МИНОБРНАУКИ РОССИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра общей физики

Исследование формирователя хаотических последовательностей на регистрах сдвига

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента (ки) 4 курса, 4022 группы Института физики

направления подготовки 03.03.02 «Физика»

Суздальцева Александра Сергеевича

Научный
руководитель,
д.т.н., профессор Л.С. Сотов

Заведующий кафедрой,
д.ф.-м.н., профессор А.А. Игнатьев

Саратов 2021 г.

Введение

Цель: разработка алгоритмов и программ для анализа хаотических последовательностей формирователя на сдвиговых регистрах

Задачи.

- Разработка алгоритмов для анализа формирователя
- Создание тест-программ
- Анализ формирователя случайных чисел по полученным результатам тест-программ

Объектом исследования был формирователь хаотических последовательностей на регистрах сдвига модельное отображение, которое определялось по формуле $X_{n+1} = \frac{f_g}{f}(M - X_n)mod(M)$. Данный формирователь был реализован на плате Atlys– это отладочный набор, основанный на ПЛИС Xilinx Spartan-6 LX45 FPGA.

Проводились практические исследования формирователя хаотических последовательностей на регистрах сдвига с помощью, созданных тест-программ на языке программирования Python.

К числу новых результатов относится: полученные новые данные о возможности использовать формирователя хаотических последовательностей на регистрах сдвига в системе защиты передачи информации.

Достоверность результатов обуславливается использованием современных и узконаправленных алгоритмов.

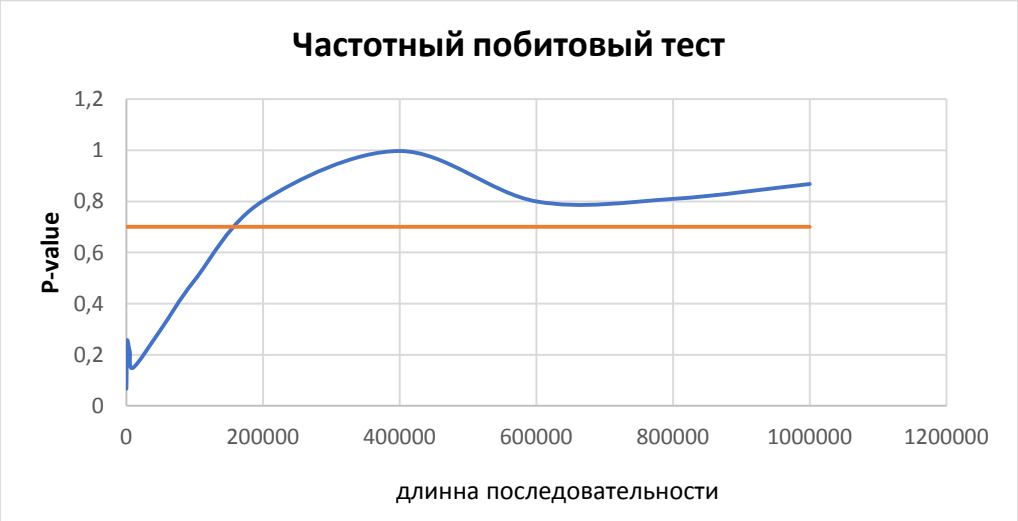
Основная часть

В ходе выпускной квалификационной работе были разработаны алгоритмы и тест – программы с помощью которых и проводились исследования формирователя хаотических последовательностей на регистрах сдвига.

1.1. Частотный побитовый тест.

Суть данного теста заключается в определении соотношения между нулями и единицами во всей двоичной последовательности. Цель — выяснить, действительно ли число нулей и единиц в последовательности приблизительно равны, как это можно было бы предположить в случае истинно случайной бинарной последовательности.

С помощью данного алгоритма создаём тест-программу на языке программирования Pythonи исследуем рассматриваемый формирователь случайных чисел. Получаем следующие данные.

