

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ

Научно-технический журнал
№ 4/2015

Издается с 1999 года
Выходит четыре раза в год

ГЛАВНЫЙ РЕДАКТОР
д.т.н., чл. кор. РАН Ю.Б. ЗУБАРЕВ

ЗАМЕСТИТЕЛИ ГЛАВНОГО РЕДАКТОРА:

д.т.н., проф. В.В. ВИТЯЗЕВ,
д.т.н., проф. В.П. ДВОРКОВИЧ

РЕДАКЦИОННАЯ КОЛЛЕГИЯ:

д.т.н., проф. Ар.С. Аджемов, д.т.н., проф. Б.А. Алпатов,
д.т.н., проф. В.Г. Бартенов, д.т.н. Ю.И. Борисов,
д.т.н., проф. Ю.А. Брюханов, д.т.н., проф. В.В. Еремеев,
д.т.н. Г.В. Зайцев, Р.В. Зубарев, А.П. Кирличников,
д.т.н., проф. М.И. Кривошеев, д.т.н., акад. РАН Н.А. Кузнецов,
В.Г. Мистюков, д.т.н., проф. С.Л. Мишенков,
д.т.н., проф. А.А. Петровский, д.т.н., проф. Ю.Н. Прохоров,
д.т.н., проф. В.Г. Санников, к.т.н., проф. В.С. Сперанский,
д.т.н., проф. Ю.С. Шинаков

Адрес редакции:

г. Москва, ул. Авиамоторная, д. 8
Научный центр МТУСИ
Тел.: 8 (495) 362-42-75
E-mail: rntores@mail.ru
vityazev.v.v@rsreu.ru
http://www.dspsa.ru

Издатель:

Российское научно-техническое общество
радиотехники, электроники и связи им. А.С. Попова
Компьютерная верстка: И.А. Благодарова
Дизайн: М.В. Аверин

Подписной индекс по каталогу
ОАО «Роспечать» – **82185**

Подписано в печать 01.12.15 г.
Формат 60x90/8.

Гарнитура «Arial». Печать офсетная.
Бумага офсетная. Печ.л. 6,5. Тираж 700 экз.

Заказ № 7974. Отпечатано в
ООО НПЦ «Информационные технологии»
Рязань, ул. Островского, д. 21/1
тел.: (4912) 98-69-84

Издание зарегистрировано в Министерстве
Российской Федерации по делам печати,
телерадиовещания и средств
массовых коммуникаций.

Свидетельство о регистрации ПИ № 77-1488
от 14.01.2000 г.

Журнал «Цифровая обработка сигналов» включен
в Перечень ведущих рецензируемых научных изданий,
в которых по рекомендации Минобрнауки РФ, должны
быть опубликованы основные научные результаты дис-
сертаций на соискание ученой степени доктора
и кандидата наук.

УЧРЕДИТЕЛИ: ЗАО «Инструментальные системы» ФГУП «НИИ радио»

В НОМЕРЕ:

Бакке А.В.

Алгоритм частотной и временной синхронизации
для приема OFDM сигналов
по многолучевым каналам связи 3

Паршин Ю.Н., Гусев С.И., Колесников С.В.

Анализ пропускной способности системы
передачи информации от вращающегося объекта
на наземный пункт приема 9

Овечкин Г.В., Шевляков Д.А.

Применение многопорогового декодирования
для коррекции ошибок в беспроводных каналах связи 14

Паршин Ю.Н., Кудряшов В.И.

Корреляционные свойства канальных коэффициентов
MIMO системы связи с маневрирующим объектом 20

Овинников А.А.

Исследование влияния спектра связанности циклов
в графе Таннера на энергетический выигрыш
кодирования известных LDPC кодов 24

Витязев В.В., Никишкин П.Б.

Банки фильтров и OFDM в системах широкополосной
передачи данных со многими несущими 30

Витязев В.В., Горюшкин Р.С.

Анализ шумов квантования многоскоростных структур
узкополосных КИХ-фильтров 35

Романов А.М., Слащёв Б.В.

Новое средство верификации ПЛИС
для задач цифровой обработки сигналов 40

Ушенина И.В.

Способы реализации на ПЛИС многоканальных
адаптивных КИХ-фильтров для систем
активного подавления акустического шума* 49

Сизов М.М., Зюбин В.Е.

Конфигурируемый вычислитель на базе FPGA
для потоковой обработки видеосигналов 55

Букварев Е.А., Кузин А.А., Приблудова Е.Н., Рындык А.Г.,

Многопроцессорные кластеры
на основе сигнальных процессоров
со статической суперскалярной архитектурой 58

Липатников В.С., Чинёнов Д.А.

Отладка алгоритмов обработки радиолокационных
сигналов в системе GNU Radio 63

Подписной индекс по каталогу
ОАО «Роспечать» – **82185**

Digital Signal Processing

Science & Technical Magazine
Issue 4, 2015 year

Is published quarterly since 1999

THE EDITOR-IN-CHIEF:

Dr.Sci. (Tech.), Professor, Corresponding Member
of Russian Academy of Sciences U.N. Zubarev

DEPUTY EDITORS-IN-CHIEF:

Dr.Sci.(Tech.), Professor V.V. Vityazev, Dr.Sci.(Tech.),
Professor V.P. Dvorkovich

EDITORIAL BOARD:

Adzhemov A.S., Dr.Sci.(Tech.), Professor

Alpatov B.A., Dr.Sci.(Tech.), Professor

Bartenev V.G., Dr.Sci.(Tech.), Professor

Borisov Y.I., Dr.Sci.(Tech.)

