ПРОГРАММНЫЙ КОМПЛЕКС ДЛЯ СИНТЕЗА И АНАЛИЗА ДИСКРЕТНЫХ ПРЕОБРАЗОВАНИЙ И ПОСЛЕДОВАТЕЛЬНОСТЕЙ

С.В. Агиевич, А.Л. Костевич

Национальный научно-исследовательский центр прикладных проблем математики и информатики Белгосуниверситета
г. Минск, Беларусь

ВВЕДЕНИЕ

При проведении исследований в области защиты информации часто требуется генерировать, преобразовывать и обрабатывать потоки двоичных данных. Перечислим некоторые типовые задачи, с которыми мы сталкивались на практике:

A. На базе нескольких регистров сдвига с линейной обратной связью сконструировать составной генератор псевдослучайных двоичных последовательностей.
B. Синтезировать блочную криптосистему (см., напр., [3]), преобразования которой состоят в многократной замене и перестановке блоков входных двоичных слов.
C. Отбросить во входной последовательности каждый восьмой бит, а затем дважды повторить каждый пятый.
D. Выполнить поразрядное сложение байтов нескольких заданных файлов.
E. Определить число единиц в заданной двоичной последовательности и одновременно применить к последовательности статистический тест серий.

Для решения подобных задач нами разработан программный комплекс «Звезда». Комплекс предоставляет исследователю визуальные средства, с помощью которых можно проводить синтез и анализ дискретных преобразований и последовательностей. Опишем структуру и модель вычислений комплекса.

БЛОКИ

Базовыми компонентами комплекса являются **блоки** – функциональные элементы с несколькими входами и выходами. Входы и выходы являются двоичными векторами. Отрезки обрабатываемых последовательностей поступают на входы, результаты обработки сохраняются на выходах, способ обработки задается функцией блока.

Функциональное поведение блока уточняется его параметрами – размерностями входов и выходов, а также параметрами функции блока. Например, блок Перестановка имеет один вход размерности n и один выход размерности m. Функция блока ставит в соответствие вектору $(x_1, x_2, \ldots, x_n)$ вектор $(x_{\pi_1}, x_{\pi_2}, \ldots, x_{\pi_m})$, $1 \le \pi_i \le n$. Числа n, m и π_i являются параметрами блока.

Блоки делятся на:

1) источники – блоки без входов, осуществляющие чтение или генерацию последовательностей;

2) промежуточные блоки – блоки с входами и выходами, осуществляющие преобразования последовательностей;
3) приемники – блоки без входов, осуществляющие обработку и сохранение последовательностей.

В данной терминологии регистр сдвига является источником, блоки логических и арифметических операций – промежуточными блоками, статистический тест – приемником.

Между блоками можно устанавливать функциональные и параметрические связи.

Функциональная связь – это связь между выходом одного блока (блок A) и входом другого (блок B). Блок B при этом считается присоединенным к выходу блока A. К выходу блока A может быть присоединено любое количество блоков, вход блока B должен быть определен однозначно.

Параметрическая связь – это связь между параметрами двух и более однотипных блоков. Изменение пользователем параметров одного блока, влечет автоматическое изменение параметров всех связанных с ним блоков.

СЦЕНАРИИ

Выбирая блоки из предоставляемого «Звездой» набора, задавая параметры блоков и устанавливая между блоками функциональные и параметрические связи, исследователь создает **сценарий**. Создание и редактирование сценария осуществляется в «Звезде» визуальными средствами: перетаскиванием мышью нужных блоков на рабочую область сценария; проведением линий, задающих функциональные связи; объединением параметрически связанных блоков; изменением размеров и компоновкой элементов окончательных схем.

После создания и редактирования сценарий компилируется – проверяется готовность блоков к вычислениям, соответствие размерностей входов и выходов, однозначность направления потока данных и т. д. Например, ошибками компиляции являются:

- отсутствие в сценарии источников или приемников;
- наличие циклов (напр., блок B присоединен к выходу блока A и одновременно блок A присоединен к выходу блока B);
- нет блоков, присоединенных к выходу блока A;
- не определен один из входов блока B.

Если компиляция прошла успешно, то сразу начинается выполнение сценария – происходит генерация, преобразование и обработка последовательностей. Реализованная в «Звезде» модель вычислений основана на сетях Петри [1]. Дадим представление о данной модели.

Пусть блоки $B_1,\ldots,B_d$ присоединены к некоторому выходу блока A. С такими присоединениями удобно отождествить дуги, по которым передаются данные от A к B_i. Каждая из дуг может находиться в двух состояниях: «зажжена» и «погашена». Если дуга зажжена (погашена), то одновременно зажжен (погашен) соответствующий вход блока B_i. Выход A считается погашенным (зажженным), если погашены (зажжены) все дуги от A к B_i.

