Secure and Resilient PLD-Based Systems
Practicum

V. Sklyar
Edited by V.S. Kharchenko

Introduction in secure and resilient PLD computing

Methods of PLD-based systems security assessment

Methods of PLD-based systems security assurance

Methods of diversity-oriented dependability and resilience assurance

Secure and Resilient PLD-Based Systems. Practicum

Tempus 2013 SEREIN 2016

PRACTICUM

SECURE AND RESILIENT PLD-BASED SYSTEMS

2017

Co-funded by the Tempus Programme of the European Union

Министерство образования и науки Украины
Национальный аэрокосмический университет
им. Н.Е. Жуковского «ХАИ»

А. Е. Перепелицын

Безопасные и резильентные системы на программируемой логике

Secure and Resilient PLD-Based Systems

Практикум

Под редакцией В.С. Харченко

Проект
SEREIN 543968-TEMPUS-1-2013-1-EE-TEMPUS-JPCR
Modernization of Postgraduate Studies on Security and Resilience for Human and Industry Related Domains

2017

Викладено матеріали практичної частини курсу «Безпечні і резільєнтні системи на програмовної логіки» (CM5. Secure and Resilient PLD-Based Systems), підготовленого в рамках проекту TEMPUS SEREIN «Modernization of Postgraduate Studies on Security and Resilience for Human and Industry Related Domains» (543968-TEMPUS-1-2013-1-EE-TEMPUS-JPCR).

Наведені описи практичних робіт, які призначені для ознайомлення з особливостями апаратної реалізації генераторів випадкових чисел, фізично неклонованих функцій, деяких криптопримітивов, а також засобами підвищення стійкості до атак по сторонніх каналах. Надається навчальна програма курсу і опис лабораторних і практичних робіт.

Призначено для інженерів, які займаються розробленням та впровадженням апаратних засобів захисту інформації, для фахівців у галузі оцінювання якості та безпеки вбудованих систем, для магістрів і аспірантів університетів, які навчаються за напрямами інформаційної безпеки, комп'ютерних наук, комп'ютерної та програмної інженерії, а також для викладачів відповідних курсів.

Рецензенты:

– **Опанасенко Владимир Николаевич**, доктор техн. наук, профессор, ведущий научный сотрудник Института кибернетики им. В.М. Глушкова НАН Украины, Киев.

Перепелицын А.Е.
Безопасные и резильентные системы на программируемой логике. Практикум / под ред. В.С. Харченко – Министерство образования и науки Украины, Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ». 2017. – 55 с.

Изложены материалы практической части курса «Безопасные и резильентные системы на программируемой логике» (CM5. Secure and Resilient PLD-Based Systems), подготовленного в рамках проекта TEMPUS SEREIN «Modernization of Postgraduate Studies on Security and Resilience for Human and Industry Related Domains» (543968-TEMPUS-1-2013-1-EE-TEMPUS-JPCR).

Приведены описания практических работ, которые предназначены для ознакомления с особенностями аппаратной реализации генераторов случайных чисел, физически неклонируемых функций, некоторых криптопримитивов, а также средствами повышения стойкости к атакам по сторонним каналам. Предоставляется учебная программа курса и описание лабораторных и практических работ.

Предназначено для инженеров, занимающихся разработкой и внедрением аппаратных средств защиты информации, для специалистов по оценке качества и безопасности встроенных систем, для магистров и аспирантов университетов, обучающихся по направлениям информационной безопасности, компьютерных наук, компьютерной и программной инженерии, а также для преподавателей соответствующих курсов.

Библиогр.: 24 наименований, рисунков – 12.

Утверждено на заседании ученого совета Национального аэрокосмического университета имени Н.Е. Жуковского «ХАИ» (протокол № 6 от 22 февраля 2017 г.).

СПИСОК СОКРАЩЕНИЙ

ГПСЧ	–	Генератор псевдослучайных чисел
КА	–	Клеточный автомат
ПЛИС	–	Программируемая логическая интегральная схема
РСЛОС	–	Регистр сдвига с линейной обратной связью
РСНОС	–	Регистр сдвига с нелинейными обратными связями
ФНФ	–	Физически неклонируемые функции
AES	–	Advanced Encryption Standard
CA	–	Cellular Automaton
FPGA	–	Field Programmable Gate Array
GF	–	Galois Field
LFSR	–	Linear Feedback Shift Register
NLFSR	–	Non-Linear Feedback Shift Register
PLD	–	Programmable Logic Device
PRNG	–	Pseudo Random Number Generator
PUF	–	Physical Unclonable Function
RTL	–	Register Transfer Level
TRNG	–	True Random Number Generator
VHDL	–	Very high speed integrated circuit (VHSIC) Hardware Description Language

ВВЕДЕНИЕ

В пособии изложены материалы практической части курса «Безопасные и резильентные системы на программируемой логике» (CM5. Secure and Resilient PLD-Based Systems), подготовленного для магистров и аспирантов в рамках проекта TEMPUS SEREIN «Modernization of Postgraduate Studies on Security and Resilience for Human and Industry Related Domains» (543968-TEMPUS-1-2013-1-EE-TEMPUS-JPCR)[1]. Курс посвящен особенностям аппаратной реализации средств защиты информации, а также повышению их стойкости к атакам.

