УДК 621.39

Мурр Пьер
Murr Pierre

ГЕНЕРАТОР ПСЕВДОСЛУЧАЙНОЙ ПЕРЕСТРОЙКИ РАБОЧЕЙ ЧАСТОТЫ НА БАЗЕ МИКРОКОНТРОЛЛЕРА

GENERATOR OF PSEUDO-RANDOM FREQUENCY HOPPING BASED ON MICROCONTROLLER

Аннотация. Разработана концепция построения ортогональных кодов псевдослучайной перестройки рабочей частоты (ППРЧ) большого периода и повышенным уровнем параметрической скрытности. Предложена структурная схема генератора кодов ППРЧ на базе генератора псевдослучайной последовательности. Приведен алгоритм быстрого и простого расчета чисел псевдослучайной последовательности.

Summary. A concept to build orthogonal codes for pseudo-random frequency hopping (FHSS) of large period and high level of security. Proposed the schema of generator code FHSS based on generator of pseudo-random sequence. Presented an algorithm for quick and sample calculation numbers of pseudo-random sequence.

В многоканальных системах связи с шумоподобными сигналами для расширения спектра сигнала чаще всего используются метод прямой последовательности (DSSS MA – Direct Sequencing Spread Spectrum Multiple Access) и метод скачкообразной перестройки частоты (FHSS MA – Frequency Hopping Spread Spectrum Multiple Access) [1]. Для реализации метода FHSS MA используется псевдослучайная последовательность (ПСП) чисел, которую вырабатывает генератор псевдослучайной перестройки рабочей частоты (ППРЧ) – генератор ППРЧ-кодов [2, с. 752–753]. До настоящего времени проблема построения такого генератора до сих пор не решена в полном объеме. Данный генератор обычно строится на основе генератора случайных чисел, который должен обладать следующими характеристиками: иметь равномерный закон распределения вероятностей, как можно больший период повторения значений, генерировать числа без пропусков и повторений в заданном диапазоне, обладать повышенным уровнем параметрической скрытности ППРЧ-кодов и быть простым в построении.

Многие известные методы генерирования псевдослучайной последовательности чисел решают проблему построения генератора ППРЧ-кодов частично и не удовлетворяют всем этим требованиям одновременно или базируются на сложных математических расчетах [3, 4], что замедляет работу генератора и для их работы необходима большая разрядность микропроцессора. Это значительно усложняет использование этих методов в генераторе ППРЧ-кодов, построенном с применением микроконтроллера. Однако в литературе отсутствует метод, который давал бы указанную возможность.

Целью данной статьи является построение алгоритма работы генератора ортогональных ППРЧ-кодов, который отличается от известных алгоритмов простотой математических расчетов и который может быть реализован в генераторах ППРЧ-кодов, построенных на базе микроконтроллера.

Для генерирования ПСП чисел с равномерным законом распределения вероятностей на практике широко используют метод Коробова [5, с. 57–59]. Согласно этому методу, каждое очередное число x_{i+1} генерируемой последовательности определяется по рекуррентной формуле вычетов

$$x_{i+1} = (qx_i) \bmod p, \qquad (1)$$

где p – большое простое число; q – параметр.

Целочисленный параметр q подбирается близким к числу $p/2$ из множества вида $q = 3^m$, где m – любое целое число. Начальное значение генерируемой последовательности x_0 – произвольное целое число.

Проведенные исследования генератора Коробова показали его практическую привлекательность для решения многих задач вероятностных расчетов, и в частности, для вычислений неберущихся интегралов на основе метода Монте-Карло. Например, для $p = 2027$, параметра $q = 3^6 = 729$ и начального значения параметра $x_0 = 1$, ПСП чисел $X = \{x_0, x_1, \ldots, x_i, \ldots, x_{N-1}\}$ длины $N = p - 1 = 2026$ имеет распределение, показанное на гистограмме рис. 1.

