stream может создавать новый объект или уничтожать его.

В общем можно сказать, что операция Create является частным случаем операции Stream, поскольку операция Create, также как и Stream, создаёт поток, однако при этом происходит порождение нового субъекта в КС. Анализ механизмов порождения потоков и субъектов позволяет сделать следующий вывод: с формальной точки зрения операция доступа субъекта S_j может быть представлена обобщенными операциями O_1 и O_2:

$$O_1 : \mathrm{Create}(S_j,O_i, P') \rightarrow S_i ,$$

$$O_2 : \text{Stream}(S_i, O_m, P^{"}) \rightarrow O_j.$$

Рассматривая в качестве субъектов и объектов доступа ресурсы аппаратных систем, можно выделить основные элементарные операции, при помощи которых реализуются все функции аппаратных систем: x_1 – инициализация команды; x_2 – чтение информации из объекта; x_3 – запись информации в объект; x_4 – чтение информации из подключённой внешней шины данных; x_5 – модификация транзитно передаваемой информации.

Приведенные элементарные операции являются частным случаем обобщенных операций O_1 и O_2.

Введем функцию управления доступом F для заданных пар субъектов-объектов, аргументами которой являются операции O_1 и O_2, а также поток P и время t осуществления доступа: $F = F(O_1, O_2)$.

Найдем область определения функции, исходя из следующих соображений: аргументы O_1 и O_2 должны подтверждать факты выполнения операций Create и Stream соответственно своими значениями, равными 1, и факты невыполнения этих операций своими значениями, равными 0. Конкретные значения этих аргументов определяются технической документацией на изделие. Значениями аргумента P может быть множество легализованных команд, например, на шине управления.

Очевидно, что функция управления доступа может быть задана в любом виде: либо в виде таблицы, либо в виде алгоритма. Она может принимать значения "1", если операция доступа разрешена, и "0", если операция доступа запрещена.

Научная новизна: в данной работе предложена субъектно-объектная модель операции несанкционированного доступа и функция управления им. Полученные результаты могут быть использованы при разработке модели безопасности КС.

Практическая значимость и новизна полученных результатов состоит в том, что определена процедура взаимодействия штатных электронных компонентов компьютерной системы с интегрированными в ее структуру аппаратными закладками, а также требуемые для этих целей аппаратные ресурсы.

Литература: 1. *Анин Б.* Защита компьютерной информации. СПб.: BHV Санкт Петербург, 2000. VIII. 368с. **2.** *Барсуков В.С., Водолажский В.В.* Современные технологии безопасности. М.: Нолидж, 2000. 496 с.**3.** *Грушо А.А., Тимонина Е.Е.* Теоретические основы защиты информации. М.: Изд-во агентства "Яхтсмен", 1996. 276 с. **4.** *Robert S. Morris,* Sr. "UNIX Operating System Security", BSTJ, Vol. 62, No. 8, 1984 Bell Systems Technical Journal. **5.** Jamie Butler and Greg Hoglund: Rootkits: Subverting the Windows Kernel. Addison Wesley, 2005. ISBN 0-321-29431-9.

Поступила в редколлегию 06.09.2006

Рецензент: д-р физ.-мат. наук, проф. Смеляков С.В.

Горбачев Валерий Александрович, профессор кафедры ЭВМ ХНУРЭ. Научные интересы: моделирование систем. Хобби: музыка, волейбол, автомобили. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. +38(057)7021-354.

Степаненко Владимир Викторович. Научные интересы: моделирование систем, компьтерные системы и сети, модели систем хащиты информации . Хобби: разаработка микропроцессорных систем управления, баскетбол, автомобили, туризм. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. +38(067)710-4188.

Иванисенко Игорь Николаевич, ассистент кафедры ЭВМ ХНУРЭ. Научные интересы: моделирование систем, компьтерные сети, модели планирования нагрузки в распределенных вычислительных системах . Хобби: компьютеры, футбол, шахматы, туризм, рыбалка. Адрес: Украина, 61166, Харьков, пр.Ленина, 14, тел. +38(057)7021-427.

УДК 681.3:621.3.013

МЕТОДИКА ОПТИМАЛЬНОГО ПРОЕКТИРОВАНИЯ АНАЛОГО-ЦИФРОВЫХ СХЕМ БИОМЕДИЦИНСКИХ УСТРОЙСТВ С УЧЕТОМ ЭЛЕКТРОМАГНИТНОЙ СОВМЕСТИМОСТИ

ПРАСОЛ И.В., КОБЫЛИНСКИЙ А.В.

Рассматриваются проблемы оптимального проектирования аналого-цифровых схем, в частности схем биомедицинских устройств. Предлагается методика решения подобных задач, использующая многокритериальный подход. Особое внимание уделяется вопросам электромагнитной совместимости (ЭМС) аналого-цифровых схем. Осуществлено ранжирование существующих рекомендаций по ЭМС.

1. Введение

Большинство современных электронных устройств являются аналого-цифровыми. Особенно это касается приборов биомедицинского назначения. Для них характерно наличие разнородных частей: аналоговых датчиков, аналого-цифровых и цифро-аналоговых преобразователей, устройств цифровой обработки сигналов и т.п. *Актуальность* задач оптимального проектирования таких схем обусловлена тем, что в настоящее время не существует какой-либо четкой методики их решения. Имеющиеся подходы к решению подобных задач, как правило, предполагают синтез схем таких устройств по частям, отдельно аналоговых блоков, отдельно цифровых. Однако вследствие взаимного влияния различных частей проектируемого устройства не всегда обеспечивается удовлетворительная работа всей схемы в целом [1].