ДИХОТОМИЧЕСКИЕ ПОСЛЕДОВАТЕЛЬНОСТИ И ИХ СВОЙСТВА

Кулаков Игорь Анатольевич

Random Art Labs Limited
chief@random-art.com

В статье закладываются теоретические основы построения стохастических систем. Рассматриваются фундаментальные свойства присущие двоичному представлению чисел натурального ряда. Производится обобщение свойств натурального ряда на другие, так называемые дихотомические последовательности. Дается анализ дихотомических последовательностей и их свойств. Приведены примеры алгоритмов их генерации. Отмечается многомерный, параметрический, существенно выраженный нелинейный характер способов генерации таких последовательностей, возможность достижения функциональной неразрешимости по существу и предельно высокой эффективности схем реализаций.

Данная статья написана по материалам доклада на 3-ей Центрально-европейской конференции по криптологии в Братиславе, TATRACRYPT 2003, прошедшей в Словакии с 26 по 28 июня 2003 года. С этой статьи начинается публикация результатов, полученных на протяжении многолетних исследований и экспериментальных разработок в области стохастических систем. На основе развития теории, разработан комплекс инструментальных средств и технологий, объединенный под общим названием - *рандомизационные системы*, предназначенных для создания стохастических и криптографических устройств и систем нового поколения, с учетом перспектив их развития и совершенствования криптографических атак [1]. Комплекс инструментальных средств охватывает в части и в целом все основополагающие разделы современной симметричной криптографии.

Полученные результаты в той или иной части затрагивают разделы теории чисел и алгебраических систем, положения функционального и статистического анализа, системный анализ и теорию динамических систем.

Схемы реализации отличаются простой исполнения, допускают параллельную одно и много разрядную, включая беспроцессорную аппаратную обработку на любых платформах вычислительных устройств, требуют небольшого объема памяти и рассчитаны на использование ключей переменной длины. Схемы реализации прозрачны для анализа, алгебраически сложны и замкнуты, носят высокодинамичный параметрический, скрытый от внешней среды простой или сложный, многомерный, существенно выраженный связанный и нелинейный характер. В итоге перечисленных факторов, они могут обладать достаточной для практических приложений статистической и функциональной надежностью. В программном исполнении по всем показателям не уступают наиболее совершенным образцам, а при аппаратной реализации отличаются высокой производительностью и малой себестоимостью.

Данные подтверждаются на основе натурного моделирования, пакета статистических тестов DIEHARD (G.Marsaglia, 1997, geo@stat.fsu.edu) и критериев FIPS PUB 140-1,2 (NIST, 2001, http://cs-www,ncsl.nist.gov/rng/rng2.html). Проведена верификация, а также предварительный статистический и функциональный анализ базовых криптографических алгоритмов.

Учитывая большой объем накопленного теоретического материала и практических разработок, предполагаем действовать последовательно от простого к сложному, по мере продвижения подтверждая представляемые результаты примерами и доказательствами.

На основе имеющихся технологий, в качестве первого шага, представляются так называемые *дихотомические генераторы*, по своим статистическим свойствам, функциональной сложности и производительности, превосходящие наиболее распространенные, функционирующие на основе регистров сдвига с линейной обратной связью (LFSR) линейные рекурентные генераторы. Например, при аппаратной реализации скорость функционирования таких генераторов, также как и простейших трехчленных LFSR, может быть соизмерима со скоростью выполнения одной операции *XOR* (сложения по модулю 2), вне зависимости от длины платформы генерации.

Дихотомические генераторы предназначены для формирования последовательностей специального вида, обладающих комплексом свойств присущим двоичному представлению чисел натурального ряда, так называемых *дихотомических последовательностей*. Дихотомические последовательности и их свойства рассматриваются особо.

Дихотомические генераторы могут быть параметрическими и многомерными, сложными. Они легко адаптируются под любые платформы вычислительных устройств, могут иметь любой, не меньший наперед заданного период повторения, способны образовывать структурные композиции любой сложности, могут быть функционально неразрешимыми по существу и при этом сохранять присущие составляющим их элементам бесповторные свойства.

Для дальнейшего изложения материала введем ряд необходимых понятий и терминов.

Последовательность $A = \{a_j\}$ определенную на множестве Π_A всевозможных значений ее элементов $a_j \in \Pi_A$ называют *моноциклической* или *МР-последовательностью*, если существует такое наименьшее число $k = T$, равное максимально возможному, так называемому *периоду повторения T*, при котором для всех $-\infty < i < \infty$ выполняются условия равенства $a_i = a_{i+k}$ ее элементов. *МР*-последовательности условно подразделяются на бесповторные и равноповторные последовательности. *МР*-последовательность в пределах периода повторения не содержащая равных между собой элементов называется *М- последовательностью*. Далее рассматриваются бесповторные последовательности. Другие типы последовательностей оговариваются особо.

Соответственно будем говорить, что объект (модель, система, оператор и др.) обладает *бесповторными свойствами*, если каким-либо реализациям его величин присущи свойства бесповторных последовательностей.