Bruchanov Y.A., Dr.Sci.(Tech.), Professor

Eremeyev V.V., Dr.Sci.(Tech.), Professor

Zaitsev G.V., Dr.Sci.(Tech.)

Zubarev R.V.,

Kirpichnikov A.P.,

Krivocheev M.I., Dr.Sci.(Tech.), Professor

Kuznetsov N.A., Dr.Sci.(Tech.), Academician

Mistyukov V.G.,

Mishenkov S.L., Dr.Sci.(Tech.), Professor

Petrovsky A.A., Dr.Sci.(Tech.), Professor

Prokhorov Y.N., Dr.Sci.(Tech.), Professor

Sannikov V.G., Dr.Sci.(Tech.), Professor

Speranskii V.S., Dr.Sci.(Tech.), Professor

Shinakov Y.S., Dr.Sci.(Tech.), Professor

Editorial office address:

Aviamotornaya, 8, Moscow, Russia

Research Center of MTUCI

Phone: 8 (495) 362-42-75

E-mail: rntores@mail.ru,

vityazev.v.v@rsreu.ru

Web: <http://www.dspsa.ru>

Publisher:

Russian A.S. Popov Society for Radioengineering, Electron-
ics & Communications

Computer makeup:

I.A. Blagodarova

FOUNDERS:

InSys

Radio Research and Development Institute

CONTENTS:

Bakker A.V.

*Frequency and Timing synchronization algorithm
for OFDM systems for communication over multipath channels*.....3

Parshin Yu. N., Gusev S.I., Kolesnikov S.V.

*High rate communication system for transferring data
from rotating object to ground receiving point*.....9

Ovechkin G.V., Shevlyakov D.A.

*Using of multithreshold decoding
for error correction in wireless channels*.....14

Parshin Y.N., Kudryashov V.I.

*Correlation properties of MIMO communication system
channel coefficients with maneuvering object*.....20

Ovinnikov A.A.

*ACE spectrum influence on channel coding gain
of known LDPC codes*.....24

Vityazev V.V., Nikishkin P.B.

*Filter banks and OFDM in broadband data
transmission multicarrier systems*30

Vityazev V.V., Gorushkin, R.S.

*Analysis of quantization noises in multirate structures
of narrowband FIR-filters*35

Romanov A.M., Slashev B.V.

*Novel FPGA verification tool
for digital signal processing applications*.....40

Ushenina I.V.

*FPGA implementation techniques for multi-channel adaptive
FIR filters of active noise control systems*.....49

Sizov M.M., Zyubin V.E.

Configurable FPGA architecture for video stream processing55

Bukvarev E.A., Kuzin A.A.,

Pribludova E.N., Ryndyk A.G.

*Multiprocessor clusters based
on signal processors
with static super-scalar architecture*58

Lipatnikov V.S., Chinenov D.A.

*Radar signal processing algorithms debug
in GNU Radio system*.....63

Subscription index: 82185 ("Rospechat")

ISSN: 1684-2634 9 771684 263005

УДК 621.3.018.51

АЛГОРИТМ ЧАСТОТНОЙ И ВРЕМЕННОЙ СИНХРОНИЗАЦИИ ДЛЯ ПРИЕМА OFDM СИГНАЛОВ ПО МНОГОЛУЧЕВЫМ КАНАЛАМ СВЯЗИ

*Бакке А.В., к.т.н., доцент Рязанского государственного радиотехнического университета,
e-mail: usr37ru@yandex.ru.*

FREQUENCY AND TIMING SYNCHRONIZATION ALGORITHM FOR OFDM SYSTEMS FOR COMMUNICATION OVER MULTIPATH CHANNELS

Bakker A.V.

In this paper, a novel synchronization algorithm based on constant amplitude zero auto correlation (CAZAC) sequence is presented for orthogonal frequency division multiplexing (OFDM) systems. It uses the CAZAC sequence weighted by a pseudo noise (PN) sequence to obtain a new timing metric, which eliminates the side lobes. The proposed algorithm has a more accurate time and frequency offset estimation suitable for communication over multipath channels. This is verified by computer simulations.

Key words: CAZAC Sequence; OFDM; PN Sequence; time synchronization; frequency offset.

Ключевые слова: OFDM сигнал, многолучевой канал, частотная, временная синхронизация.

Введение

Технология ортогонального частотного разделения с мультиплексированием потока данных (OFDM – Orthogonal frequency-division multiplexing) получила широкое распространение в высокоскоростных системах передачи данных благодаря высокой спектральной эффективности, устойчивости к многолучевому распространению и межсимвольной интерференции. Вместе с тем, эта технология демонстрирует высокую чувствительность к ошибкам символьной (STO – Symbol Time Offset) и частотной синхронизации (CFO – Carrier Frequency Offset) [1], что приводит к существенному снижению качества работы демодулятора.

Все существующие методы синхронизации, применяемые в системах с OFDM сигналами, можно условно разделить на две группы. К первой, наиболее многочисленной, группе относятся алгоритмы, основанные на добавлении в передаваемый сигнал периодически повторяемых специальных вспомогательных символов (преамбулы) [2-4] или пилотных поднесущих (Pilot Subcarriers) [5]. Ко второй группе имеют отношение методы синхронизации, не требующие наличия в сигнале какой-либо дополнительной информации и использующие, как правило, для своих целей имеющийся в OFDM сигналах циклический префикс (CP – Cyclic Prefix) [6]. Способы синхронизации, основанные на циклическом префиксе, обладают простотой реализации и нетребовательны к вычислительным затратам, однако в условиях многолучевого распространения характеризуются значительным ухудшением качества работы.

