УДК 004.382

В.С. ГЛУХОВ, Р. ИЛЬЯС

Национальный университет «Львівська політехніка», Україна

КОДИРОВАНИЕ СОСТОЯНИЙ УПРАВЛЯЮЩИХ АВТОМАТОВ В ГАРАНТОСПОСОБНЫХ СИСТЕМАХ

В статье обосновывается необходимость специальных методов перехода от абстрактных к структурным управляющим автоматам, работающим в составе гарантоспособных систем. Предлагается использовать унитарное кодирование состояний автоматов, а также осуществлять преобразование алгоритмов работы автоматов, состоящее в замене состояний ожидания парами дублирующих друг друга взаимосвязанных состояний.

Ключевые слова: *гарантоспособная система, защита информации, шифропроцессор, цифровой автомат, кодирование состояний, унитарное кодирование.*

Введение

Одной из составляющих гарантоспособных систем является их конфиденциальность. Подразумевается, что гарантосопособная система должна обеспечить защиту от несанкционированного использования информации, от подмены информации, от повреждения информации. Конфиденциальность обеспечивается специальными устройствами – шифропроцессорами, которые и осуществляют защиту информации в гарантоспособных системах.

На современном этапе математической основой для построения устройств защиты информации являются поля Галуа и эллиптические кривые.

Устройства защиты информации – шифропроцессоры, как и все процессоры, состоят из двух цифровых автоматов: операционного устройства (операционного автомата) и устройства управления (управляющего автомата). И сам шифропроцессор, и его составные части, а также каналы связи между частями должны быть защищены от различных атак, целью которых является несанкционированное получение или обрабатываемых данных или секретного ключа. А именно, должна быть защита от простых и дифференциальных атак, как на основе потребляемой мощности, так и на основе электромагнитного излучения, от анализа тепловых режимов и от других типов атак.

Целью простых атак является определение секретного ключа, целью дифференциальных атак является определение данных, которые обрабатываются. Одним из методов защиты является маскировка. В статье представлен анализ методов маскировки работы управляющих автоматов, описаны предлагаемые подходы к ее обеспечению.

1. Постановка проблемы

Конфиденциальность гарантоспособных систем обеспечивается специальными устройствами – шифропроцессорами, которые осуществляют защиту информации.

Шифропроцессоры и их составные части, а также каналы связи между частями, должны быть защищены от различных атак, целью которых является несанкционированное получение или обрабатываемых данных или секретного ключа [3 – 4].

И атаки на основе потребляемой мощности, и атаки на основе электромагнитного излучения основываются на разном потреблении КМОП-транзисторных схем в зависимости от их количества и частоты их переключения.

Если при некоторых алгоритмах количество операций над двоичными разрядами при нулевом значении некоторого разряда операнда меньше количества операций при его единичном значении, то значение этого разряда можно определить путем измерения потребляемой мощности или электромагнитного излучения.

Рис. 1 содержит пример измерения потребляемой мощности во время выполнения операций над точками эллиптической кривой:

S – операция сложения точек, D – операция удвоения точек, единицы на осях – условные. Известная последовательность операций позволяет определить разряды ключа [3].

Таким образом, актуальной является задача выравнивания количества элементов цифровых схем, которые меняют свое состояние в различные моменты времени. Далее этот процесс выравнивания называется маскировкой.

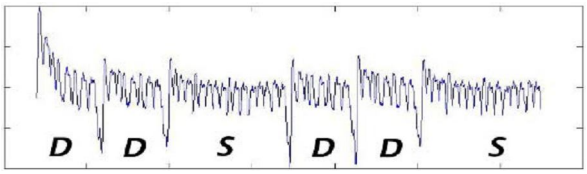

Рис. 1. Изменение потребляемой мощности

2. Анализ последних исследований и публикаций

На современном этапе математической основой для построения устройств защиты информации являются поля Галуа и эллиптические кривые [2].