Неотъемлемыми элементами любой системы защиты являются генераторы псевдослучайных кодов и чисел. Основным классом *псевдослучайных*, далее по тексту случайных последовательностей формируемых на основе указанных генераторов, являются бесповторные последовательности с максимальным периодом повторения, так называемые последовательности максимального периода или *М–последовательности*. *М*-последовательности и их гене-

раторы составляют ядро симметричной криптографии.

К известным методам формирования последовательностей максимального периода относятся *линейный рекурентный* и *линейный конгруэнтный* методы [3, 4].

Как известно, в основе линейного рекурентного метода лежит моноциклический Γ_R оператор с постоянными коэффициентами $a_j \in \{0,1\}$, вида:

$$\Gamma_R\, u_i \equiv \sum_{j=1}^{n} \oplus (a_j \wedge u_{j+i}),$$

задаваемый линейным разностным уравнением $\Gamma_R u_i = 0$ порядка n в поле Галуа *GF(2)* с элементами поля $u_i \in \{0,1\}$, определенными для всех $-\infty < i < \infty$, решением которого являются так называемые линейные рекурентные *M*-последовательности *n*-битовых двоичных чисел u с периодом $T_n^R = 2^n - 1$ и значениями элементов, удовлетворяющих соотношению

$$u_{n+\,i+1} = \sum_{j=1}^{n} \oplus\, (a_j \wedge u_{j+I}) \qquad (i = 0,1,2,\ldots).$$

Для получения таких *M*-последовательностей используют *n*-разрядные *регистры сдвига* с множеством $J = \{j: a_j \neq 0, j = \overline{1,n}\}$ элементов *XOR* в цепи обратной связи, реализующих одноразрядные операции $\oplus$ сложения по модулю *2*. Операторы данного типа, *LFSR* иначе Γ_R-*генераторы*, обладают свойством, при котором формируемые на основе указанного регистра, составленные из n битов $u_j \in u$ целые числа u принадлежат интервалу $u \in [1, T_n^R]$, а их последовательность носит бесповторный характер и не содержит нулевых элементов.

Линейный рекурентный метод наиболее прост в исполнении. Это справедливо при условии фиксированной, оптимальной структурной реализации цепей обратной связи и если тривиальным решением $u \equiv 0$ приводящим к вырождению генератора, можно пренебречь. Первое из указанных ограничений объясняется сложностью вычисления коэффициентов a_j уравнения Γ_R-генератора, когда число полусумматоров в цепи обратной связи m_Γ минимально $m_\Gamma = min \sum_{j=1}^{n} a_j - 1$. В лучших случаях это число равно единице, но для некоторых n может быть очень большим. Второе из указанных ограничений связано с неполным интервалом изменения случайных чисел и проявляется тогда, когда на каждом очередном *i*-ом шаге итерации Γ_R-генератора текущее случайное число u принимается равным содержимому регистра сдвига и обусловлено отсутствием в формируемой таким образом последовательности нулевых элементов.

Вместе с этим, данный метод не обладает достаточной статистической и функциональной надежностью. Первое связано с относительно медленным изменением состояния регистра за счет сдвига на разряд и ограниченной одним битом, предельно малой скорости распространения влияния битов на каждом шаге итерации Γ_R-генератора, и вследствие этого сильно выраженной корреляции между элементами формируемой им последовательности. Для получения статистически надежной последовательности обычно ограничиваются одним или некоторой частью битов регистра, но это ведет к резкому снижению производительности генератора за счет увеличения числа элементарных операций приходящихся на один бит генерируемой последовательности и утрате ее бесповторных свойств. Второй из указанных недостатков обусловлен наличием достаточно простых методов восстановления любых из предшествующих и предсказания каждого из последующих значений элементов формируемой

Γ_R-генератором последовательности, что позволяет относительно легко раскрыть параметры генератора, независимо от числа бит используемых из регистра для формирования элементов этой последовательности [2, 5].

В отличие от рекурентного, линейный конгруэнтный метод задается уравнением [4, 5]:

$$x_i = a \cdot x_{i-1} + b \bmod g,$$

где x_o, a, b, g - начальное значение, множитель, приращение и модуль, при этом $\{x,a,b\}$ - неотрицательные целые числа, а модуль g - положительное целое число $g > x_o,\ g > a,\ g > b$.

Длина периода $T^C = g$ линейной конгруэнтной последовательности $\{x_i\}$ максимальна и равна g, тогда и только тогда, когда:

1) b и g - взаимно простые числа;
2) $c = a - 1$ кратно p для любого простого p, являющегося делителем g;
3) c кратно 4, если g кратно 4.

При соответствующем выборе значений коэффициентов и модуля уравнения, на основе линейного конгруэнтного метода можно генерировать достаточно хорошие в статистическом отношении конгруэнтные *M*-последовательности. Однако, как показывает анализ [2], такие последовательности, также как и линейные рекуррентные, не обладают достаточной функциональной надежностью, даже при существенном усечении части их значащих бит.

