ПРИМЕНЕНИЕ ДИОДОВ-ГЕНЕРАТОРОВ ШУМА В СРЕДСТВАХ КРИПТОГРАФИЧЕСКОЙ ЗАЩИТЫ ИНФОРМАЦИИ

В.В. Буслюк, БрГТУ, г. Брест;
В.А. Тюнин, П.Н. Лаптев, ОКБ САПР, г. Пенза;
С.И. Ворончук, И.В. Лешкевич, Частное научно-исследовательское предприятие «СКБ «Запад», г. Брест

Введение

Физические источники шума находят применение в различных областях техники, в том числе в программно-аппаратных средствах защиты информации.

Шум, генерируемый физическими источниками, может быть использован, например, для маскирования информативных побочных электромагнитных излучений и наводок (ПЭМИН) от средств вычислительной техники, исключающего возможность перехвата ПЭМИН за пределами контролируемой зоны путем создания помех в широкой полосе частот.

Другое возможное применение физического шума в средствах защиты информации – использование случайных битовых последовательностей для генерации ключевой информации в средствах криптографической защиты информации. Поскольку в современной криптографии алгоритмы шифрования считаются общеизвестными, а безопасность информации основывается на правильном выборе ключа шифрования (принцип Керкгоффса), то задача качественной генерации ключевой информации является очень важной.

1 Аппаратная реализация

Схема включения шумовых диодов, приведенная на рисунке 1, соответствует рекомендуемой в [1]. В качестве источника опорного напряжения *Uоп* величиной 15,5 ± 0,5 мВ был выбран интегральный источник напряжения (DA2), с установленным на его выходе резистивным делителем (R4, R6). Использование интегральной микросхемы, прецизионных резисторов, низкое выходное сопротивление полученного источника опорного напряжения (при высоком входном сопротивлении компаратора) позволяют добиться достаточной стабильности параметров, высокой помехоустойчивости, низкой стоимости и хорошей повторяемости схемы в целом.

2 Оценка качества случайных битовых последовательностей

Для получения случайных цифровых последовательностей сигнал $U_{цифр}$ с выхода компаратора (рисунок 1) подавался на микроконтроллер Atmel ATSAM3U, после чего полученный поток битов сохранялся в файл с целью дальнейшей оценки качества случайной последовательности.

В настоящее время наибольшей популярностью пользуются два набора тестов оценки качества случайных последовательностей – пакет NIST американского Национального института стандартов и технологий (http://csrc.nist.gov/groups/ST/toolkit/rng/documentation_software.html) и тест DIEHARD Джорджа Марсальи (http://stat.fsu.edu/pub/diehard/).

Оценка качества цифрового шума выполнялась путем анализа сравнительно длинных (более 1 Мбит) последовательностей набором тестов NIST.

Пакет NIST состоит из 15 статистических тестов, целью которых является определение меры случайности двоичных последовательностей, порожденных генераторами случайных (псевдослучайных) чисел. Эти тесты основаны на различных

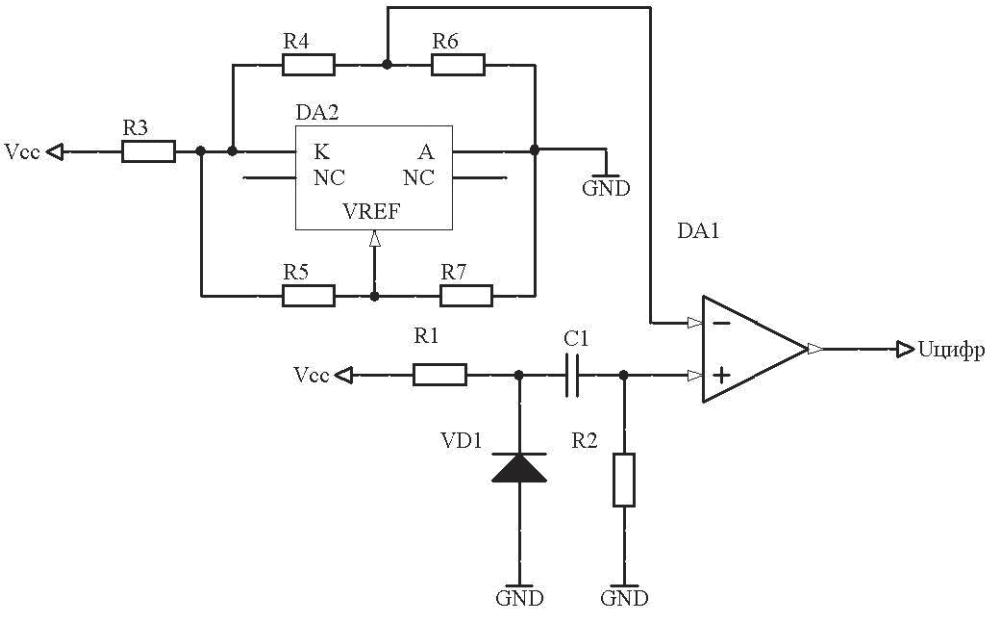

Рисунок 1 – Практическая схема включения шумовых диодов

статистических свойствах, присущих только случайным последовательностям. Обзор тестов приведен в [2], подробное описание в [3]. Пользователю доступна возможность настройки отдельных тестов, однако, в данном эксперименте пакет тестов был использован с настройками «по умолчанию».