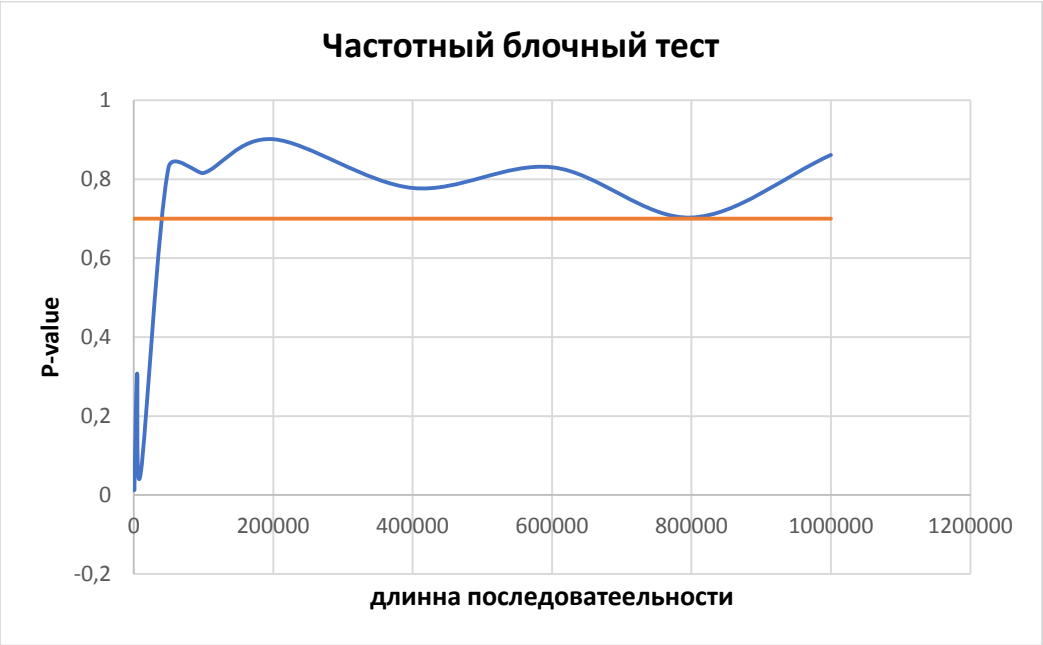


По результатам теста мы видим, что оптимальные значения вероятности истинно случайной последовательности от 0.7 до 1 принимают цепочки длинной 2000000 до 1000000, и это говорит о том, что нет ярко выраженной зависимости вероятности от длинны последовательности, что в свою очередь говорит о хорошем качестве исследуемого формирователя.

1.2. Частотный блочный тест.

Суть теста — определение доли единиц внутри блока длиной m бит. Цель — выяснить действительно ли частота повторения единиц в блоке длиной m бит приблизительно равна m/2, как можно было бы предположить в случае абсолютно случайной последовательности.

С помощью данного алгоритма создаём тест-программу на языке программирования Python и исследуем рассматриваемый формирователь случайных чисел. Получаем следующие данные.

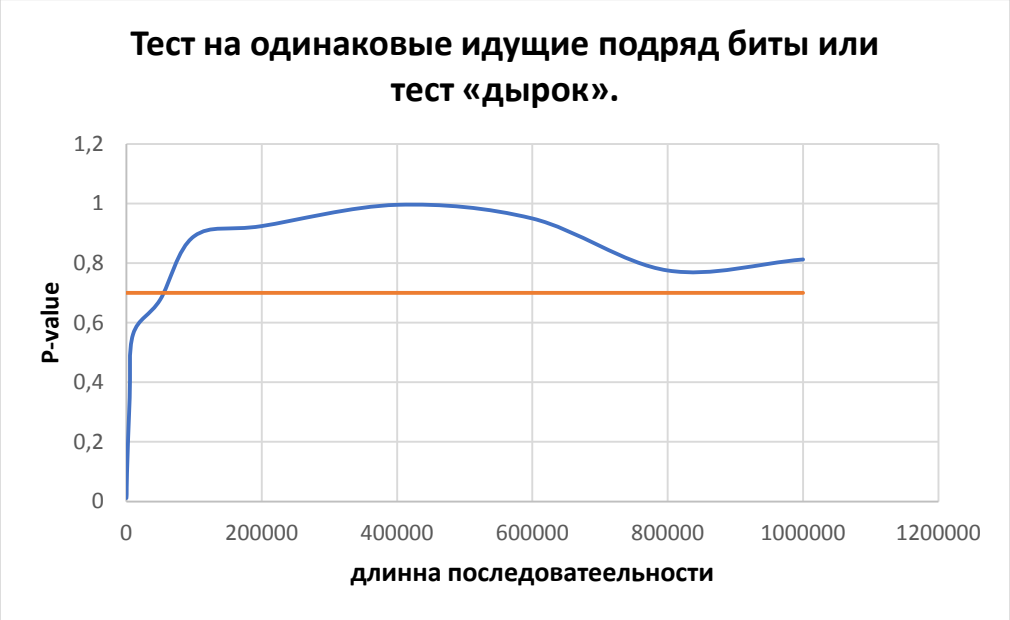


По результатам теста мы видим, что оптимальные значения вероятности истинно случайной последовательности от 0.7 до 1 принимают цепочки длинной 50000 до 1000000, и это говорит о том, что нет ярко выраженной зависимости вероятности от длинны последовательности, что в свою очередь говорит о хорошем качестве исследуемого формирователя.

1.3. Тест на одинаковые идущие подряд биты или тест «дырок».

Суть состоит в подсчёте полного числа рядов в исходной последовательности, где под словом «ряд» подразумевается непрерывная подпоследовательность одинаковых битов.Цель данного теста — сделать вывод о том, действительно ли количество рядов, состоящих из единиц и нулей с различными длинами, соответствует их количеству в случайной последовательности.

С помощью данного алгоритма создаём тест-программу на языке программирования Python и исследуем рассматриваемый формирователь случайных чисел. Получаем следующие данные.