В предлагаемой статье рассматривается новый способ построения алгоритма синхронизации OFDM сигналов, заключающийся в использовании символа преамбулы специальной структуры на основе взвешенных псевдослучайным кодом последовательностей Задова-Чу. Как будет показано ниже, применение псевдослучайно-

Статья посвящена исследованию свойств нового алгоритма синхронизации OFDM сигналов. Приводятся результаты имитационного моделирования разработанного алгоритма синхронизации для многолучевой модели беспроводного канала связи, подтверждающие преимущество предложенного алгоритма синхронизации.

го кода в преамбуле приводит к существенному снижению дисперсии оценки STO, а введение дополнительной обработки, ограничивающей интервал поиска начала символа, обеспечит устойчивость функционирования предлагаемого алгоритма синхронизации в условиях многолучевого распространения.

Модель сигнала

В общем случае, OFDM символ с N поднесущими может быть записан в виде:

$$x(n) = \frac{1}{\sqrt{N}} \sum_{k=1}^N X(k) e^{j2\pi kn/N}, \quad -N_g \leq n \leq N-1, \quad (1)$$

где $X(k)$ – последовательность комплексных данных, передаваемых на k -й поднесущей, определяемая заданным видом квадратурной модуляции (BPSK, QPSK, 16-QAM и т.д.); N_g – количество отсчетов, приходящихся на CP. После прохождения через канал связи сигнал на входе приемного устройства может быть представлен в виде:

$$r(n) = e^{j2\pi \varepsilon n/N} \sum_{m=0}^{M-1} h(m) x(n - \alpha_m) + \omega(n), \quad (2)$$

где $h(m)$ – значения импульсной характеристика многолучевого канала связи с M лучами; α_m – задержка сигнала m -го луча; $\omega(n)$ – шумовая составляющая принимаемого сигнала; $\varepsilon = \varepsilon_i + \varepsilon_f$ – нормированное к расстоянию между поднесущими значение частотного рассогласования, отражающее доплеровское смещение частоты, а также несоответствие задающих генераторов приемного и передающего устройств; $\varepsilon_i, \varepsilon_f$ – целая и дробная части частотного рассогласования соответственно.

Методы оценки STO и CFO с использованием вспомогательных символов

Из многочисленных работ, посвященных методам синхронизации OFDM сигналов с использованием вспомогательных синхросимволов, можно выделить ряд основополагающих методов, отражающих концептуальный подход к решению поставленной задачи.

Наиболее широкую известность и распространение получил подход [2], предложенный Schmid T. и Cox D. (далее метод SC). Этот метод состоит в периодической передаче символа преамбулы вида $S=[A A]$, состоящего из циклического префикса и двух идентичных последовательностей A длиной $N/2$ (рис. 1).

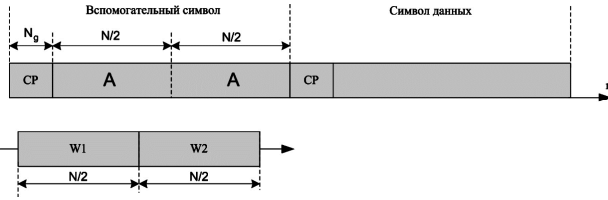


Рис. 1. Структура вспомогательного символа метода SC

Для определения момента начала OFDM символа достаточно выявить в принимаемом сигнале факт минимальной разницы двух блоков отсчетов W_1 и W_2 , разнесенных на $N/2$ выборок. Соответствующий критерий оценки временного положения STO $\hat{\delta}$ OFDM символа может быть определен максимумом функции правдоподобия вида [2]:

$$\hat{\delta} = \arg \max_{\delta} (M(\delta)) = \arg \max_{\delta} \left(\frac{|P(\delta)|}{R(\delta)} \right)^2, \quad (3)$$

$$P(\delta) = \sum_{i=\delta}^{N/2-1+\delta} r(i) \cdot r^*(N/2+i), \quad (4)$$

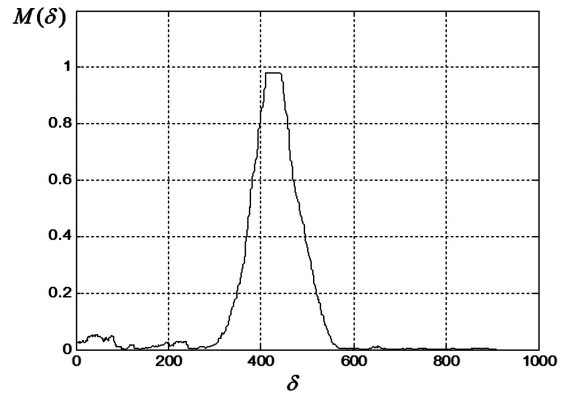
$$R(\delta) = \frac{1}{2} \sum_{i=\delta}^{N/2-1+\delta} |r(N/2+i)|^2, \quad (5)$$

где $M(\delta)$ – временная метрика обнаружения OFDM символа; $\hat{\delta}$ – номер отсчета, на котором достигается максимум функции $M(\delta)$; $r^*(n)$ – комплексно-сопряженная функция (2). Особенностью данного способа оценки STO является неоднозначность точной оценки $\hat{\delta}$ вследствие наличия в функции (3) плоского участка длиной N_g (рис 2 а, иллюстрация функции $M(\delta)$ для канала связи с белым шумом, отношение сигнал/шум $q=20$ дБ), связанного с наличием в символе циклического префикса. Это обстоятельство приводит к существенному увеличению дисперсии оценок $\hat{\delta}$.

