электронный журнал
МОЛОДЕЖНЫЙ НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК
Издатель ФГБОУ ВПО "МГТУ им. Н.Э. Баумана". Эл No. ФС77-51038.

УДК 629.127

Аппаратная архитектура дешифрования алгоритма DES

 # 09, сентябрь 2012

Кадилов Я.А.

Студент,
кафедра «Информационная безопасность»

Научный руководитель: Сухини Б.М.,
ассистент кафедры «Информационная безопасность»

МГТУ им. Н.Э. Баумана
Yaroslav.Kadilov@gmail.com

Сегодня аппаратные средства вычислительной техники переживают серьезные перемены. Данное явление вызвано в первую очередь появлением множества новых теоретических и практических задач, требующих от вычислительных систем большей гибкости по сравнению с классическими микропроцессорными технологиями. Также большое влияние оказывает потребность в более сложных схемах управления ресурсами, использующимися в современных системах, т.к. их объем благодаря удешевлению ежегодно увеличивается.

Показательной задачей является исследование стойкости криптографических алгоритмов методом тотального опробывания (полного перебора). Данный подход требует значительные вычислительные мощности, и вместе с ростом доступности аппаратных ресурсов растет привлекательность подобных исследований. В то же самое время данная задача является крайне специфичной и перебор даже сравнительно небольшого количества вариантов на универсальных микропроцессорах за адекватный временной промежуток представляется невозможным.

В качестве одного из самых популярных объектов криптоанализа в течение долгих лет был алгоритм Data Encryption Standard (DES). Полное описание алгоритма приведено в соответствующем стандарте Национального Института стандартов и технологий США [1]. Было спроектировано множество различных архитектур устройств дешифрования алгоритма DES, но подавляющее большинство из них не было реализовано либо из-за высокой стоимости, либо из-за других значительных ограничений. Существует ряд методов, позволяющих дешифровать DES быстрее полного перебора, но требующих слишком большое количество открытых текстов, зашифрованных на одном ключе. Среди них можно выделить работу Мицуро Мацуи 1994 года по линейному криптоанализу [2], требующему 2^{43} пар открытых и шифротектов. В 1997 году удалось дешифровать DES на большом количестве компьютеров добровольцев, соединенных через Интернет [3]. В атаке одновременно участвовало от 14.000 до 80.000 компьютеров. А также интерес представляет устройство стоимостью 210.000 долл. США, реализованное в 1997 году организацией Electronic Frontier Foundation [4].

Выдающихся результатов достигла группа немецких исследователей, возглавляемых профессором университета Ruhr University of Bochum К. Пааром (C. Paar) и профессором

университета Christian-Albrechts University Kiel М. Шиммлером (M. Schimmler). В 2006 году они разработали устройство под названием Cost-Optimized Parallel Code Breaker (COPACOBANA), которое при стоимости €8.980 способно было находить ключ в среднем за 8,7 дней [5]. Затем был проведен ряд усовершенствований, после которых устройство COPACOBANA в 2007 году дешифровало DES в среднем за 6,4 дня, при этом затраты на реализацию так же не превышали €10.000 [6]. В 2008 году на базе устройтсва COPACOBANA была создана дорогостоящая машина RIVYERA, дешифрующая алгоритм DES в среднем меньше, чем за один день [7]. Стоимость устройства, непревышающая €10.000, является революционной в области криптоанализа, поэтому создание более производительного устройства при сохранении низкой стоимости, описанное в данной работе, – это актуальная задача не только для накопления знаний в области криптоанализа, но и исследования возможностей новых перспективных принципов построения специализированных архитектур.

Важно отметить, что успехи группы немецких исследователей, были достигнуты именно благодаря использованию современных аппаратных вычислительных средств: программируемых логических интегральных схем (ПЛИС) класса Field-Programmable Gate Array (FPGA). Для успешного решения подобных задач в настоящее время также используются графические процессоры, но их главным недостатком по сравнению с FPGA является то, что они имеют сконфигурированную архитектуру. Низкая стоимость устройства COPACOBANA достигается благодаря использованию типовых, серийно производящихся микросхем, что является серьезным преимуществом FPGA перед Application-Specific Integrated Circuit (ASIC).