В пособии приводится описание практических работ, общей целью которых является рассмотрение встроенных систем и технологии FPGA в качестве объекта и средства обеспечения информационной безопасности. Рассматриваются как общие положения математического аппарата, применяемого в криптографии, так и исключительные особенности технологии FPGA.

Первые три лабораторные работы посвящены изучению аппаратной реализации генераторов псевдослучайных чисел (ГПСЧ), в основе которых лежат принципиально разные подходы к формированию выходной последовательности.

В первой работе рассматриваются возможные конфигурации построения ГПСЧ в ПЛИС на основе сдвиговых регистров с линейной обратной связью (РСЛОС).

Во второй работе рассматриваются особенности реализации ГПСЧ с использованием регистров сдвига с нелинейными обратными связями (РСНОС) (нелинейность второго порядка).

В третьей работе изучается построение одномерных клеточных автоматов в ПЛИС, их производительность, а также отличие свойств формируемой псевдослучайной последовательности в сравнении с результатами из предыдущих работ.

[1] *Этот проект финансируется при поддержке Европейской комиссии. Эта публикация (сообщение) отражает мнения только авторов, и Комиссия не может нести ответственность за любое использование содержащейся в нем информации.*

This project has been funded with support from the European Commission. This publication (communication) reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Четвертая лабораторная работа посвящена экспериментальному изучению реализации физически неклонируемых функций в FPGA, а также аппаратной реализации генераторов истинно случайных чисел (TRNG). Уникальность данной работы заключается в том, что технология FPGA рассматривается не только как инструмент, а и непосредственно как объект исследования. Результаты не могут быть получены вне лаборатории и не могут быть промоделированы.

Пятая и шестая работы посвящены изучению аппаратной реализации блочных алгоритмов шифрования на примере AES.

В пятой работе рассматриваются различные способы аппаратной реализации умножителей в поле Галуа в составе блока перемешивания столбцов (MixColumns) алгоритма AES.

Шестая работа предполагает изучение реализации в FPGA полного раунда алгоритма шифрования AES.

Седьмая работа посвящена рассмотрению атак по сторонним (побочным) каналам и повышению стойкости к ним применительно к технологии FPGA и встроенным системам.

Рисунки, таблицы и формулы для удобства нумеруются в пределах каждого раздела.

Курс предназначен для инженеров, занимающихся разработкой и внедрением аппаратных средств защиты информации, специалистов по оценке качества и безопасности встроенных систем, для магистров и аспирантов университетов, обучающихся по направлениям информационной безопасности, компьютерных наук, компьютерной и программной инженерии, а также для преподавателей соответствующих курсов.

Методическое пособие подготовлено сотрудником кафедры компьютерных систем и сетей Национального аэрокосмического университета им. Н.Е. Жуковского «ХАИ»: старшим преподавателем Перепелицыным А.Е. Общее редактирование проведено доктором технических наук, профессором, заслуженным изобретателем Украины Харченко В.С.

Автор выражает благодарность рецензентам, коллегам по проекту, сотрудникам кафедр академических университетов, индустриальным партнерам за ценную информацию, методическую помощь и конструктивные предложения, которые высказывались в процессе обсуждения программы курса и материалов пособия.

1. РЕАЛИЗАЦИЯ ГЕНЕРАТОРА ПСЕВДОСЛУЧАЙНЫХ ЧИСЕЛ В FPGA НА ОСНОВЕ СДВИГОВОГО РЕГИСТРА С ЛИНЕЙНОЙ ОБРАТНОЙ СВЯЗЬЮ

Форма занятия: практикум

Цель и задачи практикума – изучить существующие архитектуры аппаратной реализации генераторов псевдослучайных чисел, а также ознакомиться с возможностями таких генераторов, основанных на сдвиговом регистре с линейной обратной связью (в конфигурации Фибоначчи и Галуа).

Практические задачи:

– закрепить навыки работы в автоматизированных средах проектирования встроенных систем;

– получить навыки реализации РСЛОС в конфигурации Фибоначчи в соответствии с заданием;

– получить навыки реализации РСЛОС в конфигурации Галуа в соответствии с заданием.

Исследовательские задачи:

– провести сравнительный анализ псевдослучайных последовательностей, формируемых РСЛОС в двух конфигурациях;

– проанализировать статистические свойства формируемых последовательностей.

Подготовка к практикуму

При подготовке к лабораторной работе необходимо:

– повторить общую структуру регистра сдвига с линейными обратными связями в конфигурациях Фибоначчи и Галуа;

– изучить теоретический материал, приведенный в описании, а также в [1-6].

Ход работы

1. Реализовать в соответствии с заданием генератор псевдослучайных чисел в среде Quartus II Web Edition. В качестве способа реализации сдвигающего регистра может быть выбрано описание на языке VHDL или построение схемотехнического решения.

Генератор псевдослучайных чисел должен быть реализован на основании полинома 12-й степени:

– в конфигурации Фибоначчи;

– в конфигурации Галуа.

Генератор псевдослучайных чисел должен предусматривать возможность задания начального значения (слова инициализации).

2. Для полученного генератора псевдослучайных чисел в каждой конфигурации необходимо провести моделирование с двумя исходными последовательностями:

– 010110101110;

– произвольное ненулевое значение.

3. Для каждого начального значения необходимо получить последовательность не менее чем из 100 битов. В полученной последовательности необходимо посчитать процент '0' и '1'.

Требования к содержанию отчета

Формируемый отчёт должен включать следующие пункты:

1. Титульный лист.