Некоторые способы атак на шифропроцессоры рассмотрены в [3 – 4]. Там же рассмотрена защита от простых и дифференциальных атак, как на основе потребляемой мощности, так и на основе электромагнитного излучения, от анализа тепловых режимов и от других типов атак.

Целью простых атак является определение секретного ключа, целью дифференциальных атак является определение данных, которые обрабатываются.

В [5] описан метод защиты от атак, при котором в паузах шифропроцесором осуществляется фоновое выполнение операций со случайными или псевдослучайными операндами.

Целью статьи является анализ проблемы маскировки работы управляющих автоматов, разработка методов ее решения.

3. Методы маскировки

Для маскировки работы операционных узлов в основном используют выравнивание длительности выполнения операций в зависимости от значения отдельных разрядов операндов (рис. 2) [3], фоновое выполнение операций со случайными или псевдослучайными операндами (в [5] приведен пример умножителя, который в промежутках между необходимыми операциями выполняет умножение произвольных кодов).

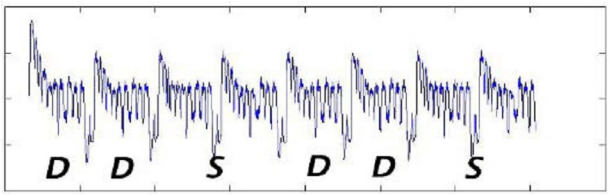

Рис. 2. Изменение потребляемой мощности выровненных операций

4. Маскировка управляющих автоматов

Шифропроцесор, как и любой другой процессор, можно представить как структуру, состоящую из двух цифровых автоматов: операционного и управляющего (рис. 3). Реально шифропроцессор представляет собой множество взаимодействующих между собой цифровых автоматов. Указанное множество образует многоуровневую систему [6].

Способы маскировки работы операционных устройств и каналов передачи данных описаны в литературе (например, [3, 5]). В то же время, методы маскировки управляющих устройств (автоматов) рассмотрены недостаточно. Поэтому ниже предлагается метод такой маскировки, который базируется на представлении управляющих устройств как конечных цифровых автоматов. Предлагается использовать унитарное кодирование состояний автоматов, а также осуществлять определенные преобразования алгоритмов работы автоматов.

Рис. 3. Процессор в составе компьютера

Цифровые автоматы разделяются на абстрактные и структурные.

Абстрактный цифровой автомат A определяется совокупностью пяти объектов {X, S, Y, φ, λ – рис.4}, где

X = $\{x_i\}$ – $i \in \overline{1,m}$ – множество входных сигналов автомата A (входной алфавит автомата A);

S = $\{s_j\}$ – $j \in \overline{1,n}$ – множество состояний автомата A (алфавит состояний автомата A). В общем случае $n_1 \neq n$, практически $n_1 = n$;

Y = $\{y_k\}$ – $k \in \overline{1,l}$ – множество выходных сигналов автомата A (выходной алфавит автомата A);

φ – функция переходов автомата A, которая задает отображение $(X \times S) \to S$;

λ – функция выходов автомата A, которая задает отображение $(X \times S) \to Y$,

КСх – комбинационная схема;

ПА – память автомата.

Одним из способов описания алгоритмов работы автоматов является представление алгоритмов с помощью графов.

При переходе от абстрактных автоматов к структурным для сложных алгоритмов значение n может достигать десятков бит, одновременное их изменение может оказать влияние и на потребление устройства, и на его электромагнитное излучение. Это приводит к необходимости маскирования работы управляющих автоматов.

Необходимость маскировки поясняется следующим примером. Во время взаимодействия двух автоматов (рис. 5) каждый из них может оказаться в состоянии ожидания.

Графы автоматов приведены на рис. 6 и рис. 7 (на графах принято, что автомат остается в некотором состоянии, если условие перехода из этого состояния не выполняется).

Временная диаграмма, которая иллюстрирует изменение состояний автоматов, приведена на рис.8. Видно, что частота изменения состояний (sreg) автоматов меняется. Измерение электромагнитного излучения, которое возникает при неравномерном изменении состояний автоматов, позволяет определить время ожидания и рабочее время, что может облегчить несанкционированную дешифрацию закрытой информации.