Задачей данной работы является разработка архитектуры устройства дешифрования алгоритма DES в режиме электронной кодовой книги (ECB), предназначенной для реализации на FPGA. Тип атаки – атака по выбранному открытому тексту (chosen plaintext attack). Предполагается, что пользователь устройства имеет возможность задать в качестве входных данных две пары выбранных открытых текстов и получившихся в результате шифрования на одном неизвестном ключе шифротекстов. Данная возможность необходима для использования свойства дополнения DES, которое будет описано подробно ниже. На выходе устройство должно выдавать искомый ключ, на котором было осуществлено шифрование. Стоимость реалиации устойства не должна превышать €10.000, среднее время поиска одного неизвестного ключа должно быть меньше 6,4 дня. Опробование архитектуры необходимо осуществить на макете устройства дешифрования, реализованном на отладочной плате Altera Cyclone II Starter Development Kit, предоставляющейся кафедрой «Информационная безопасность» МГТУ им. Н.Э. Баумана. На отладочной плате размещена модель FPGA Altera Cyclone II EP2C20F484C7. Макет должен быть способен подобрать ключ, если он заведомо будет выбран из определенного подмножества множества всевозможных ключей.

Для проектирования архитектуры необходимо выделить некоторые наиболее важные характеристики имеющейся модели FPGA:

- Цена;
- Количество логических ячеек;
- Частота работы.

Цена микросхемы Altera Cyclone II EP2C20F484C7 составляет 61,93 долл. США, если ее заказывать на сайте производителя.

Количество логических ячеек Altera Cyclone II EP2C20F484C7 составляет 18.752 шт.

В FPGA Altera Cyclone II EP2C20F484C7 логические ячейки организованы таким образом, что схема, содержащая только один регистр длиной до 32 бит, работает на частоте приблизительно равной 350 МГц, значение дает существенное представление о предельной частоте работы устройства. Частота 350 МГц для FPGA Altera Cyclone II EP2C20F484C7 является очень хорошей характеристикой, т.к. она представляет недорогой ценовой сегмент, но при использовании большого числа ячеек используется большая площадь отладочной

платы, что соответственно снижает максимальную частоту, на которой может работать устройство. Задача дешифрования – задача, требующая большого объема вычислительных ресурсов, в случае FPGA логических ячеек, поэтому с учетом исследовательских схем, занимающих большую площадь FPGA, 100 МГц является хорошей частотой устройства на Altera Cyclone II EP2C20F484C7. Наличие арифметических операций, например, сложения, используемого в счетчике ключей, так же существенно снижает максимальную частоту, на которой может работать устройство на FPGA. Поэтому в разработанной архитектуре был применен прием разделения счетчика на две части длиной 32 и 56 - 32 = 24 бита, что позволило избежать снижения частоты из-за использования сложения счетчика длины 56 бит.

Отладочная плата Altera Cyclone II Starter Development Kit предоставляет возможность использовать как память, расположенную непосредственно в микросхеме Altera Cyclone II EP2C20F484C7, представляющую собой регистры сдвига, так и внешнюю память: оперативное запоминающее устройство и постоянное запоминающее устройство. Но для задачи дешифрования требуется большая пропускная способность из-за параллельной обработки данных, поэтому из приведенных типов памяти есть возможность использовать только регистры сдвига. Частота, на которой они могут работать, составляет 200 МГц. Предполагаемая частота работы устройства, как было сказано, находится не на таком высоком уровне, поэтому использование регистров сдвига целесообразно.

Нужно заметить, что материальные средства в виде отладочной платы с FPGA Altera Cyclone II EP2C20F484C7, предоставляемые кафедрой «Информационная безопасность» МГТУ им. Н.Э. Баумана, являются очень полезными, т.к. данная модель FPGA по основным характеристикам соответствует используемой в COPACOBANA Xilinx Spartan-3 XC3S1000.

FPGA содержит множество логических ячеек и различных связей между ними. Работа FPGA программируется в соответствии с требованиями к создаваемому устройству, что позволяет соединить ячейки таким образом, чтобы FPGA содержала несколько параллельно работающих блоков. В случае дешифрования блочного алгоритма DES на FPGA размещается несколько одинаково работающих потоков шифрования и блоки сравнения результата с входным шифротекстом. Блочный алгоритм шифрования DES имеет 16 циклов шифрования. FPGA позволяет производить все циклы или часть параллельно. Соответственно число тактов, необходимых на проверку одного ключа может изменяться.