Кроме этого, по сравнению с рекурентным, для реализации конгруэнтного метода требуется наличие специализированного арифметического устройства или процессора, что существенно сказывается на конечной стоимости построенных на его основе генераторов, а наличие операций умножения и вычисления остатка от деления - на общей производительности соответствующих устройств. Метод работает достаточно хорошо при эффективной реализации арифметических действий и условии, что разрядность регистров используемых вычислительных устройств достаточна для выполнения машинных команд. Иначе требуется привлечение малоэффективной и сложной в исполнении арифметики больших чисел, что многократно сказывается на производительности и себестоимости подобных устройств.

Далее, учитывая универсальность двоичного представления данных, а также эффективность и простоту реализации двоичных команд, целесообразно ограничиться, так называемым линейным двоичным конгруэнтным методом, задаваемым уравнением, вида:

$$x_i = a \cdot x_{i-1} + b \bmod T_n^C,$$

позволяющим при $a \equiv 1 (mod\ 4)$ и нечетном b, получить *M*-последовательность двоичных чисел $x \in [0, T_n^C - 1]$ составленных из n значащих бит, с периодом $T_n^C = 2^n$.

Рассмотрим более подробно такие двоичные конгруэнтные последовательности. Начнем с последовательности чисел натурального ряда, задаваемой уравнением

$$x_i = x_{i-1} + 1 \bmod 2^n,$$

с начальным состоянием $x_0 = 0$.

Пронумеруем обычным порядком биты двоичной переменной x справа, налево, от младших числовых разрядов к старшим, начиная с единицы. Обращает на себя внимание, следующее частотное свойство данной двоичной последовательности (см. Колонку 2, Табли-

цы 1, Приложения). Период повторения T_1 первого бита равен 2^1, второго 2^2, третьего 2^3, k-го $T_k = 2^k$, а n-го 2^n.

Далее, рассмотрим двоичную конгруэнтную M-последовательность, к примеру, задаваемую уравнением

$$x_i = 133 \cdot x_{i-1} + 1 \mod 2^{16}.$$

Закон поведения битов этой последовательности (Колонка 3, Таблицы 1, Приложения), эквивалентен закону поведения битов натурального ряда, при этом для каждого k-го бита, i и $(i+T_k/2)$ элементы принадлежащие соседним полупериодам этой последовательности, посредством операции комплементации $\bar{}$ или операции *НЕ*, связаны соответствием

$$x_{ki} = \bar{x}_{k(i+Tk/2)}.$$

Указанные свойства присущи не только приведенным выше примерам, но и всем другим конгруэнтным последовательностям, для удобства обозначаемым далее символом C. Принимая во внимание сказанное, данные результаты допускают следующее естественное обобщение.

Двоичная последовательность $D = \{d_i: i = \overline{1, T_n}\}$ неотрицательных целых чисел d_i, составленных из n значащих бит $b_{ij} \in d_i$ $(j = \overline{1, n})$, называется *дихотомической* или *D-последовательностью*, если любая из входящих в ее состав $D_k \in D$ последовательностей D_k, образованных из $T_k = 2^k$ элементов исходной последовательности D путем усечения ее $n - k \in [0, n-1]$ старших $k < j \le n$ значащих бит b_{ij}, бесповторна в пределах периода T_k, т. е. для любой пары ее элементов $d_{ki} \in D_k$ и $d_{k(i+Tk)} \in D_k$, выполняются соответствия, вида:

$$d_{ki} = D_{ki} \mod T_k = d_{k(i+Tk)} = D_{k(i+Tk)} \mod T_k \quad (1 \le k \le n).$$

Указанный порядок будем называть *дихотомическим порядком*. Дихотомический порядок обладает иерархической структурой типа двоичного дерева, состоящей из уровней $k \in [1, n]$ и $e_k = T_n / T_k$ взаимно непересекающихся на этих k уровнях, независимых между собой, равных $D_{k1} = D_{k2} = \ldots = D_{kj}$ по значениям и числу T_k входящих в их состав элементов, так называемых *дихотомических классов* $D_{kj} \subseteq D \mod T_k$ $(j = \overline{1, e_k})$. Подклассы D°_{kj} и $D^\bullet_{kj}$, составленные $D^\circ_{kj} = \{D_{kji}: i = \overline{1, T_k/2}\}$ и $D^\bullet_{kj} = \{D_{kjl}: l = \overline{1 + T_k/2, T_k}\}$ из разделенных полупериодом $T_k/2$ элементов класса D_{kj}, будем именовать *самосопряженными* дихотомическими классами.

Дихотомический порядок и соответствующая ему D-последовательность называется *однородным* или *совершенным*, если все пары $\{b_{ki} \in B^\circ_k, b_{k(i+Tk/2)} \in B^\bullet_k\}$ битов i и $i + T_k/2$ $(i = \overline{1, T_k/2})$ порожденные каждым k битом сопряженных классов $\{B^\circ_k \subseteq D^\circ_k, B^\bullet_k \subseteq D^\bullet_k\} \in D_k$ класса D_k, комплементарно связаны между собой:

$$b_{ki} = \bar{b}_{k(i+Tk/2)}.$$

Указанное выше соответствие между элементами сопряженных классов D_k, будем называть *дихотомическим комплементом*.