Проведенные тесты дали следующие результаты – случайная последовательность успешно прошла 13 тестов, однако два теста не были пройдены. Тесты, выявившие отклонение исследуемой последовательности от случайной, – тест кумулятивных сумм (Cumulative Sums (Cusum) Test) и частотный побитовый тест (Frequency (Monobit) Test). На основании описаний данных тестов можно сделать вывод о том, что неуспешный результат вызван дисбалансом количества единиц и нулей в исследуемой последовательности. Эта гипотеза легко проверятся простым подсчетом микроконтроллером количества нулей и единиц в исследуемой последовательности, который действительно выявляет отличие математического ожидания от 0,5 в последовательности, которая предположительно должна быть случайной (и, соответственно, содержать в идеальном случае равное количество единиц и нулей). Такое отклонение от ожидаемого результата не является необычным и его можно объяснить неидеальностью элементов электрической схемы – самих шумовых диодов, резисторов, источника опорного напряжения.

3 Пути повышения качества случайных цифровых последовательностей

Для того, чтобы улучшить качество случайной последовательности (добиться равномерного распределения), необходимо провести некоторые дополнительные преобразования. В простейшем случае можно, например, складывать по модулю 2 (xor) соседние биты в исходной последовательности. Если входные данные независимы, и имеют отклонение от равномерного распределения ε, то последовательность на выходе будет иметь отклонение 2ε2. Также возможно использование схемы фон Неймана (von Neumann's debiasing scheme) [4], суть работы которой состоит в том, что из исходной последовательности берется пара битов и, если биты различаются, в выходную последовательность добавляется первый из них, а если биты одинаковы, то они отбрасываются. Если исходная последовательность является стационарной, то последовательность на выходе схемы фон Неймана характеризуется дискретным равномерным распределением, однако, следует помнить о том, что корректор фон Неймана генерирует последовательность с неравномерным распределением, если внутри исходной последовательности существуют циклы с периодом 2. Поэтому при практической реализации датчиков случайных чисел (ДСЧ) последовательность с выхода корректора рекомендуется подавать на вход криптографической хэш-функции, которая вносит в данную последовательность дополнительную энтропию.

Dan Boneh (Stanford University) в своих лекциях по криптографии (https://class.coursera.org/crypto-preview/class/index) рекомендует использовать для обработки «сырой» случайной последовательности с неравномерным распределением сеть Фейстеля [5]. Учитывая, что многие стандартизированные блочные шифры (DES, ГОСТ 28147-89) построены на основе сети Фейстеля, а разработчики программных (программно-аппаратных) средств защиты информации очень часто уже имеют в своем распоряжении библиотеки (модули, IP-ядра), реализующие указанные алгоритмы шифрования, видится разумным использование блочных шифров для улучшения характеристик последовательностей, полученных от физических ДСЧ.

Выводы

На основе рекомендаций [1] разработчиков шумовых диодов, рассмотрены возможности применения этих приборов в практических схемах средств защиты информации. Дальнейшего исследования требуют варианты обработки случайных цифровых последовательностей – сравнение эффективности этих методов (получение максимально качественного шума) и их быстродействия.

Литература:

1. Буслюк, В.В. Режимы применения кремниевых генераторных диодов для создания широкополосного шума / В.В.Буслюк, С.И. Ворончук, И.В. Лешкевич // Материалы и структуры современной электроники: сб. науч. тр. 5-й Междунар. науч. конф., Минск, 10-11 окт. 2012 г. – Минск: БГУ, 2012. – С. 24-27.

2. Soto, J. Statistical Testing of Random Number Generators / J.Soto // [Electronic resource]. – Mode of access: http://csrc.nist.gov/groups/ST/toolkit/rng/documents/nissc-paper.pdf

3. Rukhin, A. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications / A.Rukhin, et al. // [Electronic resource]. – Mode of access: http://csrc.nist.gov/groups/ST/toolkit/rng/documents/SP800-22rev1a.pdf

4. Entropy Key // [Electronic resource]. – Mode of access: http://www.entropykey.co.uk/res/download/diagram-explanation.pdf

5. Feistel cipher / From Wikipedia, the free encyclopedia // [Electronic resource]. – Mode of access: http://en.wikipedia.org/wiki/Feistel_cipher

ПРИМЕНЕНИЕ СОВРЕМЕННЫХ ПРОТОКОЛОВ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ В EDI СИСТЕМАХ НА ПРИМЕРЕ ОНТОЛОГИИ ПРОТОКОЛА AS2

А.В. Агафонов, О.В. Линич, Научно-инженерное республиканское унитарное предприятие «Межотраслевой научно-практический центр систем идентификации и электронных деловых операций»

В настоящее время при проектировании информационных систем все чаще находят применение новейшие технологии и разработанные специально для них инструментальные средства. Одной из таких технологий является технология семантического Web и созданный язык OWL (Web Ontology Language) – язык описания онтологий для