Алгоритм DES обладает свойством дополнения:

$$\overline{\mathrm{DES}_{\overline{k}}(\overline{P})} = \mathrm{DES}_k(P),$$

где

$\overline{X}$ – дополнение к X;

$\mathrm{DES}_k(P)$ – шифрование открытого текста P по алгоритму DES, используя ключ k.

Данное свойство может быть использовано для проверки двух ключей за один такт. Имея две пары открытого и шифротекста вида (P, C_1) и $(\overline{P}, C_2)$, во время шифрования P ключом k проверяются равенства $C = C_1$ и $C = \overline{C_2}$. Если выполняется первое равенство, то искомым ключом является ключ k, если второе – $\overline{k}$.

Т.к. разрабатываемая архитектура предполагает атаку по выбранному открытому тексту, то на вход помимо открытого текста и шифротекста подается результат шифрования на этом же неизвестном ключе дополнения открытого текста.

Алгоритм DES предполагает начальную и конечную перестановки. Чтобы не производить одни и те же вычисления при проверке каждого ключа, достаточно всего один раз выполнить операцию начальной перестановки над открытым текстом и один раз операцию, обратную операции конечной перестановки, над шифротектами и впоследствии работать с получившимися векторами.

Значительные проблемы представляет выполнение операции получения значения из S-блоков по заданным аргументам. Данная операция является самой ресурсоемкой как по

времени, так и по площади. В разработанной архитектуре она реализована на мультиплексорах. Для того чтобы сократить величину такта, все преобразования находящиеся в алгоритме перед и после непосредственного получения значения из S-блоков, необходимо отнести к соседним операциям, хотя описание алгоритма в стандарте NIST построено совсем другим образом. Так же необходимо исследовать возможность разделения операции получения значения из S-блоков на две подоперации.

Среднее время дешифрования – это математическое ожидание случайной величины времени поиска неизвестного ключа. В каждом устройстве, осуществляющем полный перебор, ключ может быть найден в первую секунду, а может быть найден только, когда пройдет проверку все пространство ключей. При шифровании блочным алгоритмом DES ключ должен выбираться случайным образом. На данный момент существует множество исследовательских работ и устройств в области генерации псевдослучайных последовательностей. Поэтому при нахождении математического ожидания времени поиска конкретного ключа можно считать, что каждый ключ из пространства ключей имеет одну и ту же вероятность.

Пусть

n – количество плат, входящих в устройство;

m – количество потоков, размещенных на каждой плате.

Эти параметры следует выбирать из значений, являющихся степенью 2, чтобы в устройстве не находилось простаивающих потоков и не расходовать дополнительное количество логических ячеек для размещения нескольких генераторов ключей. В том случае, если n и m будут представлять собой степени 2, то на плате FPGA возможно разместить один генератор ключей для всех потоков, различие подпространств ключей для каждого из потоков будет заключаться в различии номера платы и номера потока. Т.е. если n и m выбрать в виде 2 в какой-то степени, то возможно выбрать определенное количество старших бит, которые будут определять, начиная с какого ключа, следует осуществлять перебор каждому из потоков.

С учетом данного правила выведенная точная формула среднего времени дешифрования имеет вид:

$$t = \frac{h}{\nu} \cdot \frac{2^{55}}{n \cdot m} + \frac{h}{2 \cdot \nu},$$

где

ν – частота, на которой работает устройство;

h – количество тактов, необходимых для проверки одного ключа.

Количество плат в устройстве n необходимо выбрать равным 2^7, т.к. $2^7 \cdot 61{,}93 = 7.927$ долл. США удовлетворяет поставленной задаче. Большее количество плат не удовлетворяет: $2^8 \cdot 61{,}93 = 15.854$ долл. США. Очень важно заметить, что помимо затрат на сами FPGA, необходимо учитывать затраты на монтаж конечного устройства, включающего в себя все эти микросхемы. По опыту и на основании сайта компании «Резонит» (http://www.rezonit.ru/), осуществляющей монтаж печатных плат, можно сказать, что данные затраты составят около 2.000 долл. США. Соответственно в результате сумма, приблизительно равная 10.000 долл. США, удовлетворяет условиям поставленной задачи.

Значения частоты работы устройства ν, количества тактов, требующихся для проверки одного ключа, h и количества потоков в одной плате m ограничены количеством логических ячеек платы Altera Cyclone II EP2C20F484C7. Улучшение показателя каждого из параметров по-разному влияет на объем используемых ячеек. Это является важным фактором при выборе значений архитектурных параметров.