2. Задание на работу, включая цель и поставленные задачи.

3. Архитектура генератора псевдослучайных чисел в двух конфигурациях.

4. Реализация генератора псевдослучайных чисел (схема или VHDL).

5. Скриншоты полученной последовательности для двух начальных значений в каждой конфигурации.

6. Выводы по результатам выполнения работы. Выводы по работе должны содержать не только обобщение полученных результатов, а и говорить, что из этого следует. В выводах также отмечаются предложения и рекомендации по дальнейшему исследованию поставленной в работе задачи.

7. Приложения. В приложения выносится библиографический список, а также справочная и прочая информация, не включённая в основные разделы отчёта.

Контрольные вопросы

1. Что такое ГПСЧ?
2. Что такое РСЛОС?
3. Чем определяется длина неповторяющейся части последовательности?
4. Что такое неприводимый полином?
5. Какие существуют способы построения ГПСЧ на основании РСЛОС?
6. Какие достоинства и недостатки каждой из конфигураций вам известны?
7. Какой разряд регистра целесообразно использовать для формирования выходной последовательности?
8. Каковы недостатки генератора псевдослучайных чисел на основе сдвигового регистра с линейной обратной связью?
9. Как повысить стойкость формируемой последовательности к применению алгоритма Берлекэмпа-Мэсси?
10. Чем объясняется процент '1' и '0' в полученных вами последовательностях?

2. РЕАЛИЗАЦИЯ ГЕНЕРАТОРА ПСЕВДОСЛУЧАЙНЫХ ЧИСЕЛ В FPGA НА ОСНОВЕ СДВИГОВОГО РЕГИСТРА С НЕЛИНЕЙНОЙ ОБРАТНОЙ СВЯЗЬЮ

Форма занятия: практикум

Цель и задачи практикума – изучить существующие архитектуры аппаратной реализации генераторов псевдослучайных чисел, а также ознакомиться с возможностями таких генераторов, основанных на сдвиговом регистре с нелинейной обратной связью[2].

Практические задачи:

– закрепить навыки работы в автоматизированных средах моделирования встроенных систем;

– получить навыки реализации РСНОС с нелинейностью второго порядка.

Исследовательские задачи:

– провести сравнение псевдослучайных последовательностей, формируемых РСЛОС и РСНОС;

– проанализировать статистические свойства формируемой последовательности.

Подготовка к практикуму

При подготовке к лабораторной работе необходимо:

– повторить общую структуру регистра сдвига с нелинейными обратными связями с нелинейностью второго порядка;

– изучить теоретический материал, приведенный в описании, а также в [7-8].

[2] Данная работа была подготовлена по результатам исследований и при активной поддержке Полуяненко Николая Александровича (кафедра безопасности информационных систем и технологий, Харьковский национальный университет имени В.Н. Каразина, Харьков, Украина).

Ход работы

1. Реализовать в соответствии с заданием генератор псевдослучайных чисел в среде Quartus II Web Edition. В качестве способа реализации сдвигающего регистра может быть выбрано описание на языке VHDL или построение схемотехнического решения.

Генератор псевдослучайных чисел должен быть реализован на основании полинома 12-й степени с нелинейностью второго порядка.

Генератор псевдослучайных чисел должен предусматривать возможность задания начального значения.

2. Для полученного генератора псевдослучайных чисел в каждой конфигурации необходимо провести моделирование с двумя исходными последовательностями:

– 010110101110;

– произвольное ненулевое слово инициализации.

3. Для каждого начального значения необходимо получить последовательность не менее чем из 100 битов. В полученной последовательности необходимо посчитать процент '0' и '1'.

Требования к содержанию отчета

Формируемый отчёт должен включать следующие пункты:

1. Титульный лист.

2. Задание на работу, включая цель и поставленные задачи.

3. Архитектура генератора псевдослучайных чисел на основании РСНОС.

4. Реализация генератора псевдослучайных чисел (схема или VHDL).

5. Скриншоты полученной последовательности для двух начальных значений.

6. Выводы по результатам выполнения работы. Выводы по работе должны содержать не только обобщение полученных результатов, а и говорить, что из этого следует. В выводах также отмечаются предложения и рекомендации по дальнейшему исследованию поставленной в работе задачи.

7. Приложения. В приложения выносится библиографический список, а также справочная и прочая информация, не включённая в основные разделы отчёта.

Контрольные вопросы

1. Что такое ГПСЧ?
2. Что такое РСНОС?
3. Чем определяется длина неповторяющейся части последовательности для РСНОС?
4. Что такое образующий полином для регистра сдвига с нелинейной обратной связью?
5. Как найти полином для РСНОС, который позволяет формировать последовательность максимальной длины?
6. Чем определяется порядок нелинейности полинома для РСНОС?
7. Каковы достоинства и недостатки генератора псевдослучайных чисел на основе сдвигового регистра с нелинейной обратной связью?
8. Чем отличается с точки зрения криптоаналитика процесс анализа последовательности на выходе РСНОС и РСЛОС?
9. Чем объясняется процент '1' и '0' в полученных вами последовательностях?
10. Отличаются ли статистические показатели от результатов исследования регистра сдвига с линейной обратной связью из предыдущей работы? Могут ли иметь практическую значимость такие отличия?