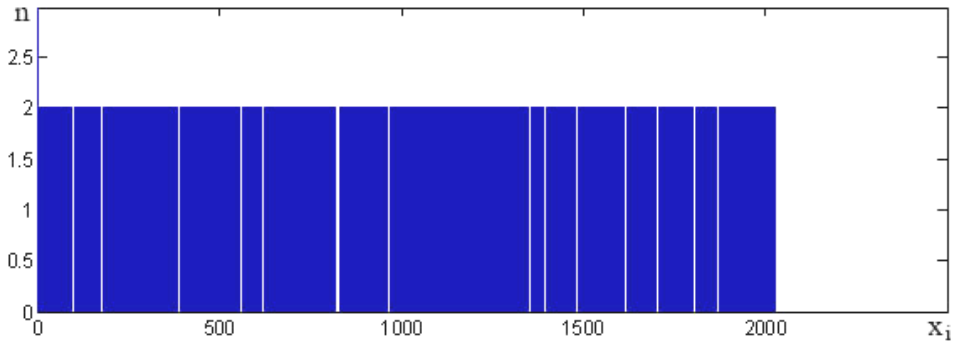

Рисунок. 1 – Гистограмма распределения ПСП чисел $X = \{x_0, x_1, \ldots, x_i, \ldots, x_{N-1}\}$ с параметрами генерации: $p = 2027$, $q = 3^6 = 729$, $x_0 = 1$

Из анализа этой гистограммы видно, что ПСП чисел $X = \{x_0, x_1, \ldots, x_i, \ldots, x_{N-1}\}$, генерируемая с параметрами: $p = 2027$, $q = 3^6 = 729$, $x_0 = 1$, имеет период $T = 1013 < p = 2027$, при этом числа x_i повторяются по 2 раза, а другие числа x_i вообще не генерируются.

Понятно, что применение генератора Коробова в генераторе ППРЧ-кодов является не целесообразным, так как он не обеспечивает генерацию чисел без пропусков и повторений в заданном диапазоне.

Исследования, проведенные автором, показали, что если в генераторе Коробова в качестве параметра q выбрать один из множества первообразных корней поля Галуа $\theta \in GF(p)$, например $\theta = 2$, то всегда будет формироваться ПСП чисел, гистограмма распределений которых приведена на рис. 2. В этом случае формируется ПСП чисел без пропусков и повторений и соблюдается идеальное равномерное распределение генерируемых чисел.

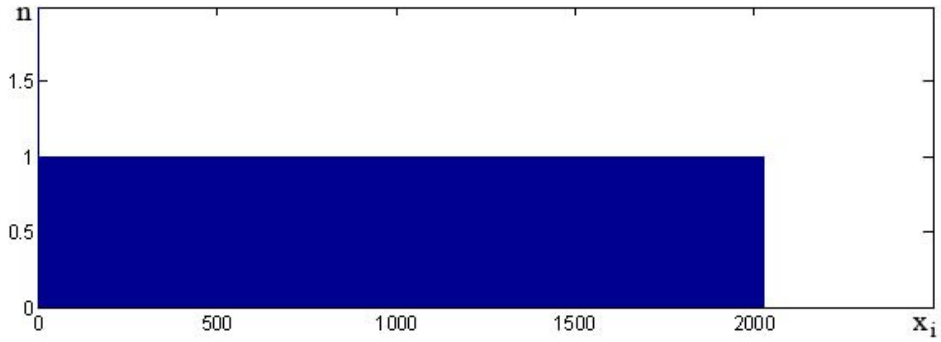

Рисунок 2 – Гистограмма распределения ПСП чисел $X = \{x_0, x_1, \ldots, x_i, \ldots, x_{N-1}\}$ с параметрами генерации: $p = 2027$, $q = \theta = 2$, $x_0 = 1$

Количество первообразных корней поля Галуа $GF(p)$ определяется по формуле $\varphi(p-1)$, где $\varphi(k)$ – функция Эйлера [6, с. 29–31], которая определена для всех целых положительных k и представляет собой число чисел ряда $0, 1, \ldots, k-1$ взаимно простых с k.

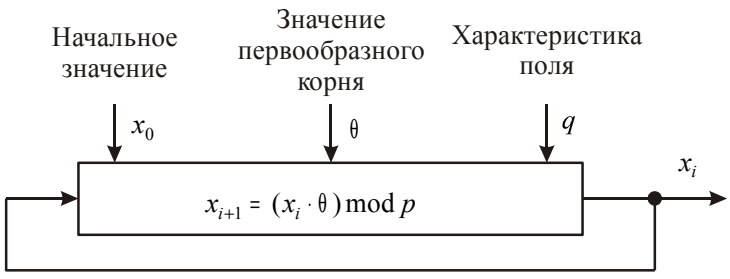

Рисунок 3 – Схема генератора ППРЧ-кода на основе рекуррентной последовательности

В рассматриваемом примере для характеристики $p = 2027$ поля $GF(p)$ существует всего $\varphi(2026) = \varphi(2 \cdot 1013) = 1012$ различных между собой первообразных корней.