Параметр количества тактов, необходимых для проверки одного ключа, h по сути определяет, используется ли в устройстве конвейерная схема, частично-конвейерная или же простая схема реализации алгоритма. Пользуясь свойством дополнения алгоритма DES при любой из схем реализаций количество тактов, необходимых на проверку одного ключа уменьшается в два раза, т.к. одновременно с проверкой ключа проверяется его дополнение.

Алгоритм DES состоит из 16 итераций. Соответственно, при простой схеме параметр h равен 8. Но используя дополнительное количество ячеек можно выполнять все 16 итераций или часть в параллельном режиме. Тогда параметр h будет равен 16/(количество параллельно выполняющихся итераций). Значению h = 0,5 соответствует конвейерной схеме устройства. Т.к. в FPGA наиболее похожие блоки схемы используют общие и соседние ячейки, то при распараллеливании итераций объем используемых ячеек будет увеличиваться в меньшее количество раз по сравнению с увеличением количества параллельно выполняющихся итераций.

Распараллеливание потоков так же увеличивает объем используемых ячеек в меньшее количество раз, чем увеличивается количество параллельно работающих потоков. Но потоки – емкие блоки, и использование общих и соседних ячеек менее эффективно по сравнению с распараллеливанием относительно небольших групп регистров или других емких элементов.

Увеличение частоты работы устройства осуществляется добавлением дополнительных регистров, делящих итерацию на несколько дочерних итераций, требующих соответственно меньшее время для выполнения. Данный способ уменьшения времени дешифрования является менее эффективным по сравнению с распараллеливанием по следующим причинам:

- Одну итерацию практически невозможно разделить на равные по времени выполнения дочерние итерации, одна из дочерних итераций будет занимать существенную часть главной родительской итерации;
- Из-за специфики итерации дополнительные регистры не являются чистым дублированием уже имеющихся, поэтому сложнее образовать группу регистров, использующих множество общих ячеек;
- В случае использования неконвейерной схемы работы устройства, при увеличении частоты увеличивается число тактов, требующихся для проверки одного ключа, т.к. из-за дополнительных регистров увеличивается число тактов, необходимых для получения результата каждой итерации.

При реализации макета устройства были использованы все методы увеличения производительности устройства, но в разной степени. Для практического подтверждения сделанных на этапе проектирования выводов об эффективности различных методов сокращения времени дешифрования было реализовано несколько архитектур.

Таблица 1. Характеристики, получившиеся в результате реализации архитектур с различными методами увеличения производительности

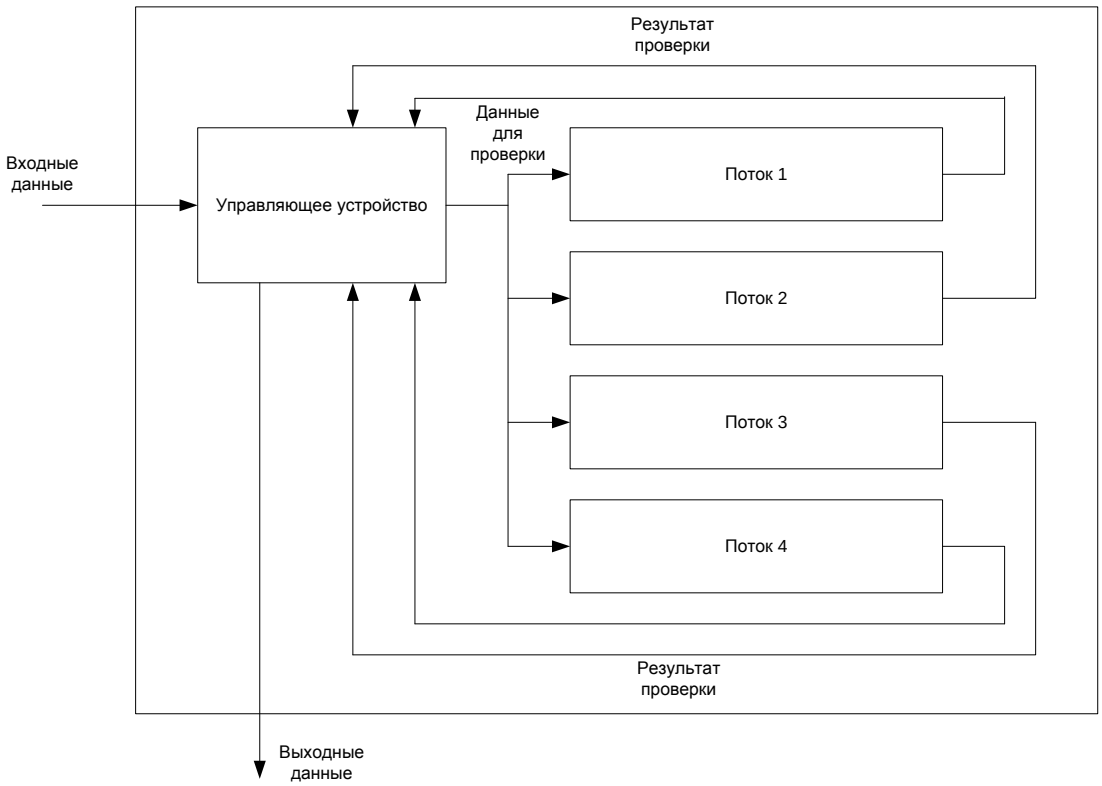
Тип архитектуры	Частота, ν, МГц	Количество тактов, необходимых для проверки одного ключа, h, шт	Количество потоков в одной FPGA, m, шт	Количество используемых логических ячеек, шт	Используемая память, бит (%)	Время дешифрования одного неизвестного ключа, t, дн
Простая схема	106,32	8	32	17 564 (94%)	56 788 (24%)	15,32
Конвейерная схема	135,41	0,5	4	18 182 (97%)	41 984 (18%)	3,01
Конвейерная схема с повышенной частотой	171,7	0,5	2	10 898 (58%)	44 174 (18%)	9,49