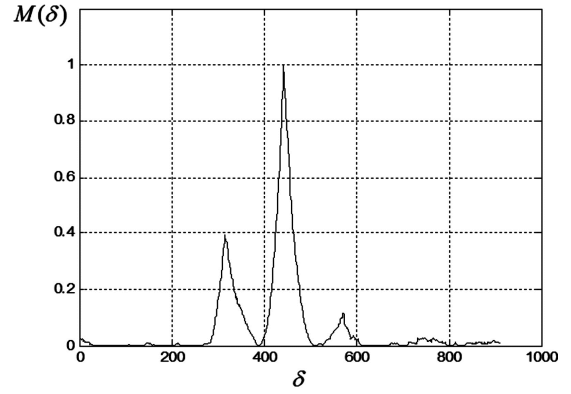
Это обстоятельство приводит к существенному увеличению дисперсии оценок $\hat{\delta}$ (рис. 2 а, иллюстрация функции $M(\delta)$ для канала связи с белым шумом, отношение сигнал/шум $q=20$ дБ).

В соответствии с [2], оценка дробной составляющей CFO $\hat{\varepsilon}_f$ может быть получена вычислением аргумента значения $P(\hat{\delta})$:

$$\hat{\varepsilon}_f = -\frac{1}{2\pi} \arctan \left\{ \frac{\text{Im}(P(\hat{\delta}))}{\text{Re}(P(\hat{\delta}))} \right\}. \quad (6)$$



а)



б)

Рис. 2. Примеры временных метрик $M(\delta)$ обнаружения OFDM символа для методов SC (а) и Minn (б)

Выражение (6) обеспечивает однозначное определение CFO в диапазоне нормированных рассогласований $|\hat{\varepsilon}_f| < 0.5$, т.е. с точностью до половины расстояния между поднесущими.

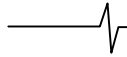
Дальнейшим развитием метода SC стали алгоритмы синхронизации, обеспечивающие существенное меньшее значение дисперсии оценки $\hat{\delta}$ за счет использования преамбулы специальной структуры. К примеру, авторами в [3] (далее метод Minn) для уменьшения дисперсии оценки $\hat{\delta}$ предложено использовать вспомогательные синхросимволы в виде комбинации L идентичных блоков A размером N/L отсчетов, взвешенных элементами $p(l)$ некоторой бинарной последовательности $\{p(l)\}_{l=0}^{L-1}$. В этом случае автокорреляционные свойства $\{p(l)\}_{l=0}^{L-1}$ будут определять форму результирующей метрики $M(\delta)$. К примеру, для случая $L=4$ и взвешивающей последовательности $p=[1 -1 1 1]$ символ преамбулы может быть представлен в виде $s = [A -A A A]$.

С учетом (3), выражение для оценки STO по методу Minn будет иметь вид [3]:

$$\hat{\delta} = \arg \max_{\delta} (M(\delta)) = \arg \max_{\delta} \left(\frac{L}{L-1} \frac{|P(\delta)|}{R(\delta)} \right)^2, \quad (7)$$

$$P(\delta) = \sum_{j=0}^{L-2} b(j) \sum_{i=\delta}^{N_1-1+\delta} r(jN_1+i) \cdot r^*(N_1(j+1)+i), \quad (8)$$

$$R(\delta) = \sum_{j=0}^{L-1} \sum_{i=\delta}^{N_1-1+\delta} |r(jN_1+i)|^2, \quad (9)$$



где $b(j) = p(j)p(j+1)$, $j = \overline{0, L-2}$; $N_1 = N/L$. Пример метрики $M(\delta)$ для $L=4$ и последовательности $p = [1 -1 1 1]$ представлен на рис. 2 б. Указанный способ синтеза символа преамбулы обеспечивает существенно меньшее значение дисперсии оценки $\hat{\delta}$ по сравнению с методом SC, однако в силу наличия в $M(\delta)$ боковых пиков это уменьшение может оказаться недостаточно низким. Оценка дробной составляющей CFO ε_f для алгоритма Minn может быть получена аналогично (6), но с той лишь разницей, что рассматриваемая структура символа преамбулы ($L=4$) дает однозначное определение CFO в диапазоне нормированных рассогласований $|\varepsilon_f| < 1$, т.е. с точностью до расстояния между поднесущими.

Дальнейшее уменьшение дисперсии оценки $\hat{\delta}$ достигается возможным при использовании в составе символов преамбулы псевдослучайных последовательностей, обладающих низким уровнем боковых выбросов автокорреляционной функции (АКФ), или последовательностей Задова-Чу с нулевыми боковыми выбросами АКФ. Для уменьшения боковых выбросов в методе Minn в работе [4] предложен способ формирования преамбулы, характеризующийся низкой дисперсией оценки $\hat{\delta}$. Тем не менее, в условиях быстро меняющегося многолучевого канала связи, отмечается снижение точности оценки этого метода.

В работе [7] рассматривается вариант применения вспомогательного символа вида $S = [A D^*]$, в котором D^* является комплексно-сопряженной последовательностью, полученной зеркальным отражением последовательности A . Особенностью этого метода является применение в качестве A последовательности Задова-Чу [8], характеризующейся нулевыми боковыми выбросами периодической АКФ. Экспериментальная проверка такого способа синхронизации при достаточно большой длине символа ($N=1024$) показывает низкие значения дисперсии оценки $\hat{\delta}$ в условиях быстро меняющегося многолучевого канала связи [7], однако такая структура преамбулы не позволяет формировать какие-либо оценки частотного рассогласования ε и, следовательно, приводит к быстрой деградации качества оценивания $\hat{\delta}$ с ростом ε .