3. РЕАЛИЗАЦИЯ ГЕНЕРАТОРА ПСЕВДОСЛУЧАЙНЫХ ЧИСЕЛ В FPGA НА ОСНОВЕ КЛЕТОЧНЫХ АВТОМАТОВ

Форма занятия: практикум

Цель и задачи практикума – ознакомиться с аппаратной реализацией генераторов псевдослучайных чисел на основе одномерных клеточных автоматов, а также изучить существующие правила, определяющие закон их работы[3].

Практические задачи:

– получить навыки аппаратной реализации одномерных клеточных автоматов;

– реализовать ГПСЧ на основе клеточного автомата.

Исследовательские задачи:

– исследовать зависимость свойств формируемых последовательностей он начального состояния клеточного автомата для заданного правила;

– проанализировать статистические свойства формируемой последовательности.

Подготовка к практикуму

При подготовке к лабораторной работе необходимо:

– повторить закон работы одномерного клеточного автомата для произвольного правила;

– изучить теоретический материал, приведенный в описании, а также в [9-10].

[3] Данная работа была подготовлена при активной поддержке и по результатам исследований Остапова Сергея Эдуардовича (доктор физ.-мат. наук, профессор, заведующий кафедрой программного обеспечения компьютерных систем Черновицкого национального университета имени Юрия Федьковича, Черновцы, Украина).

Ход работы

1. Реализовать в среде Quartus II Web Edition одномерный клеточный автомат на основе десятичного представления правила, указанного в задании. В качестве способа реализации может быть выбрано описание на языке VHDL или построение схемотехнического решения.

Генератор псевдослучайных чисел на основе клеточных автоматов должен предусматривать возможность задания начального значения (слова инициализации) и формировать на выходе значение сразу всего поколения. Разрядность регистра равна 12. Рекомендуется использовать два регистра: первый для текущего значения, второй – для формируемых значений в соответствии с правилом. После окончания формирования новых значений, содержимое второго регистра помещается в первый.

2. Для полученного генератора псевдослучайных чисел на основе клеточного автомата необходимо провести моделирование с двумя исходными последовательностями:

– 010110101110;

– произвольное ненулевое значение.

3. Во время моделирования необходимо сформировать 12 поколений, начиная с исходного. Формирование каждого поколения подразумевает перезапись регистра клеточного автомата. Для каждой клетки необходимо получить последовательность из 12 битов.

Требования к содержанию отчета

Формируемый отчёт должен включать следующие пункты:

1. Титульный лист.

2. Задание на работу, включая цель и поставленные задачи.

3. Архитектура клеточного автомата на основе требуемого правила.

4. Реализация генератора псевдослучайных чисел (схема или VHDL).

5. Скриншоты полученной последовательности для двух начальных значений.

6. Выводы по результатам выполнения работы. Выводы по работе должны содержать не только обобщение полученных результатов, а и говорить, что из этого следует. В выводах также отмеча-

ются предложения и рекомендации по дальнейшему исследованию поставленной в работе задачи.

7. Приложения. В приложения выносится библиографический список, а также справочная и прочая информация, не включённая в основные разделы отчёта.

Контрольные вопросы

1. Каковы особенности генератора псевдослучайных чисел на основе клеточного автомата?
2. Какие существуют правила?

4. ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ РЕАЛИЗАЦИИ ФИЗИЧЕСКИ НЕКЛОНИРУЕМЫХ ФУНКЦИЙ В FPGA

Форма занятия: практикум

Цель и задачи практикума – изучить реализацию в FPGA комбинированных физически неклонируемых функций, а также экспериментально определить статистические характеристики их отдельных разрядов.

Практические задачи:

– закрепить навыки работы с макетными платами Altera при прототипировании встроенных систем;

– получить навыки реализации физически неклонируемых функций в FPGA;

– получить навыки реализации генераторов истинно случайных чисел в FPGA.

Исследовательские задачи:

– экспериментально исследовать свойства созданной физически неклонируемой функции в разных экземплярах FPGA;

– сделать вывод о применимости отдельных разрядов ФНФ для задачи идентификации в каждом конкретном случае;

– сделать вывод о применимости отдельных разрядов ФНФ для генерации истинно случайных чисел для каждого конкретного экземпляра FPGA;

– предложить универсальную архитектуру, повышающую качество формируемых наборов, как для ФНФ, так и для TRNG.

Подготовка к практикуму

При подготовке к лабораторной работе необходимо:

– повторить базовые сведения о физически неклонируемых функциях, включая свойства каждой из архитектур;

– проанализировать архитектуру физически неклонируемой функции из задания и выбрать способ реализации (схемотехническая реализация или описание на языке VHDL);

– изучить теоретический материал, приведенный в описании, а также в [11-12].

Ход работы

1. Реализовать в среде Quartus II Web Edition схему комбинированной физически неклонируемой функции. В качестве базового элемента следует использовать мультиплексор с инверсным выходом, подключённым через прямую и инверсную связь к его входам. Такая схема позволяет переключаться между устойчивым и нестабильным состояниями (Рис. 4.1).