Дихотомический порядок и последовательности будем называть *неоднородным*, если самосопряженные классы не являются комплементарными, т.е. среди всех пар $(b_{ki}, b_{k(i+Tk/2)})$ битов из самосопряженных классов $\{D^\circ_k, D^\bullet_k\}$, найдется хотя бы одна такая пара, что

$$b_{ki} = b_{k(i+Tk/2)}.$$

Двоичные величины, реализации которых обладают указанными дихотомическими свойствами, будем именовать *дихотомическими величинами*. Далее рассматриваются совершенные дихотомические последовательности и величины, и это специально не оговаривается.

Можно показать, что в поле вычетов $\mathbf{Z}/2^n$ по модулю 2^n, кардинальное число $card\, D_Z$ множества всех различных дихотомических последовательностей D_Z, приблизительно равно

$$card\, D_Z \approx 2^{n(n+3)/2\,-\,1}$$

и с точностью до изоморфизма определяется рекуррентным выражением:

$$d = \omega_0 + \sum_{k=1}^{n} \omega_k\, b_k\, 2^{k-1}\ mod\ 2^n,$$

относительно n-разрядной двоичной переменной d и составляющих ее бит $b_k \in d$, при любых нечетных коэффициентах $\omega_0 < 2^n$ и $\omega_k < 2^{(n-k)+1}$ и условии $\omega_1 \equiv 1 (mod\ 4)$, именуемым далее *дихотомическим уравнением в поле* $\mathbf{Z}/2^n$ или просто *линейным дихотомическим уравнением*.

Конгруэнтные последовательности C, формируемые на основе линейного двоичного конгруэнтного метода, являются частным случаем $C \subseteq D_Z$ дихотомических последовательностей D_Z. Кардинальное число $card\, C$ таких двоичных последовательностей, в сравнении с $card\, D_Z$, при больших n, мало и составляет величину, около $2^{3(n-1)}$.

Рассмотрим другие, наиболее характерные особенности дихотомических последовательностей и величин. Так, младшие биты дихотомических величин обладают сильно выраженными детерминированными, функциональными нерегулярными свойствами, тогда как характер изменения остальных битов, существенно зависит от значений коэффициентов упомянутого дихотомического уравнения. В зависимости от коэффициентов, старшие биты дихотомических величин и формируемые на их основе последовательности могут приобретать более или менее сильно выраженные нерегулярные, недетерминированные и хаотические свойства.

В силу существенно выраженного, нерегулярного поведения младших битов дихотомических величин и комплементации порождаемых ими дихотомических классов, составляющим битам этих величин, также присуща довольно сильная корреляция и функциональная зависимость между элементами реализаций этих величин. Между тем, в отличие от других методов, эта зависимость быстро убывает за счет экспоненциального роста периода $T_k = 2^k$ $(k \le n)$ от бита к биту и связанного с этим существенно выраженного, сложно предсказуемого, недетерминированного поведения старших битов и еще более сильно выраженного, чем выше разрядность n элементов генерируемой последовательности.

Другими словами можно сказать, что дихотомическим последовательностям и величинам присущи двойственные, детерминированные и недетерминированные, регулярные и нерегулярные комлиментарные функциональные и стохастические свойства.

При всех своих достоинствах, для формирования дихотомических последовательностей на основе упомянутого выше дихотомического уравнения в поле $\mathbf{Z}/2^n$ требуется привле-

чение больших вычислительных ресурсов. Даже сверхмощных 64-х разрядных и выше процессоров, явно недостаточно для эффективной реализации практически значимых генераторов данных последовательностей. Поэтому данное дихотомическое уравнение имеет больше содержательно-теоретический, нежели чем практический интерес.

Между тем, кардинальное число *card D* множества всевозможных различных дихотомических последовательностей *D*, равно:

$$card\, D = 2^{2^n - 1}.$$

Кардинальное число очень велико и существенно превосходит число всевозможных различных линейных рекурентных, конгруэнтных и подобных им дихотомических последовательностей. Кардинальное число указывает, на существование других алгебраических структур способных порождать *D*-последовательности. Вполне естественен вопрос. Существуют ли другие, более общие алгебраические структуры, чем поле вычетов $\boldsymbol{Z}/2^n$, на основе которых можно строить дихотомические последовательности обладающие приемлемыми статистическими свойствами и высокой функциональной сложностью и при этом допускающие организовать эффективную последовательную и параллельную обработку? Теоретические исследования в области динамических систем и результаты натурного моделирования, подтвердили существование таких алгебраических структур. В силу наличия у них, в целом или части, линейных свойств присущих полю $\boldsymbol{Z}/2^n$, с одной стороны и дискретных свойств, присущих полю $\boldsymbol{Z}/2$ с другой, а также линейных и дискретных свойств присущих полям $\boldsymbol{Z}/2^{n-1}$, $\boldsymbol{Z}/2^{n-2}$ и далее, данные алгебраические структуры получили название - *линейчатые структуры*. На основе этих структур разработан, так называемый *рандомизационный способ*, предназначенный для построения ординарных, одно и много параметрических, простых и сложных многомерных дихотомических генераторов с любым, не меньшим наперед заданного периодом повторения, обладающих достаточно хорошими статистическими свойствами и высокой функциональной сложностью, допускающих высокоэффективную последовательную и многоканальную параллельную, одно и много разрядную, включая беспроцессорную обработку на любых платформах вычислительных устройств.