Схема рекуррентного генератора ПСП чисел построенного по формуле (1) на основе регистра сдвига с обратной связью представлена на рис. 3.

Одним из важных свойств системы FHSS является уровень (структурной) параметрической скрытности ППРЧ-кода, которая включает понятие сложности разгадывания структуры ППРЧ-кода или правила и параметры его формирования. Отметим, что ППРЧ-коды, построенные на основе рекуррентной последовательности, имеют относительно низкий уровень параметрической скрытности [7, с. 333–402], которую будем оценивать (на основе анализа работы схемы рис. 3) величиной

$$\Psi = (p-1)\cdot\varphi(p-1), \tag{2}$$

где $x_0 \in \overline{1, p-1}$, а число различных корней θ равно $\varphi(p-1)$.

Другими словами, если использовать ПСП на основе линейной рекуррентной последовательности чисел в качестве ППРЧ-кода, то система FHSS будет иметь низкий уровень (2) защиты информации от несанкционированного доступа. Второй недостаток ППРЧ-кодов на основе рекуррентной последовательности состоит в том, что при формировании кодов большой длины (периода) всегда получаем N-ю последовательность ($N = p$), при этом может оказаться, что $N >> K$, где K – число частотных каналов в системе расширенного спектра FHSS.

Для увеличения уровня параметрической скрытности ППРЧ-кодов предлагается применить в основу их построения частотно-временные коды (ЧВК) большого объема [8]. Наиболее практически привлекательными, с точки зрения получения максимальных объемов кодовых слов, являются композиционные системы, получаемые путем объединения оптимальных систем частотно-временных кодов.

Композиционным частотно-временным кодом – $S(p)$-кодом над простым полем $GF(p)$ называют p-й код, каждое кодовое слово которого определяется правилом

$$S_k^{ry} = (ki^r + \nu) \bmod p, \quad i = \overline{0, p-1}, \tag{3}$$

где $k = \overline{1, p-1}$, $\nu = \overline{0, p-1}$, для каждого $r = \overline{2, p-2}$.

Базовым кодовым словом $S(p)$-кода будем называть кодовое слово S_k^{ry} с параметрами: $k = 1$, $r = p-2$, $\nu = 0$, т.е. слово $S_1^{p-2,0}$. Базовое кодовое слово $S(p)$-кода для различных значений характеристики поля p можно рассчитать заранее и хранить в памяти генератора. Например, для $S(7)$-кода базовое кодовое слово имеет вид $S_1^{5,0} = 0, 1, 4, 5, 2, 3, 6$. Кодовые слова вида $S_k^{p-2,0} = kS_1^{p-2,0}$, $k = \overline{2, p-1}$ назовем порождающими. Каждое порождающее кодовое слово порождает путем его циклических сдвигов по частоте $\nu = \overline{0, p-1}$ (3) оптимальную систему дискретно-частотных сигналов (ДЧ-сигналов) [9]. В табл. 1 построен композиционный $S(7)$-код и представлен в виде объединения оптимальных циклических по частоте подкодов. Порождающие кодовые слова (табл. 1) $S(7)$-кода выделены жирным шрифтом. Путем последовательной конкатенации кодовых слов $S(7)$-кода (табл. 1), построим 7-ю ПСП периода $T = 7^2 \cdot 6 = 294$ символа с идеальным равномерным распределением.

Непосредственно из определения (3) следует, что p-ичный $S(p)$-код имеет длину $N = p$, при этом его мощность

$$J_k(p) = p(p-1). \tag{4}$$

Таблица 1 – Структура минимаксного композиционного кода

ν / k	1	2	3	4	5	6
0	**0145236**	**0213465**	**0351624**	**0426153**	**0564312**	**0632541**
1	1256340	1324506	1462035	1530264	1605423	1043652
2	2360451	2435610	2503146	2641305	2016534	2154063
3	3401562	3546021	3614250	3052416	3120645	3265104
4	4512603	4650132	4025361	4163520	4231056	4306215
5	5623014	5061243	5136402	5204631	5342160	5410326
6	6034125	6102354	6240513	6315042	6453201	6521430

Отметим, что увеличение ансамбля сигналов всегда приводит к улучшению помехозащищенности системы передачи информации, к ее энергетической и структурной скрытности работы.

На рис. 4 предложена схема генератора ПСП на основе композиционных кодов степенных вычетов с фиксированным параметром $r = p - 2$.