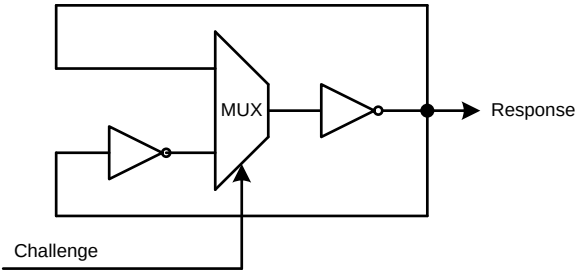

Рис. 4.1. Один элемент физически неклонируемой функции

2. Необходимо собрать схему из 26 таких элементов, соединенных параллельно и объединенных по входу общей линией Challenge (схема ФНФ с 1 входом и 26 выходами). Задание может быть выполнено также на языке описания аппаратуры VHDL.

3. Разработанный проект необходимо настроить для использования в макетной плате Altera DE2 board (указать соответствующий чип FPGA). Выходы схемы необходимо подключить последовательно к красным светодиодам LEDR0 – LEDR17 и зеленым светодиодам LEDG0 – LEDG7.

4. Вход схемы необходимо подключить к кнопке KEY0 для организации управления процессом формирования ответных значений. После чего необходимо скомпилировать проект и запрограммировать макетную плату с FPGA.

Для полученной реализации необходимо экспериментально определить какое количество раз горел каждый из 26 светодиодов при 20 запусках (при нажатии на кнопку KEY0). Для удобства подсчёта статистики следует использовать Microsoft Excel с автоматическим нахождением суммы в строках (или в столбцах).

5. Вход схемы необходимо подключить к линиям каждого доступного кварцевого генератора (27 МГц в первом эксперименте и 50 МГц во втором) для организации автоматического формирования с фиксированным периодом следования сигналов. После настройки одного из источников тактирования необходимо скомпилировать проект (сперва для PIN_D13, затем для PIN_N2) и запрограммировать макетную плату с FPGA.

Для полученной реализации необходимо визуально определить номера светодиодов (интенсивность свечения светодиодов может быть указана в вещественном диапазоне от 0 до 1.0):

– практически погасших;

– произвольно мигающих;

– непрерывно горящих.

Указанные действия необходимо повторить не менее чем для трёх макетных плат с одним и тем же проектом (без перекомпиляции проекта). Серийные номера используемых плат следует внести в отчёт для полученных в эксперименте наборов.

Предложить схему самонастройки, которая после инициализации будет формировать всегда только детерминированные значения для каждого выходного разряда в конкретной плате (т.е. позволит перейти от вероятностного результата к детерминированному). Решение может быть основано на наборе счётчиков, анализирующих результаты генерации во время инициализации.

Требования к содержанию отчета

Формируемый отчёт должен включать следующие пункты:

1. Титульный лист.

2. Задание на работу, включая цель и поставленные задачи.

3. Схема (или VHDL описание) одной ячейки комбинированной физически неклонируемой функции.

4. Схема (или VHDL описание) 26-разрядной реализации ФНФ.

5. RTL представление проекта 26-разрядной ФНФ.

6. Таблица с результатами эксперимента из второй части задания.

7. Номера светодиодов каждой указанной категории для каж-

дой использованной макетной платы (необходимо указать серийный номер макетной платы).

8. Выводы по результатам выполнения работы. Выводы по работе должны содержать не только обобщение полученных результатов, а и говорить, что из этого следует. В выводах также отмечаются предложения и рекомендации по дальнейшему исследованию поставленной в работе задачи.

9. Приложения. В приложения выносится библиографический список, а также справочная и прочая информация, не включённая в основные разделы отчёта.

Контрольные вопросы

1. Какие из исследованных разрядов подходят для организации ФНФ и почему?

5. РЕАЛИЗАЦИЯ БЛОКА ПЕРЕМЕШИВАНИЯ СТОЛБЦОВ АЛГОРИТМА ШИФРОВАНИЯ AES В FPGA

Форма занятия: практикум

Цель и задачи практикума – изучить реализацию в FPGA умножителей в поле Галуа, а также отдельных частей алгоритма шифрования AES.

Практические задачи:

– получить навыки аппаратной реализации умножителей в поле Галуа;

– закрепить навыки работы в автоматизированных средах проектирования встроенных систем;

– закрепить навыки работы в автоматизированных средах моделирования встроенных систем.

Исследовательские задачи:

– провести сравнительный анализ количества используемых ресурсов FPGA для различных способов реализации умножителя в поле Галуа;

– выполнить аналитические вычисления для подтверждения результатов моделирования.

Подготовка к практикуму

При подготовке к лабораторной работе необходимо:

– повторить способы реализации умножителей в поле Галуа;

– изучить теоретический материал, приведенный в описании, а также в [15-16].

Теоретический материал

Аналитическая запись перемешивания имеет следующий вид:

$$a'(x) = C(x) \cdot a(x) (\mathrm{mod}(x^4 + 1)),$$

где $C(x) = '03' x^3 + '01' x^2 + '01' x + '02'$,

$$a(x) = a_{3j} x^3 + a_{2j} x^2 + a_{1j} x + a_{0j} (\mathrm{mod} x^4 + 1)$$

Матричное представление имеет вид:

$$\begin{vmatrix} a'_0 \\ a'_1 \\ a'_2 \\ a'_3 \end{vmatrix} = \begin{vmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{vmatrix} \cdot \begin{vmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{vmatrix}$$

Ход работы

1. Разработать в среде Quartus II Web Edition блок, реализующий функциональность умножителя в поле Галуа. Используя этот умножитель как компонент, реализовать блок перемешивания столбцов алгоритма шифрования AES. В качестве способа реализации умножителя в поле Галуа и самого блока перемешивания может быть выбрано описание на языке VHDL, построение схемотехнического решения или оба подхода совместно (приветствуется алгоритмическая реализация, однако допустима и табличная).