Разработка алгоритма синхронизации OFDM сигналов

В данной работе предлагается новый алгоритм синхронизации (далее PR), который, как будет показано ниже, при относительно короткой длине символа преамбулы обеспечивает получение устойчивых оценок STO и CFO в условиях быстро меняющегося многолучевого канала связи. Предлагаемый способ синхронизации является развитием идей работ [3, 9] и состоит в применении преамбулы следующего вида:

$$s = [B - A B A], \quad (10)$$

где $A = \{a(k)\}_{k=0}^{N_1}$ – матрица строка, состоящая из значений последовательности Задова-Чу, $a(k) = \exp(j\pi(N_1 - 1)k^2 / N_1)$, $N_1 = N/4$; матрица B является копией A , взвешенной псевдослучайным ко-

дом: $B = A \cdot PN^T = \{a(k) \cdot pn(k)\}_{k=0}^{N_1}$; $PN = \{pn(k)\}_{k=0}^{N_1-1}$ – матрица строка, состоящая из элементов бинарной псевдослучайной последовательности. Взвешивание блоков преамбулы кодом $p = [1 -1 1 1]$, согласно методу Minn [3], а также первого и третьего блоков псевдослучайным кодом позволяет сконцентрировать отклик функции правдоподобия $M(\delta)$ в узком диапазоне возможных оценок STO $\hat{\delta}$. Такое решение позволит обеспечить надежное определение временного положения символа преамбулы, необходимое для оценки и последующей компенсации частотного рассогласования CFO.

Процедура синхронизации по синхросимволу (10) будет включать в себя следующие этапы.

1. Предварительная (грубая) временная синхронизация.

Согласно (7-9) критерий оценки временного положения STO δ для символа (10) определяется максимумом функции правдоподобия следующего вида:

$$\hat{\delta} = \arg \max_{\delta} (M(\delta)) = \arg \max_{\delta} \left(\frac{4 |P(\delta)|}{3 R(\delta)} \cdot |F(\delta)| \right)^2, \quad (11)$$

$$P(\delta) = \sum_{j=0}^2 b(j^{(0)}) \times \sum_{i=\delta}^{N_1-1+\delta} W(i) \cdot r(jN_1 + i) \cdot r^*(N_1(j+1) + i), \quad (12)$$

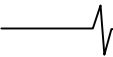
$$R(\delta) = \sum_{j=0}^3 \sum_{i=\delta}^{N_1-1+\delta} |r(jN_1 + i)|^2, \quad (13)$$

$$F(\delta) = \frac{\sum_{k=1}^{L-2} \left(\sum_{j=0}^{L-k-2} \mathbf{b}^{(k)}(j) \sum_{i=\delta}^{N_1-1+\delta} r(jN_1 + i) \cdot r^*(N_1(j+k+1) + i) \right)}{2 \sum_{i=\delta}^{N_1-1+\delta} |r(i)|^2} = \frac{P_F(\delta)}{R_F(\delta)}, \quad (14)$$

где $W = [PN I PN]$ – матрица строка, состоящая из матриц псевдослучайного кода и единичной матрицы строки I размером N_1 элементов; $\mathbf{b}^{(k)}(j) = p(j)p(j+k+1)$, $j = \overline{0, L-k-2}$, $k = \overline{1, L-2}$. Отличительной особенностью предлагаемого критерия оценки STO является наличие в функции правдоподобия дополнительного множителя $F(\delta)$, который в целом соответствует метрике (5), рис. 2 б. На рис. 3. ($q=0$ дБ, используется релейевская модель 6-и лучевого канала связи; $N=256$) наглядно поясняется роль этой функции: использование в (11) $F(\delta)$ приводит к эффективному ограничению области принятия решения относительно δ и к преимущественному подавлению побочных выбросов функции $P(\delta)$.

2. Определение и компенсация дробной составляющей CFO ε_f .

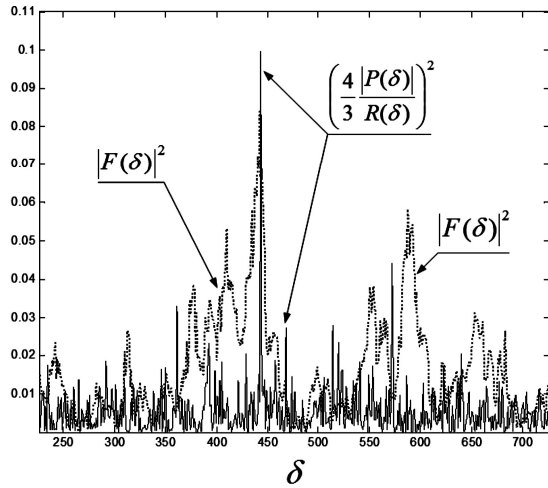
Фактическое частотное смещение согласно (2) можно представить в виде суммы двух компонент: целочисленной и дробной: $\varepsilon = \varepsilon_i + \varepsilon_f$. В соответствии с [2], оценка ε_f может быть получена вычислением аргумента зна-