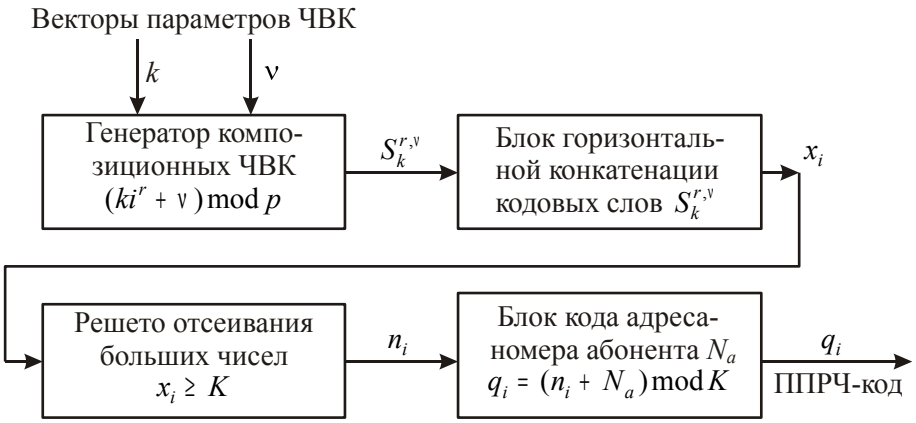

Рисунок 4 – Схема генератора ППРЧ-кода на основе кодов степенных вычетов

Для работы генератора композиционных ЧВК необходимо задать два ключевых вектора:

$$k = [k_1, k_2, \ldots, k_n, \ldots, k_{p-1}], \text{ из } (p-1)! \text{ векторов,}$$

$$\nu = [\nu_0, \nu_1, \ldots, \nu_m, \ldots, \nu_{p-1}], \text{ из } p! \text{ векторов.}$$

Блок горизонтальной конкатенации соединяет последовательно кодовые слова $S_k^{r,\nu}$ в единую ПСП, затем происходит (при необходимости) отсеивание больших чисел $x_i > K$, где K – число частотных каналов в системе FHSS. Наконец, блок кода адреса формирует ортогональный ППРЧ-код, по отношению к ППРЧ-кодам других номеров N_a абонентов системы FHSS. Из анализа работы схемы рис. 4 с учетом (4) следует, что период ПСП

$$T = p \cdot J_k(p) = p^2(p-1),$$

при этом уровень ее параметрической скрытности

$$\Psi = p!(p-1)!.$$

Принципиальная особенность предложенного способа состоит в том, что ПСП всегда является p–й, при этом p определяет собой максимальное число. Например, пусть в системе максимальное число $K = 200$. Выбираем ближайшее сверху подходящее простое число $p = 211$, тогда период последовательности $T \approx 9{,}3 \cdot 10^6$, а коэффициент параметрической скрытности равен практически бесконечности ($\Psi = 211!.210! = 2{,}36 \cdot 10^{798}$).

Вместе с тем, основная задача при разработке правил формирования хороших ПСП состоит в том, чтобы обеспечить простые и недорогие средства их генерирования.

Рассмотрим, каким образом можно упростить вычисления по основной формуле генерации ПСП (3). Известно, что

$$\begin{aligned}(a + b) \bmod c &= (a \bmod c + b \bmod c) \bmod c;\\ (a - b) \bmod c &= (a \bmod c - b \bmod c) \bmod c;\\ (a \cdot b) \bmod c &= (a \bmod c \cdot b \bmod c) \bmod c.\end{aligned} \quad (5)$$

Формулу (3) с учетом (5) можно представить в виде

$$\begin{aligned}S_k^{r,\nu} &= (ki^r + \nu) \bmod p = (ki^r \bmod p + \nu \bmod p) \bmod p =\\ &= ((k \bmod p \cdot i^r \bmod p) \bmod p + \nu \bmod p) \bmod p.\end{aligned} \quad (6)$$