Рекомендуется использовать следующие источники: [1], [2], [3].

2. Провести моделирование полученного блока, подав на его вход наборы значений из задания. Для подтверждения правильности результата необходимо сравнить результаты моделирования с программным расчетом результатов для заданного набора значений. С этой целью могут быть использованы различные программы, такие как MATLAB. Возможно написание собственной программы для проверки корректности результатов моделирования (приветствуется).

Требования к содержанию отчета

Формируемый отчёт должен включать следующие пункты:

1. Титульный лист.
2. Задание на работу, включая цель и поставленные задачи.
3. Реализация умножителя в поле Галуа (схема или листинг VHDL кода).
4. Реализация блока перемешивания столбцов (схема или листинг VHDL кода).
5. Скриншоты результатов моделирования каждого из шести наборов по варианту задания.
6. Результаты перемешивания, полученные с использованием ПО.
7. Листинг кода разработанного ПО (если выполнялась разработка).
8. Выводы по результатам выполнения работы. Выводы по работе должны содержать не только обобщение полученных результатов, а и говорить, что из этого следует. В выводах также отмеча-

ются предложения и рекомендации по дальнейшему исследованию поставленной в работе задачи.

9. Приложения. В приложения выносится библиографический список, а также справочная и прочая информация, не включённая в основные разделы отчёта.

6. РЕАЛИЗАЦИЯ ОПЕРАЦИЙ ОДНОГО РАУНДА АЛГОРИТМА ШИФРОВАНИЯ AES В FPGA

Форма занятия: практикум

Цель и задачи практикума – изучить реализацию в FPGA отдельных частей алгоритма шифрования AES.

Практические задачи:

– получить навыки аппаратной реализации криптопримитивов, лежащих в основе большинства блочных шифров;

– закрепить навыки работы в автоматизированных средах проектирования встроенных систем;

– закрепить навыки работы в автоматизированных средах моделирования встроенных систем.

Исследовательские задачи:

– провести сравнительный анализ количества используемых ресурсов FPGA для различных способов реализации полного раунда алгоритма AES;

– выполнить аналитические вычисления для подтверждения результатов моделирования.

Подготовка к практикуму

При подготовке к лабораторной работе необходимо:

– повторить состав операций полного раунда алгоритма AES;

– изучить теоретический материал, приведенный в описании, а также в [17-18].

Ход работы

1. Реализовать в среде Quartus II Web Edition один раунд шифрования алгоритма AES. Для этого, в дополнение к блоку перемешивания из предыдущей работы, следует разработать блоки трансформации SubBytes, ShiftRows и AddRoundKey. Используя эти блоки как компоненты, собрать полный раунд шифрования алгоритма AES. В качестве способа реализации блоков трансформации и всего раунда может быть выбрано описание на языке VHDL, построение схемотехнического решения или оба подхода совместно. Рекомендуется использовать следующие источники: [1], [2].

2. Необходимо провести моделирование полученных операций раунда алгоритма шифрования AES, подав на его вход первые шестнадцать шестнадцатеричных значений из варианта задания. Для тестирования следует использовать следующий раундовый ключ

A0	88	23	2A
FA	54	A3	6C
FE	2C	39	76
17	B1	39	05

Для подтверждения правильности результата необходимо сравнить результаты моделирования с программным расчетом результатов для заданного набора значений. С этой целью могут быть использованы различные программы, такие как MATLAB. Приветствуется написание собственной программы для проверки корректности результатов моделирования.

Требования к содержанию отчета

Формируемый отчёт должен включать следующие пункты:

1. Титульный лист.
2. Задание на работу, включая цель и поставленные задачи.
3. Реализация блоков SubBytes, ShiftRows и AddRoundKey (схема или листинг VHDL кода).
4. Реализация одного раунда алгоритма AES (схема или листинг VHDL кода).
5. Скриншоты результата моделирования с заданным ключом раунда по варианту задания.

6. Результаты тестирования, полученные для тех же значений с использованием ПО.

7. Листинг кода разработанного ПО (если выполнялась разработка).

8. Выводы по результатам выполнения работы. Выводы по работе должны содержать не только обобщение полученных результатов, а и говорить, что из этого следует. В выводах также отмечаются предложения и рекомендации по дальнейшему исследованию поставленной в работе задачи.

9. Приложения. В приложения выносится библиографический список, а также справочная и прочая информация, не включённая в основные разделы отчёта.

АНОТАЦІЯ

Перепелицин А.Є., **Безпечні і резільєнтні системи на програмовної логіки. Практикум**. / За ред. Харченка В.С.– Міністерство освіти і науки України, Національний аерокосмічний університет імені М. Є. Жуковського «ХАІ». - 2017. – 55 с.

У посібнику викладені матеріали практичної частини курсу «Безпечні і резільєнтні системи на програмовної логіки» (CM5. Secure and Resilient PLD-Based Systems), підготовленого в рамках проекту TEMPUS SEREIN «Modernization of Postgraduate Studies on Security and Resilience for Human and Industry Related Domains» (543968-TEMPUS-1-2013-1-EE-TEMPUS-JPCR).