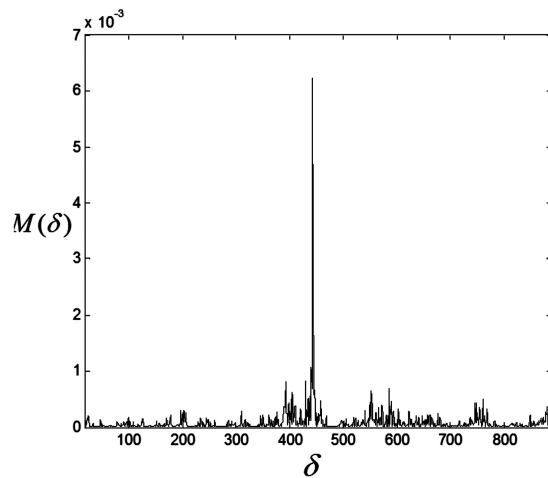
чения функции $P_F(\hat{\delta})$ из (14):

$$\hat{\varepsilon}_f = -\frac{1}{\pi} \arctan \left\{ \frac{\text{Im}(P_F(\hat{\delta}))}{\text{Re}(P_F(\hat{\delta}))} \right\}. \quad (15)$$

В отличие от (6) выражение (15) обеспечивает однозначное определение CFO в диапазоне нормированных рассогласований $|\hat{\varepsilon}_f| < 1$, т.е. с точностью до расстояния между поднесущими. После получения оценки $\hat{\varepsilon}_f$ осуществляется компенсация дробной составляющей CFO в принятом сигнале: $r(k) \cdot \exp(-j2\pi k \hat{\varepsilon}_f / N)$, где k – порядковый номер отсчета сигнала.



а)



б)

Рис. 3. Иллюстрации компонент метрики $M(\delta)$ (а) и результирующий вид $M(\delta)$

(б) в 6-и лучевом нестационарном канале связи ($q=0$ дБ, $N=256$)
3. Определение и компенсация целочисленной составляющей CFO ε_i .

Для определения частотного рассогласования ε_i , выраженного в единицах расстояния между поднесущими, используется взаимная корреляция спектров символа преамбулы (10) и принятого синхрослова:

$$G(\varepsilon) = \frac{\left| \sum_{n=0}^{N-1} X(n+\varepsilon) \cdot S(n) \right|^2}{\left(\sum_{n=0}^{N-1} |X(n+\varepsilon)|^2 \right)^2}, \quad \varepsilon = \overline{0, N-1}, \quad (16)$$

где $S(n) = FFT \{s(k)\}_{k=0}^{N-1}$ и $X(n) = FFT \{r(k)\}_{k=\hat{\delta}}^{N+\hat{\delta}-1}$ – спектры преамбулы (10) и принятого синхрослова соответственно, $s(k)$ – отсчеты символа преамбулы; FFT – операция быстрого преобразования Фурье. В качестве примера на рис. 4 представлена иллюстрация функции

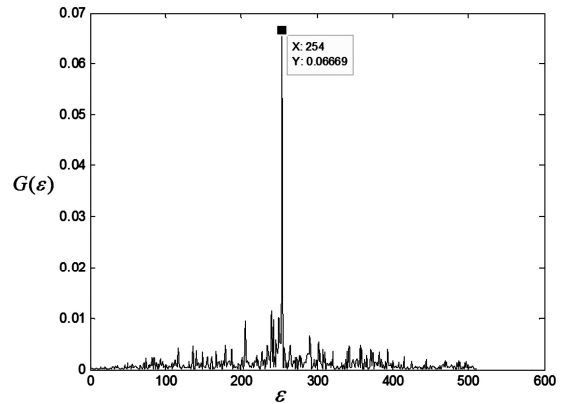


Рис. 4. Пример функции $G(\varepsilon)$ для фактического значения $\varepsilon_i = -2$ ($q=0$ дБ, $N=256$)

$G(\varepsilon)$, отражающая случай фактического значения $\varepsilon_i = -2$ ($q=0$ дБ, используется релейевская модель 6-и лучевого канала связи; $N=256$). На основании (16) может быть получена оценка $\hat{\varepsilon}_i$ в форме:

$$\hat{\varepsilon}_i = \arg \max_{\varepsilon} (G(\varepsilon)) - N. \quad (17)$$

После получения оценки $\hat{\varepsilon}_i$ осуществляется компенсация целочисленной составляющей CFO в принятом сигнале (аналогично п. 2).

4. Точная временная синхронизация.

На последнем этапе синхронизации выполняется точное определение начала принятого символа преамбулы. Эта операция состоит в поиске максимума функции взаимной корреляции символа преамбулы (10) и принятого синхрослова:

$$\hat{\delta}' = \arg \max_{\delta'} \left(\sum_{k=0}^{N-1} r(k+\delta') \cdot s(k) \right), \quad k = \overline{0, N-1}, \quad (18)$$

где $\hat{\delta}'$ – точная оценка времени начала принятого символа преамбулы.

Результаты численного моделирования

Цель эксперимента состояла в проверке эффективности работы рассмотренного выше алгоритма синхронизации PR в условиях нестационарного многолучевого канала связи. В качестве модели канала связи была выбрана типовая 6-лучевая модель распространения радиоволн для высокоскоростных транспортных средств (Vehicular Test Environment) [10]. Моделирование проводилось для OFDM сигналов с количеством поднесущих $N = 256$ и величиной циклического префикса $N_g = 32$ отсчета.

На рис. 5 представлены результаты моделирования первого этапа синхронизации: зависимости среднеквадратической ошибки (СКО) получаемых оценок STO $\hat{\delta}$ от отношения сигнал-шум q согласно алгоритму Minn (7) и предлагаемому методу PR (11). Полученные данные указывают на высокую эффективность предлагаемого

алгоритма в условиях многолучевости: начиная с отношения сигнал-шум 5 дБ использование взвешивания двух блоков преамбулы псевдослучайным кодом и применение нормирующего множителя $F(\delta)$ в (9) обеспечивают резкое снижение СКО.