В связи с тем, что $k = \overline{1, p-1}$ и $\nu = \overline{0, p-1}$ (3), то $k \bmod p = k$ и $\nu \bmod p = \nu$. Поэтому формулу (6) можно упростить

$$S_k^{r,\nu} = ((k \cdot i^r \bmod p) \bmod p + \nu) \bmod p. \quad (7)$$

Из анализа формулы (7), следует, что самой сложной операцией при программировании этого метода на микроконтроллере является

$$i^r \bmod p. \tag{8}$$

Сложность состоит в том, что на практике часто приходится разрабатывать системы генераторов ПСП с большими значениями i, r и p. Например для $i = 3000$, $r = 5379$ и $p = 5381$

$$3000^{5379} \bmod 5381 = 2.7 \cdot 10^{18703} \bmod 5381 = 1756.$$

Вычислить такие большие значения трудно даже на мощных современных компьютерах, не говоря о микроконтроллерах, где из-за малой разрядности регистров, решение такой задачи почти невозможно. К тому же многие языки программирования микроконтроллеров даже высокого уровня не содержат операций целочисленного возведения в степень. При этом программисты обычно осуществляют операцию возведения в степень путем многократного выполнения умножения в цикле

$$x^y = \underbrace{x \cdot x \cdot x \cdot x \cdot \ldots \cdot x}_{y}.$$

Для нашего примера, в случае вычисления 3000^{5379} придётся повторить операцию умножение 5378 раз, при этом на выполнение операции возведения в степень потребуется значительное время, а результат вычисления – $2.7 \cdot 10^{18703}$ займёт огромное количество ячеек памяти.

В литературе предлагается множество разных быстрых алгоритмов для решения задачи вычисления x^y с помощью операций умножения, один из наиболее известных – метод «русского крестьянина» [10, с. 503-505]. Покажем как применить этот метод для упрощения вычислений формулы (8).

Метод «русского крестьянина» предлагает эффективный и быстрый способ вычисления операции возведения в степень. Например, для вычисления $x^{10} \bmod p$ сначала представим x^{10} в следующим виде:

$$\begin{aligned} x^{10} &= x \cdot x \cdot x \cdot x \cdot x \cdot x \cdot x \cdot x \cdot x \cdot x = (x \cdot x)(x \cdot x)(x \cdot x)(x \cdot x)(x \cdot x) = x^2 \cdot x^2 \cdot x^2 \cdot x^2 \cdot x^2 = \\ &= (x^2 \cdot x^2)(x^2 \cdot x^2)(x^2) = \left(x^2\right)^2 \cdot \left(x^2\right)^2 \cdot x^2 = \left(\left(x^2\right)^2\right)^2 \cdot x^2, \end{aligned} \tag{9}$$

и мы получим тот же результат, но всего с помощью четырех операций умножения. Далее учитывая (5) можно представить наш пример в следующем виде

$$x^{10} \bmod p = \left[\left(\left(x^2 \bmod p\right)^2 \bmod p\right)^2 \bmod p \cdot x^2 \bmod p\right] \bmod p. \tag{10}$$

Используя выражения (9) и (10) предложим регулярный алгоритм для построения ПСП по формуле (3).

АЛГОРИТМ

Исходными данными для построения ПСП являются простое число p, целое число $r = \overline{2, p-2}$ и два произвольных целочисленных вектора $k = [k_1, k_2, \ldots, k_n, \ldots, k_{p-1}]$, $n = \overline{1, p-1}$ и $v = [v_0, v_1, \ldots, v_m, \ldots, v_{p-1}]$, $m = \overline{0, p-1}$. Обозначим значения элементов векторов как $k[n]$ и $v[m]$.

Шаг 1. Организовать цикл по всем значениям вектора k используя переменную $n = \overline{1, p-1}$, включающий шаги 2 … 10.

Шаг 2. Организовать цикл по всем значениям вектора v используя переменную $m = \overline{0, p-1}$, включающий шаги 3 … 10.

Шаг 3. Организовать цикл по всем значениям параметра $i = \overline{0, p-1}$, включающий шаги 4 … 10.

Шаг 4. Ввести временные переменные x, u и y для хранения промежуточных результатов вычислений и присвоить им начальные значения $x = 1$, $u = r$ и $y = i$.

Шаг 5. Организовать цикл, включающий шаги 6 … 9 до тех пор пока значение $u > 0$.

Шаг 6. Если значение $u \,\%\, 2$ равно нулю (знак $\%$ обозначает остаток от деления u на 2 или $u \bmod 2$, что равнозначно), то перейти к шагу 8, в противном случае перейти к шагу 7.

Шаг 7. Вычислить выражение $(y \cdot x) \,\%\, p$ и присвоить результат вспомогательной переменной x.

Шаг 8. Вычислить выражение $(y \cdot y) \,\%\, p$ и присвоить результат вспомогательной переменной y.

Шаг 9. Присвоить вспомогательной переменной u целую часть от деления u на 2.