Особенно заметным излучение будет в автоматах с большим количеством состояний (с большим значением n).

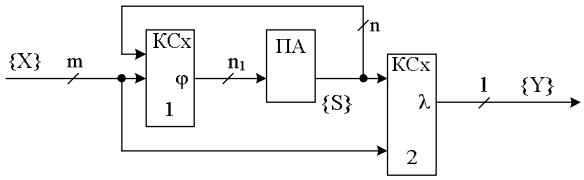

Рис. 4. Конечный цифровой автомат

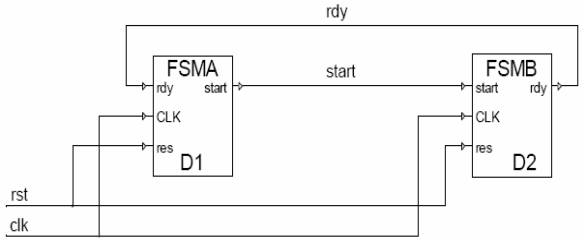

Рис. 5. Взаимодействие двух автоматов (rst – сброс, clk – синхроимпульсы)

Для решения этой проблемы предлагается использовать специальные способы кодировки состояний автоматов, которые дополняются изменением алгоритмов работы (что проявляется в модернизации графов автоматов).

5. Кодирование состояний автомата

Одним из этапов перехода от абстрактного автомата к структурному является кодировка состояний автомата.

Существуют такие основные варианты кодировки состояний автомата: двоичное кодирование; соседнее кодирование; унитарное кодирование.

Примеры их использования для автомата (рис. 6) содержит табл. 1.

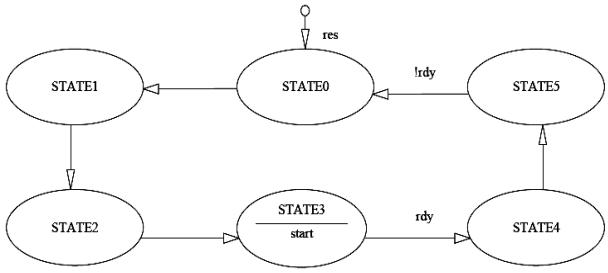

Рис. 6. Граф автомата FSMA (D1)

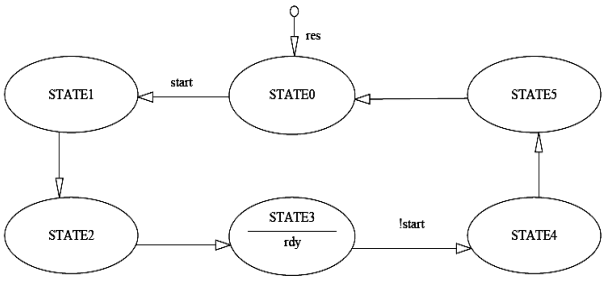

Рис. 7. Граф автомата FSMB (D2)

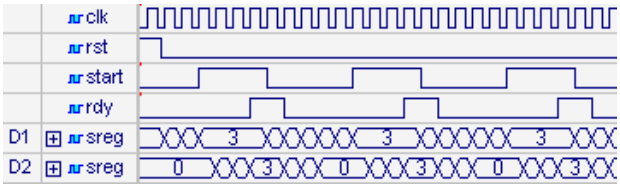

Рис. 8. Временная диаграмма изменения состояний (sreg) двух автоматов

Таблица 1

Варианты кодировки состояний автомата

Кодирование	State0	State1	State2	State3	State4	State5
двоичное	000	001	010	011	100	101
соседнее	000	001	011	111	101	100
унитарное	000001	000010	000100	001000	010000	100000

На практике не всегда удается закодировать все состояния графа автомата соседними кодами. Поэтому, даже при соседнем кодировании могут возникать ситуации, когда при переходе из одного состояния в другое меняется несколько бит кода

состояния вместо одного. В худшем случае могут поменяться сразу все *n* бит (как и в варианте двоичного кодирования).