Курс присвячений ознайомленню з особливостями апаратної реалізації генераторів випадкових чисел, фізично неклонованих функцій, деяких кріптопрімітівов, а також засобами підвищення стійкості до атак по сторонніх каналах. Надається навчальна програма курсу і опис лабораторних і практичних робіт.

Книга призначена для інженерів, які займаються розробленням та впровадженням апаратних засобів захисту інформації, для фахівців у галузі оцінювання якості та безпеки вбудованих систем, для магістрів і аспірантів університетів, які навчаються за напрямами інформаційної безпеки, комп'ютерних наук, комп'ютерної та програмної інженерії, а також для викладачів відповідних курсів.

Бібл. – 24 найменувань, рисунків – 12.

ЗМІСТ

ABSTRACT

Perepelitsyn A.E., **Secure and Resilient PLD-Based Systems. Practicum.** / Edited by Kharchenko V. S. – Department of Education and Science of Ukraine, National Aerospace University named after N. E. Zhukovsky “KhAI”, 2017. – 55 p.

The materials of the practical part of the practicum course “CM5. Secure and Resilient PLD-Based Systems”, developed in the framework of the TEMPUS SEREIN project "Modernization of Postgraduate Studies on Security Resilience for Human and Industry Related Domains" (543968-TEMPUS-1-2013-1-EE-TEMPUS-JPCR) are described.

Descriptions of practical works are provided that are intended for acquaintance with hardware implementation of random number generators, physical unclonable functions, some cryptographic primitives, as well as means of increasing of resistance to side channel attacks. A course curriculum and a description of laboratory work are provided.

It is intended for engineers engaged in the development and implementation of information security of hardware systems, for experts assessing the quality of embedded systems, for masters and postgraduate students of universities studying in areas of information security, computer science, computer and software engineering, as well as for teachers of relevant courses.

Ref. – 24 items, figures – 12.

CONTENT

ПРИЛОЖЕНИЕ А. УЧЕБНАЯ ПРОГРАММА КУРСА

DESCRIPTION OF THE MODULE

TITLE OF THE MODULE	Code
Secure and Resilient PLD-Based Systems	

Teacher(s)	Department
Coordinating: Artem Perepelitsyn	Computer systems and networks, Security of information and communication systems

Study cycle	Level of the module	Type of the module
Master	A	Full-time tuition.

Form of delivery	Duration	Langage(s)
Full-time tuition	One semester	English

Prerequisites	
Prerequisites:	**Co-requisites (if necessary):**
Applied Cryptography; Computer Systems Design Technologies; Finite State Machines	Foundations of Dependability and Security System and Network Security and Resilience

Credits of the module	Total student workload	Contact hours	Individual work hours
4	108	36	72

Aim of the module (course unit): competences foreseeen by the study programme		
The aim of module is to study of hardware implementation of pseudo-random number generators, true-random number generators, physical unclonable functions and some cryptographic primitives. The study also provides an ability to prevent the side channel attacks to FPGA system.		
Learning outcomes of module (course unit)	**Teaching/learning methods**	**Assessment methods**
At the end of course, the successful student will be able to: 1. Implement the PRNG based on LFSR in FPGA.	Interactive lectures, Learning in laboratories, Just-in-Time Teaching	Module Evaluation Questionnaire

2. Implement the PRNG based on NLFSR in FPGA.	Interactive lectures, Learning in laboratories, Just-in-Time Teaching	Module Evaluation Questionnaire
3. Implement the PRNG based on cellular automata in FPGA.	Interactive lectures, Learning in laboratories, Just-in-Time Teaching	Module Evaluation Questionnaire
4, 5. Implement the PUF and TRNG in FPGA.	Interactive lectures, Learning in laboratories, Just-in-Time Teaching	Module Evaluation Questionnaire
6. Develop the units of GF-arithmetic in FPGA.	Interactive lectures, Learning in laboratories, Just-in-Time Teaching	Module Evaluation Questionnaire
7. Construct the cryptographic primitives for block cipher in FPGA.	Interactive lectures, Learning in laboratories, Just-in-Time Teachingy	Module Evaluation Questionnaire
8. Propose the way of prevention of side channel attacks to FPGA system.	Interactive lectures, Learning in laboratories, Just-in-Time Teaching	Module Evaluation Questionnaire
9. Propose the way of hardware backdoors detection.	Interactive lectures, Learning in laboratories, Just-in-Time Teaching	Module Evaluation Questionnaire

Themes	Contact **work hours**							**Time and tasks for individual work**	
	Lectures	Consultations	Seminars	Practiacl work	Laboratory work	Placements	**Total contact work**	**Individual work**	**Tasks**
1.1. Linear Feedback Shift Registers. 1.1.1 Subject, goal and objectives of the study module 1.1.2. The structure and content of the discipline and a guideline (roadmap) for its study. 1.1.3. Discipline positioning in the study module. 1.1.4. Recommended references. 1.1.5. Theory of PRNG. 1.1.6. Theory of LFSR.	2							7	1.1.7. Implementation of PRNG based on LFSR in FPGA.
1.2. Non-Linear Feedback Shift Registers. 1.2.1. Theory of NLFSR	2							8	1.2.2. Implementation of PRNG based on NLFSR in FPGA.
1.3 Cellular automata. 1.3.1 Theory of CELLULAR automata.	2							9	1.3.2. Implementation of PRNG based on cellular automata in FPGA.
2.1. Physical unclonable functions. 2.1.1. Theory of PUF.	3							8	2.1.2. Implementation of PUF in FPGA.
2.2. True random number generators. 2.2.1. Theory of TRNG.	3				4			8	2.2.2. Implementation of TRNG in FPGA.