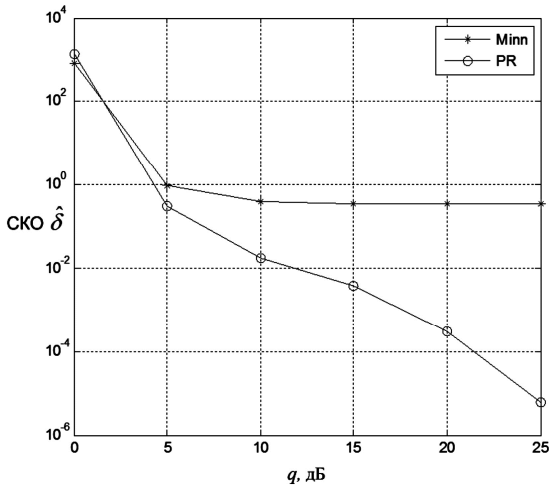


Рис. 5. Зависимости СКО $\hat{\delta}$ в условиях многолучевой модели канала связи для методов Minn и PR

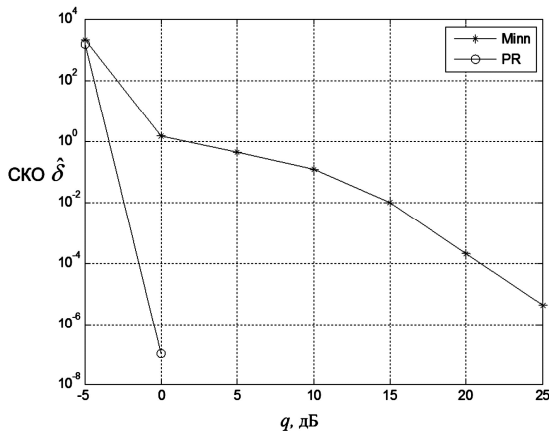


Рис. 6. Зависимости СКО $\hat{\delta}$ в условиях стационарного канала с белым шумом для методов Minn и PR

Для сравнения на рис. 6 отражены аналогичные зависимости для стационарного канала связи с белым шумом. В этом случае значения оценок STO $\hat{\delta}$ для предлагаемого алгоритма стремятся к нулю начиная с $q=0$ дБ.

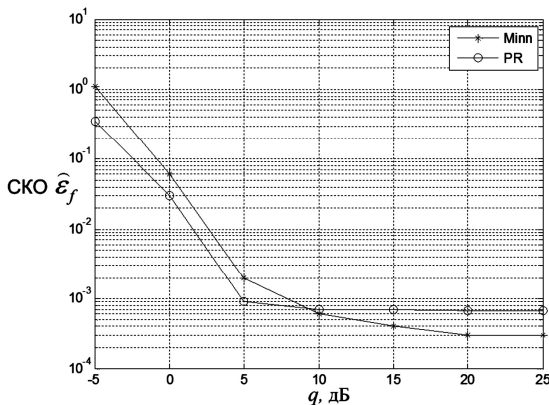


Рис. 7. Зависимости СКО $\hat{\epsilon}_f$ в условиях многолучевой модели канала связи для методов Minn и PR

На рис. 6 представлены зависимости нормированной среднеквадратической ошибки *дробной составляющей* CFO $\hat{\epsilon}_f$ от отношения сигнал-шум для алгоритма Minn (8) и предлагаемого метода PR (2) в условиях многолучевого канала связи с заданным смещением частоты $\epsilon_f = 0.3$. Аналогичные зависимости для стационарного канала связи с белым шумом изображены на рис. 8.

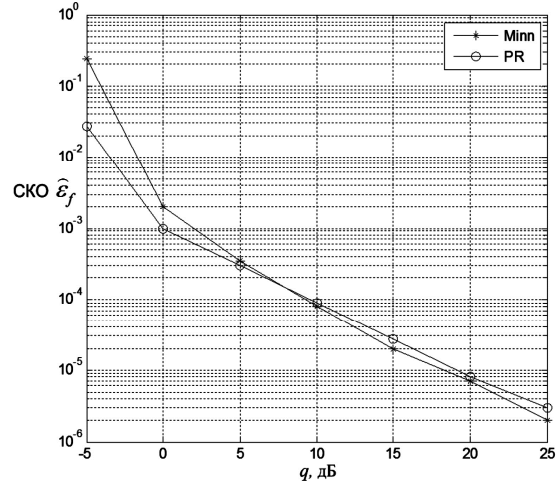


Рис. 8. Зависимости СКО $\hat{\epsilon}_f$ в условиях стационарного канала с белым шумом для методов Minn и PR

О качестве оценки *целочисленной составляющей* CFO $\hat{\epsilon}_i$ позволяет судить рис. 9, на котором приведены зависимости вероятности правильного определения P_ϵ оценки $\hat{\epsilon}_i$ от отношения сигнал-шум согласно (17) для многолучевого канала связи (кривая MP) и канала связи с белым шумом (кривая AWGN).