Перед выполнением сценария менеджер вычислений гасит все дуги. Затем менеджер находит блоки, у которых зажжены все входы (либо входы отсутствуют) и погашены все выходы (либо выходы отсутствуют). Если таких блоков нет, то выпол-

нение сценария прекращается. Если же подходящий блок обнаружен, то вызывается его функция, т. е. по входам определяются выходы, затем зажигаются все выходы и гасятся все входы. Таким образом, событие «дуга от A к B_i зажжена» означает, что выход A определен и еще не использован блоком B_i, а событие «дуга от A к B_i погашена» означает, что выход A либо не определен, либо не использован.

В комплексе реализовано управление длительными вычислениями. Выполнение сценария всегда можно прервать и перейти в состояние паузы. Далее можно 1) сохранить сценарий со всеми полученными данными, 2) вернуться к выполнению или 3) перейти к поствыполнению – получению предварительных результатов вычислений с последующим возвратом в состояние паузы.

КОНТЕЙНЕРЫ

Для удобства редактирования и управления сценариями предусмотрена возможность создания блоков-**контейнеров** или **листов**. Лист содержит вложенные блоки (в том числе и другие листы) с установленными между ними связями, а также списки входных и выходных разъемов. Входной (выходной) разъем отвечает за соответствие между входами (выходами) листа и входами (выходами) вложенных блоков. На основании списков разъемов осуществляется перенос данных с входов листа на входы вложенных блоков и с выходов вложенных блоков на выходы листа. Любую выделенную группу блоков сценария можно свернуть в лист – заменить контейнером.

Сценарий содержит ссылку на главный лист, в который вложены контейнеры уровня 1, в которые, в свою очередь, вложены контейнеры уровня 2 и т. д. Функциональные и параметрические связи можно задавать только между блоками, которые вложены в один контейнер.

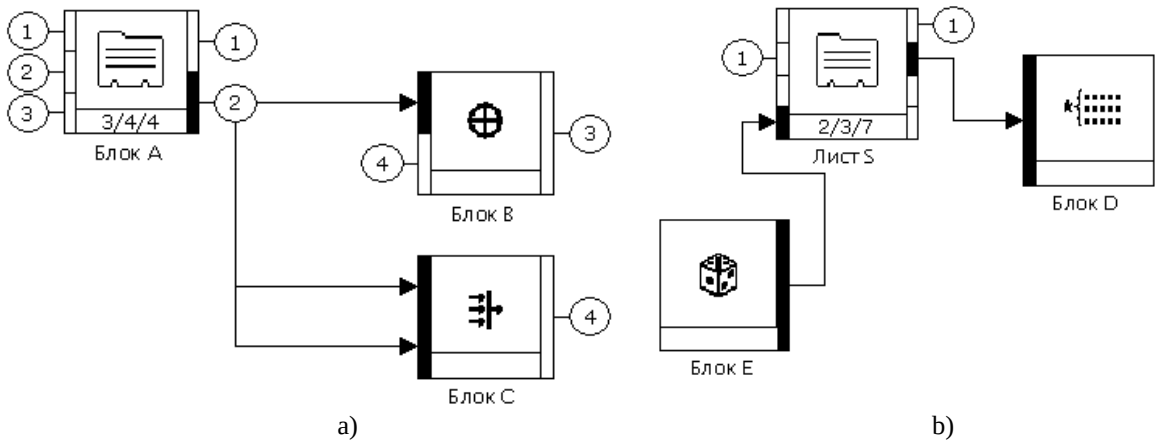

Рис. 1. Контейнеры

На рис. 1а изображены блоки A, B и C, которые вложены в лист S. В листе S открыты 4 входных разъема и 4 выходных (разъем изображается в виде кружка с номером внутри). Щелчок левой кнопкой мыши внутри кружка приводит к переходу на уровень вложенности листа S (см. рис. 1б). Здесь разъем S – это обычный вход или выход блока.

Обратим внимание на то, что ко 2-му выходу блока A присоединены блоки B и C. Кроме того, данный выход поддерживает 2-й выходной разъем листа S и к нему в дальнейшем могут быть подсоединены блоки, которые находятся на уровне вложенности S. После такого присоединения выходной разъем закрывать запрещено. То же касается и входных разъемов. В частности, нельзя закрыть 4-й входной и 2-й выходной разъемы блока S после задания связей, изображенных на рис. 1б.

Заметим также, что лист S поддерживает два разъема родительского листа. Это значит, что последовательности, поступающие на 1-й вход родительского листа, переадресуются на 2-й вход листа S, с которого переадресуются на 2-й вход вложенного листа A и т. д. Последовательности, формируемые на 1-м выходе листа A, переадресуются на 1-й выход листа S, а затем – на 1-й выход родительского листа.