Сравнение свойств разных способов кодировки с точки зрения маскировки содержит табл. 2, где обозначено:

Таблица 2

Сравнение способов кодировки

Способ	Nmax	Nmin	Nw	Nd
двоичный	*n*	1	0	*n*
соседний	*n*	1	0	*n*
унитарный	2	2	0	2

n – разрядность кода;

Nmax – максимальное количество бит, которые изменяют свое значение при переходе из одного состояния в другой;

Nmin – минимальное количество бит, которые изменяют свое значение при переходе из одного состояния в другой;

Nw – количество бит, которые изменяют свое значение при ожидании (без дублирования);

Nd – количество бит, которые изменяют свое значение при ожидании (с дублированием).

Таким образом, для маскировки лучше всего подходит унитарный способ кодировки состояний автомата. Модифицированный граф автомата FSMA (D1)

При таком способе переход из любого состояния автомату к любому другому всегда вызывает изменение состояния только двух двоичных разрядов, и, соответственно, двух триггеров, которые хранят эти два разряда кода состояния автомата.

6. Модификация графов автоматов

Во время перехода «сам на себя» (ожидание) код состояния автомата не меняется.

Соответственно не меняются состояния триггеров, что резко выделяет такое состояние автомата из всех других.

Поэтому, для маскировки, целесообразно проводить модификацию графов автоматов для исключения состояний ожидания путем их дублирования

Для приведенного выше примера (рис. 5) временная диаграмма взаимодействия модифицированных автоматов приведена на рис. 9, где отличия состояний ожидания от других состояний уже незаметны.

При этом взаимодействие автоматов не поменялось (порядок формирования выходных сигналов обоих автоматов по сравнению с рис. 8 не изменился). Графы автоматов дополняются состояниями, дублирующими состояния ожидания.

7. Реализация модификации графов автоматов

Дублированные состояния ожидания загромождают граф автомата. На следующих этапах проектирования (например, при моделировании или синтезе топологии программированных логических интегральных схем) используются не сами графы, а их описания на специальных языках (VHDL, Verilog). Специализированные графические редакторы, обеспечивающие создание графов автоматов, обеспечивают и их преобразование в VHDL-описание (Verilog-описание).

Поэтому, дублирование состояний можно проводить путем модификации VHDL-описаний, не изменяя граф. Указанные преобразования легко реализуются программным способом.

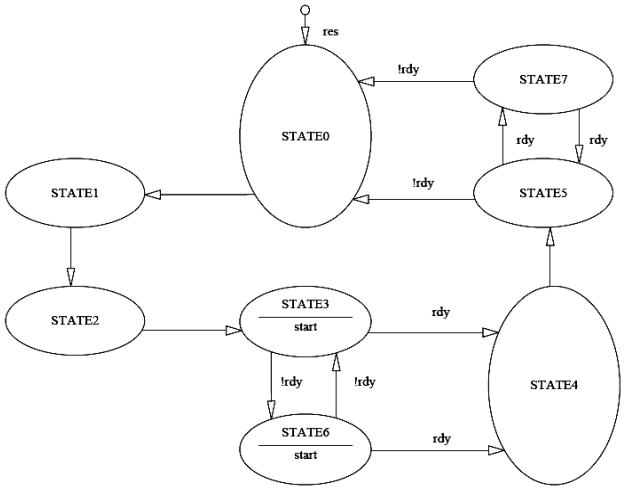

Рис. 1 Модифицированный граф автомата FSMA (D1)

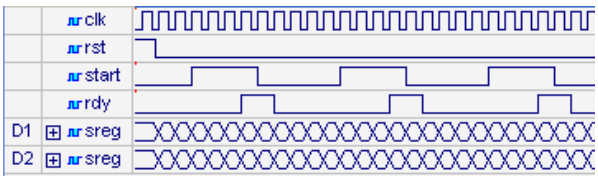

Рис. 9. Временная диаграмма изменения состояний (sreg) двух маскированных автоматов

Выводы

В статье обосновывается необходимость специального кодирования управляющих автоматов, работающих в составе гарантоспособных систем. Разработан метод унитарного кодирования, который предполагает осуществление преобразования алгоритмов работы автоматов, состоящего в замене состояний ожидания парами дублирующих друг друга состояний.