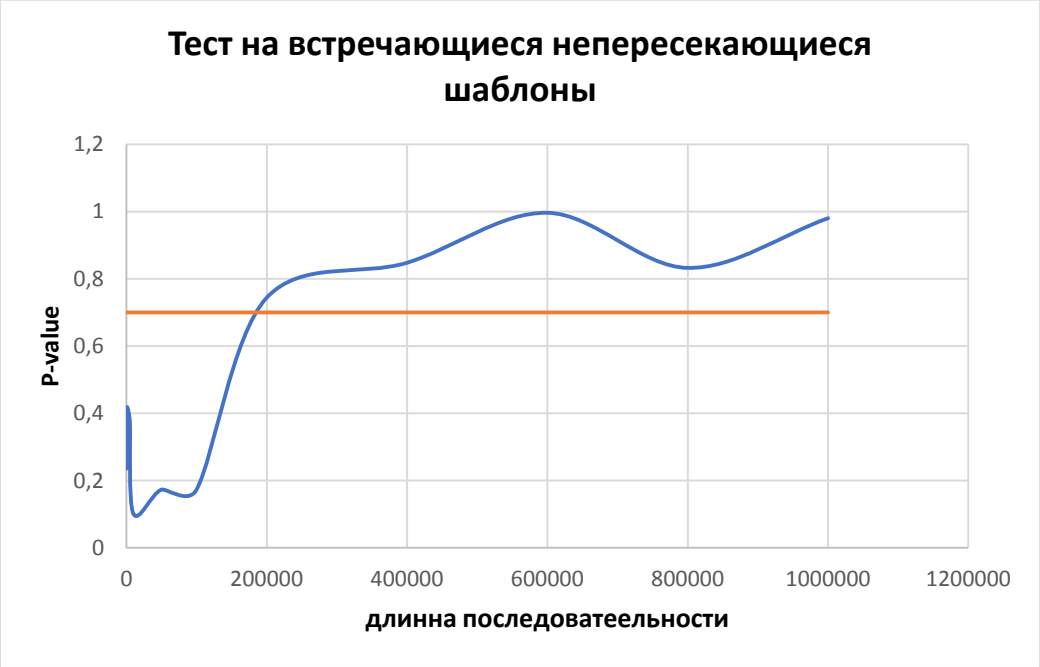


По результатам теста мы видим, что оптимальные значения вероятности истинно случайной последовательности от 0.7 до 1 для криптографии принимают цепочки длинной 1000000 до 1000000, и это говорит о том, что нет ярко выраженной зависимости вероятности от длинны последовательности, тов свою очередь говорит о хорошем качестве исследуемого формирователя.

1.4. Тест на совпадение неперекрывающихся шаблонов.

Суть данного теста заключается в подсчете количества заранее определенных шаблонов, найденных в исходной последовательности. Цель — выявить генераторы случайных или псевдослучайных чисел, формирующие слишком часто заданные непериодические шаблоны.

С помощью данного алгоритма создаём тест-программу на языке программирования Python и исследуем рассматриваемый формирователь случайных чисел. Получаем следующие данные.



По результатам теста мы видим, что оптимальные значения вероятности истинно случайной последовательности от 0.7 до 1 принимают цепочки длинной 2000000 до 1000000, и это говорит о том, что нет ярко выраженной зависимости вероятности от длинны последовательности, что в свою очередь говорит о хорошем качестве исследуемого формирователя.

1.5. Спектральный тест.

Суть теста заключается в оценке высоты пиков дискретного преобразования Фурье исходной последовательности. Цель — выявление периодических свойств входной последовательности, например, близко расположенных друг к другу повторяющихся участков.

С помощью данного алгоритма создаём тест-программу на языке программирования Python и исследуем рассматриваемый формирователь случайных чисел. Получаем следующие данные.

По результатам теста мы видим, что оптимальные значения вероятности истинно случайной последовательности от 0.7 до 1 для криптографии принимают цепочки длинной 200000 до 1000000, и это говорит о том, что нет ярко выраженной зависимости вероятности от длинны последовательности, что в свою очередь говорит о хорошем качестве исследуемого формирователя.

Заключение

На языке программирования Pyton разработаны и апробированы следующие тест – программы:

- Частотный побитовй тест
- Частотный блочный тест
- Тест на одинаковые идущие подряд биты или тест «дырок»
- Тест на совпадение неперекрывающихся шаблонов
- Спектральный тест

Исходя из проведенных исследований с использованием разработанных программ следует, что формирователь хаотических последовательностей на сдвиговых регистрах может быть использован в системах защиты информации, так как при длинах последовательности от 200000 значение Pvalue (вероятность что формирователь истинно случайный) устанавливается на уровне, соответствующем случайной последовательности с равномерным распределением.

Список использованных источников

1. Слеповичв И.И. Генераторы псевдослучайных чисел //Учебное пособие 2017.
2. Головин Д. В. Шахов В. Г. Исследование генератора псевдослучайной последовательности и возможностей его использования //Приборостроение, метрология, информационно-измерительныеные прибры и системы . 2002 . с.118-123.
3. Иванов М. А. Теория, применение и оценка качества генераторов псевдослучайных последовательностей / М. А. Иванов, И. В. Чугунков. М.: КУДИЦ-ОБРАЗ, 2003.