Шаг 10. Результат вычисления выражения $(k[n]\cdot x + v[m])\,\%\,p$ вывести как очередное число псевдослучайной последовательности.

Исследования показали, что количество умножений, требуемых для расчета числа ПСП по приведенному алгоритму, составляет

$$\lambda = \rceil \log_2(r) \lceil , \qquad (11)$$

где $\rceil s \lceil$ – обозначает наименьшее целое число, большее или равное s.

Как видно из формулы (11) количество умножений необходимых для расчета одного числа ПСП будет очень медленно расти по сравнению с ростом параметра r.

В табл. 2 приведены некоторые значения параметра r и необходимое количество умножений для получения числа ПСП.

Таблица 2 – Количество операций умножение λ, необходимое для расчета числа ПСП в зависимости от параметра r

r	1 000	5 000	50 000	200 000	500 000	1 000 000	10 000 000
λ	10	13	16	18	19	20	24

Из анализа приведенного алгоритма и табл. 2 можно прийти к выводу, что использование этого алгоритма для проектирования генератора ПСП чисел проводит к резкому увеличению быстродействия генератора, особенно для очень больших чисел, по сравнению с обычными алгоритмами построения генератора ПСП. Помимо этого в приведенном алгоритме практически отсутствует операция возведения в степень и, как показали исследования, все промежуточные числа, получаемые при работе алгоритма, имеют небольшую разрядность, что позволяет использовать его при построении генератора ПСП на базе микроконтроллера.

В заключение отметим, что в данной статье разработана концепция построения ортогональных ППРЧ-кодов большого периода и повышенным уровнем параметрической скрытности. Разработана структурная схема генератора ППРЧ-кодов на базе генератора ПСП, который генерирует числа без пропусков и повторений в заданном диапазоне с равномерным законом распределения. Так же предложен алгоритм быстрого расчёта чисел ПСП. Отметим так же что, использование микроконтроллера для генератора ППРЧ-кодов имеет много преимуществ, в том числе: может обеспечить быстродействующую генерацию кодов, стабильность данных, возможность перестройки входных данных, например, векторов k, v и r, а также может управляться компьютером. Данный генератор ППРЧ-кодов может найти применение при построении многоканальных систем связи с шумоподобными сигналами на базе метода скачкообразной перестройки частоты (FHSS MA).

Литература

1. *Scholtz R.A.* The Origins of Spread Spectrum Communication // IEEE Trans. Commun. – 1982. – Vol. COM30. – № 5, May. – P. 822–854.
2. *Скляр Б.* Цифровая связь. Теоретические основы и практическое применение / Скляр Б. – [2-е изд.] испр.; пер. с англ.– М.: Изд. дом «Вильямс», 2003.– 1104 с.
3. *Computational* Alternatives to Random number Generators [M'Rahi D., Naccache D., Pointcheval D., Vaudenay S.] // Proceedings of Selected Areas in Cryptography'98. – 1998. – August, Kingston, Ontario, Canada. – LNCS 1556. – P. 72−80.
4. *Pei-Chi Wu.* Random Number Generation with Primitive Pentanomials // ACM Transactions on Modeling and Computer Simulation. – 2001. – Vol. 11. – No. 4. – P. 346–351.
5. *Трохименко Я.К.* Радиотехнические расчеты на программируемых микрокалькуляторах: Справочник / Я.К. Трохименко, Ф.Д. Любич – [2-е изд.] перер. и доп. – М.: Радио и связь, 1988. – 304 с.
6. *Виноградов И.М.* Основы теории чисел / Виноградов И.М. – М.: Наука, 1981.–176 с.
7. *Шеннон К.* Работы по теории информации и кибернетике / Шеннон К. – М.: И.Л., 1963. – С. 333-402.
8. *Мазурков М.И.* Частотно-временные коды квадратичных вычетов в расширенных полях Галуа / Мазурков М.И. // Радиоэлектроника. – 1997. – № 12. – С. 30–39. (Изв. вузов).
9. *Варакин Л.Е.* Нелинейные композиционные системы дискретных частотных сигналов / Л.Е. Варакин, О.В. Матвеева// Труды институтов связи. Приемо-передающая техника. ЛЭИС, 1978. – С. 61 – 67.
10. *Кнут Д.Э.* Искусство программирования. Том 2: Получисленные алгоритмы / Кнут Д.Э. – [3-е изд.] – М.: Вильямс, Addison Wesley Longman, 2003. – 788 с.