Изменение алгоритмов работы автоматов осуществляется специальной программой, которая преобразовывает VHDL-описания (Verilog-описания) автомата.

Литература

1. Avizienis A. Concepts and Taxonomy of Dependable and Secure Computing / A. Avizienis, J.-C. Laprie, B. Randell, C. Landwehr // IEEE Transactions on Dependable and Secure Computing. – 2004. – Vol. 1. – P. 11-33.

2. IEEE Std 1363-2000 IEEE Standard Specifications for Public-Key Cryptography Sponsor Microprocessor and Microcomputer Standards Committee of the IEEE Computer Society. Approved 30 January 2000.

3. Hankerson D.R. Guide to elliptic curve cryptography / D. Hankerson, A.J. Menezes, S. Vanstone. – New York: Springer-Verlag, 2004. – P. 36-47.

4. Trichina E. Small size, low power, side channel immune AES co-processor: Design and synthesis resultys / E. Trichina, T. Korkishko, K.-H. Lee // In Proc. Of Forth Conf. on Advanced Encryption Standard (AES 2005): Lecture Notes in Computer Science. – 2006. – V. 3373. – P.113-127.

5. Kamala R.V. VLSI Implementation of High Speed Galois Field Modular inversion resistive to Side-Channel Attacks / R.V.Kamala // Center for VLSI and Embedded System Technologies, International Institute of Information Technology, Hyderabad. iiit_kamala_present[1].ppt

6. Глухов В.С. Багаторівнева організація операційного пристрою для роботи з елементами поля Галуа, представленими у нормальній формі/ В.С. Глухов // Збірник матеріалів міжвузівської науково-технічної конференція науково-педагогічних працівників «Проблеми та перспективи розвитку економіки і підприємництва та комп'ютерних технологій в Україні» . – Львів: Ліга-Прес, 2007. – 120 с.

Поступила в редакцию 1.02.2009

Рецензент: д-р техн. наук, проф., проф. кафедры ЭВМ А.А. Мельник, Национальный университет «Львівська політехніка», Львов.

КОДУВАННЯ СТАНІВ КЕРУЮЧИХ АВТОМАТІВ У ГАРАНТОЗДАТНИХ СИСТЕМАХ

В.С. Глухов, Р. Ільяс

У статті обґрунтовується необхідність спеціальних методів переходу від абстрактних до структурних керуючих автоматів, що працюють у складі гарантоздатних систем. Пропонується використовувати унітарне кодування станів автоматів, а також здійснювати перетворення алгоритмів роботи автоматів, що полягає в заміні станів очікування парами дублюючих один одного взаємозв'язаних станів.

Ключові слова: гарантоздатна система, захист інформації, шифропроцесор, цифровий автомат, кодування станів, унітарне кодування.

FINITE STATE MACHINE CODING IN DEPENDABLE SYSTEMS

V.S. Hlukhov, Rodrigue Elias

In the article the necessity of the special methods of transition from abstract to the structural finite state machines that work in dependability systems is grounded. It is suggested to use the one-shot coding of the states of finite state machines, and also to perform algorithms transformation, consisting of wait states replacement by associated states pairs which duplicate each other.

Key words: dependable system, information security, criptoprocessor, finite state machine, states coding, one-shot code.

Глухов Валерий Сергеевич – канд. техн. наук, доцент, доцент кафедры электронных вычислительных машин Национального университета «Львівська політехніка», Львов, Украина, e-mail: valeriygl@ukr.net.

Родриг Ильяс – аспирант, кафедра электронных вычислительных машин Национального университета «Львівська політехніка», Львов, Украина, e-mail: relias@ndu.edu.lb.