ПРИМЕР

На рис. 2 изображен созданный средствами комплекса сценарий, предназначенный для анализа выходных последовательностей генератора Геффе [2].

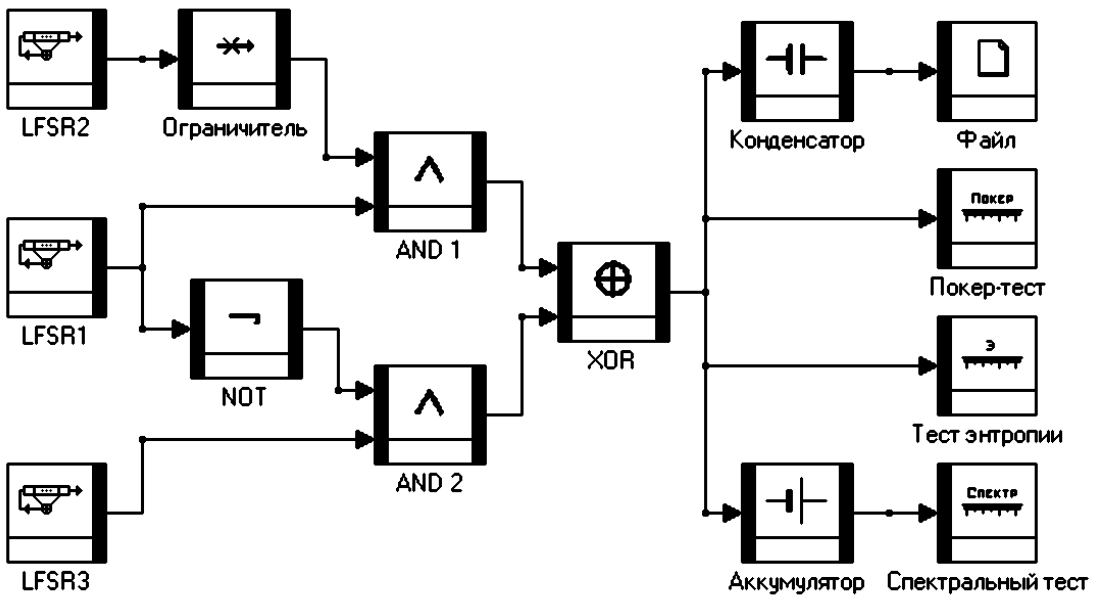

Рис. 2. Сценарий для анализа выходных последовательностей генератора Геффе

Блоки-источники LFSR1, LFSR2, LFSR3 сценария вырабатывают двоичные линейные рекуррентные последовательности $s_1^{(i)},\ldots,s_T^{(i)}, i=1,2,3$. Длина последовательностей T задается на блоке Ограничитель. С помощью блоков логического отрицания NOT, логического умножения AND1, AND2 и логического сложения XOR реализовано вычисление битов

$$g_t = s_t^{(1)} s_t^{(2)} \oplus (s_t^{(1)} \oplus 1) s_t^{(3)}.$$

По построению бит $s_t^{(1)}$ задает выбор между битами $s_t^{(2)}$ и $s_t^{(3)}$. Именно таким образом и функционирует генератор Геффе. Последовательность $g_1,\ldots,g_T$ сохраняется в

файловом приемнике Файл и обрабатывается блоками статистического тестирования Покер-тест, Тест энтропии, Спектральный тест.

Заметим, что в файловый приемник можно записать только целое число байтов. Поэтому в сценарий включен блок Конденсатор, который укрупняет поступающие на вход биты до байта. Блоки-приемники Покер-тест и Тест энтропии «на лету» обрабатывают биты g_t. С другой стороны, для выполнения спектрального теста требуется располагать всей последовательностью $g_1, \ldots, g_T$. Поэтому в сценарий включен блок Аккумулятор, накапливающий все поступившие на вход биты.

БЛАГОДАРНОСТИ

Авторы выражают благодарность коллегам В. Адамовичу, А. Афоненко, А. Акинфину, В. Галинскому, М. Кондратюк, А. Мартиневскому, А. Маслову, К. Мирановичу, А. Михадюку, О. Соловью, В. Щеглику, которые участвовали в проектировании, разработке и сопровождении программных систем на базе комплекса «Звезда».

ЛИТЕРАТУРА

1. Питерсон Дж. Теория сетей Петри и моделирование систем. М. 1984.
2. Geffe P. How to protect data with ciphers that are really hard to break // Electronics. 1973. V. 46, p. 99-101.
3. Schneier B. Applied Cryptography. N.Y. 1995.