Из таблицы 1 видно, что снижение количества тактов, необходимых на проверку одного ключа, является самым эффективным способом. При использовании конвейерной схемы количество тактов уменьшается в 16 раз, а количество потоков снижается всего лишь в 8 раз по сравнению с простой схемой.

Конвейерная схема с повышенной частотой посредством разделения операции получения значения из S-блоков на подоперации дает возможность устройству работать на частоте 171,7 МГц. Т.е. по сравнению с конвейерной схемой увеличение частоты происходит в 1,27 раз. Но количество потоков из-за возросшего объема ячеек снизилось в 2 раза. Такой вариант оказался неприемлимым для имеющейся FPGA Altera Cyclone II EP2C20F484C7, т.к. несмотря на значительно уменьшенную длительность одного такта, уменьшение количества потоков в одной FPGA является для задачи дешифрования критичным фактором. Но как видно из таблицы 1 схема с двумя потоками занимает всего 58%. Поэтому были рассмотрены FPGA, относящиеся к более высокому ценовому сегменту, но по-прежнему предполагающую доступную стоимость конечного устройства. И были сделаны выводы, что реализация устройства дешифрования на таких FPGA может быть в будущем выбрана в качестве направления дальнейшего исследования.

В результате конвейерная схема с 4 потоками, использующая спроектированные методы увеличения производительности, показывает отличные результаты и является решением поставленной задачи. Несмотря на то, что оптимизировались преобразования, производимые по алгоритму DES, практически все методы увеличения производительности основаны на распараллеливании вычислений. Поэтому достигнутые результаты могут быть легко обобщены на другие криптографические алгоритмы, причем для некоторых будет достаточно изменить только блоки потоков, и устройство будет готово для криптоанализа другого алгоритма.

На рисунке представлена структурная схема одной из 128 FPGA устройства дешифрования, соответствующая выбранным параметрам архитектуры.



При реализации макета устройства ввиду необходимости запаса частоты и ограничений на возможные значения была назначена конечная рабочая частота 120,00 МГц. Следовательно, среднее время дешифрования алгоритма DES на созданном устройстве равняется:

$$t = \frac{h}{\nu} \cdot \frac{2^{55}}{n \cdot m} + \frac{h}{2 \cdot \nu} = \frac{\frac{1}{2}}{120 \cdot 10^6} \cdot \frac{2^{55}}{2^7 \cdot 2^2} + \frac{\frac{1}{2}}{2 \cdot 120 \cdot 10^6} = 293203 \text{ с} \approx 3{,}39 \text{ дн.}$$