3.1. GF Implementation. 3.1.1. Theoretical basis of GF.	2							8	3.1.2. Implementation of GF multiplier in FPGA.
3.2. Advanced encryption standard implementation. 3.2.1. Theory of AES.	2				4			8	3.2.2. Implementation of AES in FPGA.
4.1. Side channel attacks. 4.1.1. side channel attack classification.	2				4			8	4.1.2. Considering of side channel attacks to FPGA system.
4.2. FPGA backdoors. 4.2.1. FPGA backdoor classification.	2				4			8	4.2.2. Considering of backdoors in FPGA system.
Iš viso	**20**				**16**			**72**	

Assessment strategy	Weight in %	Deadlines	Assessment criteria
Lecture activity, including fulfilling special self-tasks	30	7,14	85% – 100% Outstanding work, showing a full grasp of all the questions answered. 70% – 84% Perfect or near perfect answers to a high proportion of the questions answered. There should be a thorough understanding and appreciation of the material. 60% – 69% A very good knowledge of much of the important material, possibly excellent in places, but with a limited account of some significant topics. 50% – 59% There should be a good grasp of several important topics, but with only a limited understanding or ability in places. There may be significant omissions. 45% – 49% Students will show some relevant knowledge of some of the issues involved, but with a good grasp of only a minority of the material. Some topics may be answered well, but others will be either omitted or incorrect. 40% – 44% There should be some work of some merit. There may be a few topics answered partly or there may be scattered or

			perfunctory knowledge across a larger range. 20% – 39% There should be substantial deficiencies, or no answers, across large parts of the topics set, but with a little relevant and correct material in places. 0% – 19% Very little or nothing that is correct and relevant.
Learning in laboratories	50	7,14	85% – 100% An outstanding piece of work, superbly organised and presented, excellent achievement of the objectives, evidence of original thought. 70% – 84% Students will show a thorough understanding and appreciation of the material, producing work without significant error or omission. Objectives achieved well. Excellent organisation and presentation. 60% – 69% Students will show a clear understanding of the issues involved and the work should be well written and well organised. Good work towards the objectives. The exercise should show evidence that the student has thought about the topic and has not simply reproduced standard solutions or arguments. 50% – 59% The work should show evidence that the student has a reasonable understanding of the basic material. There may be some signs of weakness, but overall the grasp of the topic should be sound. The presentation and organisation should be reasonably clear, and the objectives should at least be partially achieved. 45% – 49% Students will show some appreciation of the issues involved. The exercise will indicate a basic understanding of the topic, but will not have gone beyond this, and there may well be signs of confusion about more complex material. There should be fair work towards the laboratory work objectives. 40% – 44% There should be some work towards the laboratory work objectives, but significant issues are likely to be neglected, and there will be little or no appreciation of the

			complexity of the problem. 20% – 39% The work may contain some correct and relevant material, but most issues are neglected or are covered incorrectly. There should be some signs of appreciation of the laboratory work requirements. 0% – 19% Very little or nothing that is correct and relevant and no real appreciation of the laboratory work requirements.
Module Evaluation Quest	20	8,16	The score corresponds to the percentage of correct answers to the test questions

ПРИЛОЖЕНИЕ Б. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Дисциплина включает четыре модуля:

Модуль 1: Исследование особенностей аппаратной реализации генераторов псевдослучайных чисел.

Модуль 2: Исследование аппаратной реализации физической криптографии на примере физически неклонируемых функций и генераторов истинно случайных чисел.

Модуль 3: Исследование методов аппаратной реализации блочных шифров и других криптопримитивов.

Модуль 4: Исследование атак по сторонним каналам на аппаратные реализации криптомодулей и встроенных систем.

По каждому из модулей предусмотрены практические занятия, семинары и дан список рекомендуемой литературы.

Методы оценки

По окончании курса проводится 90-минутный экзамен.

Отчетность по дисциплине включает отчеты по каждому виду практического занятия, а также экзамен, который включает типовые вопросы и задачи.

СОДЕРЖАНИЕ

Перепелицин Артем Євгенович

БЕЗПЕЧНІ І РЕЗІЛЬЄНТНІ СИСТЕМИ НА ПРОГРАМОВНОЇ ЛОГІКИ
Практикум
(російською мовою)

Редактор Харченко В.С.

Комп'ютерна верстка
А.Є. Перепелицин

Зв. план, 2017
Підписаний до друку 25.01.2017
Формат 60x84 1/16. Папір офс. №2. Офс. друк.
Умов. друк. арк. 21,89. Уч.-вид. л. 22,31. Наклад 100 прим.
Замовлення 2. Ціна вільна

Національний аерокосмічний університет ім. М. Є. Жуковського
"Харківський авіаційний інститут"
61070, Харків-70, вул. Чкалова, 17
http://www.khai.edu

Віддруковано ФОП Лисенко І.Б.
61070, Харків-70, вул. Чкалова, 17, моторний корпус, к. 147
Свідоцтво про внесення суб'єкта видавничої справи в державний реєстр
видавців, виготовлювачів і розповсюджувачів видавничої продукції
ДК №2607 от 11.09.06 р.