В качестве примера реализации, в колонке 4, Таблицы 1 Приложения, представлена дихотомическая последовательность, полученная на основе одного из гипотетических вариантов реализации однопараметрического дихотомического 16-ти разрядного генератора, рассчитанного на 2-х разрядные регистры и ориентированного на последовательную программную реализацию. Ниже представлен исходный текст алгоритма на языке Си,

```
for( s = 3, i = 0; i < 8; i++ )    /* реализация последовательного дихотомического генератора */
{
        r = dBlock[i];             /*                выход дихотомического генератора                */
        g = r ^ ( ( s > r ) ? 1: 2 );  dBlock[i] = ( r  –  s ) & 3;  s ^= Pg[i];
        Pg[i] = g;                 /*          изменение внутреннего состояния генератора          */
}
```

при нулевых начальных условиях генерации дихотомической последовательности, задаваемых векторами dBlock и Pg, за исключением первого элемента параметра Pg[0], например, устанавливаемого равным 1.

Представленный алгоритм характеризуется смешанной алгебраической структурой в поле вычетов, отличается от известных методов способом задания признака результата и параметрическим представлением уравнений функционирования генератора.

Среди других, наиболее существенных особенностей поведения дихотомических генераторов параметрического типа, следует отметить наличие, обычно небольшого, переходного нестационарного участка τ, присущего широкому классу динамических систем, после которого указанные генераторы самостоятельно переходят в устойчивое состояние и далее всюду, в пределах периода 2^n, ведут себя как бесповторные генераторы псевдослучайных чисел. Например, такой участок (см. колонку 4, Таблицы 1), длиной равной *4*, имеет представленный выше параметрический генератор. В зависимости от способа реализации генератора и числа входящих в его состав параметров p, в лучших случаях это *2-3*, длина переходного участка τ может достигать величины $p \cdot n$. Этот недостаток не являются препятствием к практическому использованию представленных генераторов, так как при вполне определенных условиях он может быть легко устранен, либо в зависимости от максимально используемой длины статистической выборки L_S, в битах, принят равным $p \cdot log_2 L_S$.

Наличие переходных участков сопряжено с наличием притягивающего множества, так называемого *аттрактора*, характеризующего множество возможных начальных положений или состояний устойчивого равновесия генераторов, при которых эти генераторы носят невырожденный, бесповторный характер. Из-за определенного типа возмущений существующих при переходах через границы интервала длиной 2^n от одной дихотомической последовательности к другой, упомянутые генераторы способны обладать моноциклическими свойствами, с периодом $2^{\lambda \cdot n}$. Здесь λ - показатель моноцикличности, зависящий от начальных условий и структуры построения генератора, по величине не превосходящий $p+1$. Отсюда, в силу наличия переходных процессов и аттракторов, в отличие от упомянутых ранее строго бесповторных $(\tau = 0)$, далее будем различать *квазипериодические* $(\tau > 0)$ и *моноциклические* $(\lambda > 1)$ дихотомические последовательности и генераторы.

Дихотомические генераторы способны образовывать структурные композиции любой сложности. Формирование функционально сложных дихотомических последовательностей и структурных композиций осуществляется на основе комплексирования дихотомических генераторов. В таблице 2, Приложения, приведены результаты комплексирования (колонка 3), представленных выше параметрического дихотомического (колонка 2) и линейного конгруэнтного (колонка 1) 26-ти разрядных генераторов, задаваемых дихотомическими r и x переменными, соответственно, на основе линейчатой функции, вида:

$$z = r + 2 \cdot (h \oplus x) \bmod 2^{26},$$

где $\{h,z\}$ - двоичные 26-ти разрядные вектора, h - постоянная (модификатор), а z - результирующая дихотомическая величина.

Дихотомические генераторы допускают многоканальную параллельную обработку. В колонке 3, Таблицы 3 Приложения представлена дихотомическая последовательность, полученная на основе одного из вариантов реализации однопараметрического 13-ти канального

дихотомического 26-ти разрядного генератора, рассчитанного на 2-х разрядные регистры. Ниже представлен исходный текст алгоритма на языке Си,

```
for( s = 3, i = 0; i < 13; i++ )     /* реализация многоканального дихотомического генератора */
{
        r = dBlock[i];
        rBlock[i] = ((s ^ r) + Hg[i]) & 3;                  /*  выход дихотомического генератора  */
        g = r ^ ( (s > r) ? 1 : 2 );  dBlock[i] = ( r  –  s ) & 3;  s = Pg[i];
        Pg[i] = g;                  /*          изменение внутреннего состояния генератора          */
}
```

при нулевом векторе модификации Hg и нулевых начальных условиях генерации дихотомической последовательности, задаваемых векторами dBlock и Pg, за исключением первого элемента параметра Pg[0], устанавливаемого равным 1. Длина нестационарного участка представленного генератора, равна 10. В колонке 4, Таблицы 3 Приложения представлена последовательность, формируемая на основе двоичной векторной переменной Pg, характеризующая изменение внутреннего состояния генератора.