Величина такта равняется:

$$\frac{1}{\nu} = \frac{1}{120 \cdot 10^6} = 8{,}33 \text{ нс.}$$

В законченном устройстве за один такт проверяется:

$$\frac{1}{h} \cdot n \cdot m = 2 \cdot 27 \cdot 22 = 1024 \text{ ключа.}$$

За 1 секунду устройство проверяет:

$$1024 \cdot \frac{1}{8{,}33 \cdot 10^{-9}} = 1{,}23 \cdot 10^{11} \text{ ключей,}$$

а на проверку одного ключа требуется:

$$\frac{8{,}33 \cdot 10^{-9}}{1024} = 8{,}14 \cdot 10^{-12} \text{ с.}$$

Макет устройства был успешно протестирован. Для FPGA был произвольно выбран номер от 0 до 127 и выставлен на переключателях отладочной платы, посредством которых FPGA присваивался необходимый номер. Затем были заданы тройки открытых, шифротекстов и шифротекстов, получившихся в результате шифрования дополнения к открытому тексту. Тестирование проводилось несколько раз на разных векторах, каждый раз устройство находило ключ. Ключ находился за короткий промежуток времени, т.к. он заведомо выбирался с большим количеством нулей или единиц (для свойства дополнения в алгоритме DES, при использовании которого искомый ключ представляет собой инверсию ключа, по которому фактически происходит текущее шифрование) после номера FPGA и номера потока. В таблице 2 представлены некоторые векторы, которые использовались при тестировании макета устройства.

Таблица 2. Тестовые векторы алгоритма DES

Открытый текст	Ключ	Шифротекст
D80DA4C32D3FA677	000000000000A5D6	123A8813A54168C1
27F25B3CD2C05988	000000000000A5D6	1793D3EF9D86E110
D80DA4C32D3FA677	FFFFFFFFFFFF5A29	E86C2C1062791EEF
27F25B3CD2C05988	FFFFFFFFFFFF5A29	EDC577EC5ABE973E

Итак, стоимость реализации спроектированной архитектуры не превышает 10.000 долл. США, при этом среднее время дешифрования почти в 2 раза меньше аналогичного показателя устройства COPACOBANA: 3,49 дня и 6,4 дня соответственно. Использованные методы оптимизации являются легко обобщаемыми на другие архитектуры, главным образом предназначенные для криптоанализа. Предложенная архитектура может использоваться для продолжения исследований в области производительности параллельной обработки данных, а также для исследования криптостойкости различных алгоритмов. Благодаря использованию универсальных методов увеличения производительности устройств на FPGA созданная аппаратная архитектура дешифрования DES может быть легко модифицирована для решения других схожих задач.

Список литературы:

1. The National Institute of Standards and Technology (NIST) Federal Information Processing Standards Publication (FIPS Pub) 46-3, Data Encryption Standard (DES), 1999. http:// csrc.nist.gov/ publications/ fips/ archive/ fips46-3/ fips46-3.pdf.
2. Mitsuru Matsui, Linear Cryptanalysis Method for DES Cipher, Advances in Cryptology, proceedings of EUROCRYPT ’93, Lecture Notes in Computer Science 765, Springer-Verlag, Berlin, 1994.
3. CNET News.com, Users take crack at 56-bit crypto. Электронное издание, http://news.com.com/2100-1023-278658.html?legacy=cnet, 1997.

4. Electronic Frontier Foundation, Cracking DES: Secrets of Encryption Research, Wiretap Politics & Chip Design, O'reilly Media, Sebastopol, 1998.
5. Sandeep Kumar, Christof Paar, Jan Pelzl, Gerd Pfeier, Andy Rupp, Manfred Schimmler, How to Break DES for €8,980, 2006. http:// www.copacobana.org/ paper/ copacobana_SHARCS2006.pdf.
6. Tim Guneysu, Timo Kasper, Martin Novotny, Christof Paar, Member, IEEE, and Andy Rupp, Cryptanalysis with COPACOBANA, 2008. http:// www.copacobana.org/ paper/ TC_COPACOBANA.pdf.
7. SciEngines sciengines.com, Break DES in less than a single Day. Электронное издательство, 2008. http:// www.sciengines.com/ company/ news-a-events/ 74-des-in-1-day.html?eab1dd0ce8f296f6302f76f8761818c0=0b59c5b91984fe01986ef 3bbc6de9871.
8. Cyclone II FPGA Starter Development Board: User Guide, 2006.
9. Cyclone II FPGA Starter Development Board: Reference Manual, 2006.
10. А.К. Поляков, Языки VHDL и VERILOG в проектировании цифровой аппаратуры, 2003.