Указанные выше свойства дихотомических последовательностей позволяют используя простые и прозрачные для анализа способы, достаточно эффективно и просто получить статистически и функционально надежные для практических приложений последовательности равномерно распределенных чисел. При этом под статистической надежностью формируемых последовательностей в широком смысле, будем понимать равновероятность и независимость по Шеннону ее элементов, а в узком - устойчивость к общепризнанной системе статистических тестов, при всевозможных начальных условиях их генерации. С другой стороны, под функциональной надежностью формируемых последовательностей в широком смысле, будем понимать глубину перебора необходимую для восстановлением предшествующих и предсказанием последующих значений ее элементов, а в узком - устойчивость по отношению к восстановлению дихотомических свойств исходной для нее *D*-последовательности, при любом, ограниченным возможностями техники, доступном объеме выборки ее элементов.

В общем плане, формирование статистически надежных последовательностей может быть осуществлено путем распространения влияния старших битов на младшие, младших битов на старшие и перемешивания распространяемых битов, в целях придания старшим битам нерегулярных, а младшим битам существенно выраженных недетерминированных свойств, включая разрушение присущих дихотомическим последовательностям комплементарных свойств.

Достижение статистической надежности свидетельствует о высоком уровне рассеяния и перемешивания битов формируемой последовательности. В свою очередь, функциональная надежность формируемой последовательности обеспечивается за счет параметризации и многомерности используемых преобразований, а также путем сокрытия дихотомических свойств исходной для нее *D*-последовательности, за счет нелинейной и недетерминированной, сохраняющей необходимые статистические свойства, катенации (сцепления) битов ее элементов.

Как правило, практический результат оптимален, если задачи обеспечения статистической и функциональной надежности взаимно обусловлены и супераддитивны, почти всюду дополняют друг друга и гармонично сочетаются между собой.

Обобщая все выше сказанное, отметим следующие основные свойства присущие дихотомическим последовательностям и генераторам, а именно:

1. Частотные свойства битов, выражающиеся в экспоненциальном росте периода повторения от бита к биту.

2. Комплементарные свойства, связывающие полупериоды отрицанием НЕ.

3. Бесповторные свойства, обусловленные порядком катенации младших битов со старшими.

4. Лавинные свойства, обусловленные распространением влияния младших битов на старшие посредствам операции XOR.

5. Нелинейные свойства, обусловленные конъюнкцией AND младших битов со старшими.

6. Способность дихотомических последовательностей и генераторов образовывать, обладающие дихотомическими свойствами структурные композиции.

7. Функциональная неопределенность обусловленная сложным, параметрическим и многомерным характером уравнений генерации.

Предварительный анализ показывает, что параметризация и многомерность позволяет достичь функциональной неразрешимости уравнений генерации по существу, это с одной стороны. С другой на криптографическую силу дихотомических генераторов указывает лавинная, а при необходимости экспоненциальная скорость распространения влияния и существенно нелинейная катенация младших битов со старшими. При этом не в ущерб производительности, эта сила может быть существенно увеличена и регулируема за счет использования более сильных алгебраических структур и составляющих их примитивов.

В заключение следует отметить концептуальную связь дихотомических последовательностей, процессов и величин с динамическими системами, с реальными процессами протекающими в природе. На это указывают двойственность свойств и своеобразный, линейчатый характер дихотомических последовательностей, а также переходные процессы и аттракторы присущие сложным дихотомическим системам и генераторам, их феноменальная способность самостоятельно преодолевать нестационарные участки и приобретать бесповторные свойства.

Полагаю, данный материал будет интересен не только для криптографических приложений, но и для других естественно научных дисциплин.

Ключевые слова: Динамическая система. Аттрактор. Хаос. Стохастическая система. Симметричная криптография. Сетевая криптография. Генератор случайных чисел. Дихотомический порядок. Дихотомическая последовательность. Дихотомический генератор. Рандомизационный способ. Рандомизационный оператор. Рандомизационная система.

ЛИТЕРАТУРА

1. И.А. Кулаков, " Способ придания реальному объекту рандомизационных свойств и рандомизационная система", Заявка на международный патент.
2. E.F. Brickell and A M. Odlyzko, "Cryptanalysis: A Survey of Recent Results," Proc. of the IEEE, Vol.76, no.5, May 1988.
3. R. Lidl and H. Niederreiter, "Finite Fields," Encyclopedia of Mathematics and its Applications, v.20, Addison-Wesley, 1983..
4. D. Knuth, The Art of Computer Programming, Volume.2, Seminumerical Algorithms, 2nd edition Addison-Wesley, 1981.
5. B. Schneier, APPLIED CRYPTOGRAPHY. Protocols, Algorithms, and Source Code in C, John Wiley & Sons, Inc, 1996.

Appendix

Table 1. **Congruential & Dichotomic Sequences**

1	2	3	4
001	00000001	0000000000000001	1010101010101001
002	00000010	0000000010000110	0011001100110010
003	00000011	0100010110011111	0000011100000111
004	00000100	0010101110011100	0110100100110100
005	00000101	1010100000001101	1111111100101101
006	00000110	0100111011000010	1001101001000110
007	00000111	1110101011001011	0010001111101011
008	00001000	1111101101111000	1000101010101000
009	00001001	1010010101011001	1101001100000001
010	00001010	1110011100111110	1100110110101010
011	00001011	0010001100110111	1010001111001111
012	00001100	0100101110010100	1001001100011100
013	00001101	0100001111100101	1001100010100101
014	00001110	0100010111111010	0111111011111110
015	00001111	0101101011100011	0010011111000011
016	00010000	0011011111110000	1011111011110000
017	00010001	0000111110110001	0010101010111001
018	00010010	0010011011110110	0100111100000010
019	00010011	0011110111001111	1110100110010111
020	00010100	0001110010001100	1101100010000100
021	00010101	1101010010111101	1000011000111101
022	00010110	1000011000110010	0110111101010110
023	00010111	1011011111111011	0101100110111011
024	00011000	1001010101101000	0101010110111000
025	00011001	1001111100001001	1001011011010001
026	00011010	1001111110101110	1010011000111010
027	00011011	1111010101100111	1010101110011111
028	00011100	0111111010000100	1010100000101100
029	00011101	1011101010010101	0000000101110101
030	00011110	1110111101101010	0000000001001110
031	00011111	0110001000010011	1000101111010011
032	00100000	1111001111100000	0000011111000000
033	00100001	1011001101100001	1101100100001001
034	00100010	0011000101100110	1100010111010010
035	00100011	1010100111111111	0000011001100111
036	00100100	0101000101111100	1101011000010100
037	00100110	0101010101101101	0000110010100110
039	00100111	0110000110100010	0111111011001011
040	00101000	1011100100101011	1100011111001000
041	00101001	0011001101011000	0101111011100001
042	00101010	1010110010111001	1110000000001010
043	00101011	1011110000011110	0100010110101111
044	00101100	1011101110010111	1001111000111100
045	00101101	0111010101110100	0101111000000101
046	00101110	0000010101000101	1100111010011110
047	00101111	1011110011011010	1010001100100011

Transient section

Table 2. **Complexation of Dichotomic Generators**

1	2	3
000000000000000000000000001	101010101010101010101001	010111101101100100110100 01
000000000000000000010000110	110011001100110011001100 10	100000001111101110010110 00
000000000001000101100111111	110000011100000111000001 11	011101011100111000000110 11
000010010000101011100111100	010011011001101001001101 00	111100111101010110010001 10
101100101010101000000011101	110010111111111111001011 01	100111010111101001010111 01
110100100101001110110000102	100100011010011010010001 10	101000100011000001111101 00
010001010111101010110010111	001001111100100011111010 11	011001101010010011101001 11
000110001011111011011110000	001000100110001010101010 00	101001111011010111100000 10
110110110010100101010110011	110110111111010011000000 01	110111100111000011100110 01
110111000111100111001111102	100010000011001101101010 10	100101010001000010000000 00
100010110100100011001101111	110010110010100011110011 11	011011011110100000000100 11
010111001101001011100101000	111000001110010011000111 00	111011100111000000000111 10
001110011001000011111001015	101000111110011000101001 05	011010101111010110100001 05
111010000101000101111110102	111111111001111110111111 10	011001000010110000110111 00
101100100101011010111000113	001110110100100111110000 13	000010111100110111101011 11
101001110000110111111100000	001010100010111110111000 00	001001000110010100101110 10
110010100100001111101100015	000101100100101010101110 01	001101101111010000000000 01
000101010100100110111101106	101111001001001111000000 10	010110110101000110110010 00
000011110100111101110011115	110000011111101001100101 11	011011001010101011010010 11
111101000100011100100011000	010110101111011000100001 00	101101111001011011101101 10
111010001111010100101111015	000101011110000110001111 01	011110111010011001100011 01
000001110110000110001100102	100000001101101111010101 10	001110111100100101101001 00
110101011010110111111110113	001110111101011001101110 13	010110110100101111100101 11
000000110110010101011010008	010111010001010101101110 00	000011111111100110101100102
110000111010011111000010019	111011000010010110110100 01	000111111000011011000010 09
101001100010011111101011106	111010100010100110001110 10	111000101000101011101100 00
010100101011110101011001117	010011000010101011100111 11	010111010111111100100000 11
111111000101111110100001004	001101100010101000001011 00	100000101011101111010011 10
000111011010111010100101015	111100101000000001011101 05	100000011111010000011101 05
011010111011101111011010102	000111011000000000010011 10	100000001101100101010011 00
111110001001100010000100113	000011011010001011110100 13	010100101100001001110111 11
001001110011110011111000006	101111101100000111110000 06	101110010001100101101010 10
011000101010110011011000019	000001001011011001000010 09	011101100010110101111100 01
010000111100110001011001106	001101000111000101110100 10	011010000010011110101110 00
001110010010101001111111115	100001101100000110011001 11	010011010011110000001110 11
101100110001010001011111004	111000101111010110000101 00	101101001111101110111001 10
000010011001010101011011015	011100101011001100110011 01	000110011011011101101111 01
111110101001100001101000102	111001110000001100101001 10	001010000010000110000101 00
001100010010111001001010113	101000011001111111011001 13	011110000001000111010001 11
100011010000110011010110008	101001111111000111110010 08	010101100010100100011000 10
010001111010101100101110019	101000100101011110111000 01	110111011101000010001110 01
001110111110111100000111106	101001001011100000000010 10	011010001010100010001000 00
001000110010111011100101117	111111100001000101101111 11	111100001000010010101100 11
010001110101110101011101004	110000001010011110001111 00	111110110011101110111111 10
000100111000000101010001015	111100000101011110000001 05	100000111000001110101001 05
001000100010111100110110102	000100101111001110100111 10	000000110110010010001111 00
110000101000011101010000113	011100011010100011001000 11	101000101100100011110011 11
000100000100110011110100008	011110111100111010110100 00	000100001000011000010110 10

Table 3. **Dichotomic Sequence & Multichannel Generator**

1	2	3	4
001	00000001	000000000000000000000000111	101010101010101010100101
002	00000010	101010101010101010011010	010101010101010101010100
003	00000011	111111111111111111101001	000000000000000000010011
004	00000100	010101010101010100100100	111111111111111111001101
005	00000101	101010101010101000011011	000000000000000010110101
006	00000110	101010101010100011101110	000000000000000001010000
007	00000111	101010101010100100100101	000000000000000111001111
008	00001000	101010101010110000011000	000000000000000110110001
009	00001001	101010101010001011111111	000000000000011010101001
010	00001010	101010101001011100110010	000000000001101001101100
011	00001011	101010101001101110010000001	000000000110101000101101 1
012	00001100	101010100110111111010110 0	000000011010001000101010 1
013	00001101	101010011010001100111001 1	000001101001101110000110 1
014	00001110	101001101001010000011001 10	000011010011010110110001 00
015	00001111	100110100110101011011111 01	001101001101010101110001 11
016	00010000	011010011011111110010000 00	110100110100000000111101 01
017	00010001	101001101001010101100101 11	010011010011111111000101 01
018	00010010	100110100110101011111010 10	001101001100000001000101 00
019	00010011	011010011110101111001110 01	110100110100000001110000 11
020	00010100	101001101110100000000101 00	010011010100001101101111 01
021	00010101	100110111110010110111010 11	001101000100110101010001 01
022	00010110	011010001101101011011111 10	110100110111010100110000 00
023	00010111	101001010010111010010101 01	010011010101000000111111 11
024	00011000	100110100110011001101010 00	001101010100110011000000 01
025	00011001	011011111101010111101011 11	110100000111111010000000 01
026	00011010	101001000000101011101000 10	010011110101000101000011 00
027	00011011	100110010100111111100100 01	001100010110000000100111 011
028	00011100	011111001000100110101110 0	110100010110111101110101 01
029	00011101	101100111111100100100000 11	010111000001000101011111 01
030	00011110	110000011001111001011101 10	000110110111000101110101 00
031	00011111	001011000101001100100011 01	110101111111110001011101 11
032	00100000	011110110010000001010000 0	010000010100101101110101 01
033	00100001	110111000000110100101001 11	011100111000010101010101
034	00100010	000110100001001001111111 010	011100011001110101000001 00
035	00100011	110111000100011111010010 01	011100000101010001111100 11
036	00100100	100101011001110000100001 00	001111010101011101001011 01
037	00100101	011000101110000100001110 11	110011010001010110000101 01
038	00100110	110101100011010100010011 10	011101001100010110011100 00
039	00100111	001011010000101101000001 01	010101111011010001011011 11
040	00101000	011100011101101010101110 00	010001010101000101010100 01
041	00101001	110001000011111111111011 111	011101111101000000011000 1
042	00101010	000100110111010101100100 10	010101010101111111001111 00
043	00101011	011001001000101001011000 01	010101110101000011110010 11
044	00101100	101100011100100110110011 00	010001010010001100010001 01
045	00101101	100001111100011111110100 11	001101010010110100010111 01
046	00101110	010110110111101100100001 10	110101010101010111010101 00
047	00101111	101011100000110001110111 01	010100011111000101000101 11
048	00110000	111000011100000111001000 00	000011110110110101110101 01