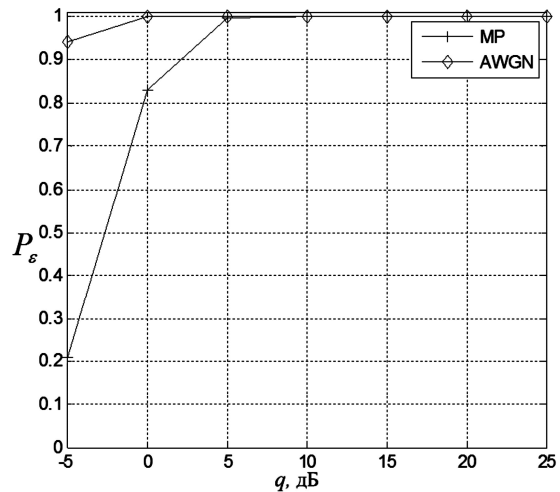
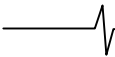


Рис. 9. Зависимости вероятности правильного определения P_ϵ оценки $\hat{\epsilon}_i$ от отношения сигнал-шум для многолучевого канала связи и канала связи с белым шумом

Как следует из результатов вычислительного эксперимента, применение предложенного алгоритма синхронизации с использованием относительно короткого ($N=256$) символа преамбулы (10) и функции правдоподобия (11) позволяет весьма успешно решать задачу оценки частотного и временного рассогласований как для каналов связи с белым шумом, так и в условиях многолучевого распространения. Эффект от применения рассматриваемого алгоритма проявляется в существен-



но меньших значениях оценок STO $\hat{\delta}$ по сравнению с алгоритмом Minn при сопоставимой точности определения CFO $\hat{\varepsilon}_f, \hat{\varepsilon}_i$.

Заключение

Предлагаемый в статье алгоритм синхронизации обеспечивает получение устойчивых оценок частотного и временного рассогласований в условиях нестационарного многолучевого канала связи при относительно короткой длине символа преамбулы. Было достигнуто существенное улучшение точности оценки временного рассогласования благодаря использованию в символах преамбулы псевдослучайной последовательности и применению функции правдоподобия вида (11).

Исследование выполнено при поддержке гранта Российского научного фонда (проект 14-19-01263) в Рязанском государственном радиотехническом университете

Литература

1. T. Pollet, M. Van Bladel, and M. Moeneclaey. Ber sensitivity of OFDM systems to carrier frequency offset and Wiener phase noise, IEEE Trans. Commun., vol. 43, pp. 191-193, 1995.
2. Schmidl T.M., Cox D.C. Robust Frequency and Timing Synchronization for OFDM. IEEE Trans. Communications, vol. 45, no.12, pp. 1613-1621, 1997.
3. Hlaing Minn, Vijay K. Bhargava, Khaled Ben Letaief. A

Robust Timing and Frequency Synchronization for OFDM Systems, IEEE Transactions on Wireless communications, vol. 2, no. 4, pp. 822-838, 2003.

4. B. Park and H. Cheon, C.G. Kang, and D. S. Hong. A Novel Timing Estimation Method for OFDM systems, IEEE Commun. Lett., vol. 7, pp. 239-241, May 2003

5. Classen, F. and Myer, H. Frequency synchronization algorithm for OFDM systems suitable for communication over frequency selective fading channels. IEEE VTC'94, pp. 1655-1659, 1994.

6. J.J. van de Beek, M. Sandell, and P.O. Bojesson, ML estimation of time and frequency offset in OFDM systems, IEEE Transactions on Signal Processing, vol. 45, no. 7, pp. 1800-1805, 1997.

7. S.D. Choi, J.M. Choi, J.H. Lee. An initial timing offset estimation method for OFDM systems in Rayleigh fading channel, IEEE 64th Vehicular Technology Conference, pp. 1-5, September 2006.

8. D.C. Chu. Polyphase codes with good periodic correlation properties, IEEE Trans. Inform. Theory, vol. IT-18, pp. 531-532, July 1972.

9. H. Wang, L. Zhu, Y. Shi, T. Xing, and Y. Wang. A novel synchronization algorithm for OFDM systems with weighted CAZAC sequence. J. Comput. Inf. Syst. 8(6), pp. 2275-2283, 2012.

10. IEEE 802.20 Mobile Broadband Wireless Access Working Group, "Channel Models for IEEE 802.20 MBWA System Simulations", Version 9, Revision 1, July 2005.

УДК 621.391.812

АНАЛИЗ ПРОПУСКНОЙ СПОСОБНОСТИ СИСТЕМЫ ПЕРЕДАЧИ ИНФОРМАЦИИ ОТ ВРАЩАЮЩЕГОСЯ ОБЪЕКТА НА НАЗЕМНЫЙ ПУНКТ ПРИЕМА

Паршин Ю.Н., д.т.н., проф., заведующий кафедрой «Радиотехнические устройства» РГРТУ, e-mail: parshin.y.n@rsreu.ru;

Гусев С.И., к.т.н., доцент, ведущий научный сотрудник РГРТУ, e-mail: s.i.gusev@inbox.ru;

Колесников С.В., старший преподаватель кафедры «Радиотехнические устройства» РГРТУ, e-mail: sergey_v.k@mail.ru.

HIGH RATE COMMUNICATION SYSTEM FOR TRANSFERRING DATA FROM ROTATING OBJECT TO GROUND RECEIVING POINT

Parshin Yu. N., Gusev S.I., Kolesnikov S.V.

The paper discusses the design options of a system for data transmission from a rotating object – airborne vehicle or spacecraft to the ground receiving station with various spatial antenna structures: SISO, SIMO, MISO, MIMO. Different variants of the onboard antenna system with a different number and spatial structure of antennas are investigated. Individual antenna directivity pattern and antenna system directivity patterns are calculated. Values of the channel matrix for the communication system for the aircraft rotating around the longitudinal axis are calculated. Transmission rate averaged over the period of rotation according to the position of the aircraft on the trajectory and spatial structure of the ground receiving station is calculated. Questions of technical implementation of communication system with different spatial structure are discussed.

Key words: high rate communication system, rotating object, structure of antennas, ground receiving point, MIMO.

Ключевые слова: высокоскоростная связь, пропускная способность, вращающийся летательный аппарат, антенная система, диаграмма направленности, MIMO.

Введение

Построение высокоскоростной систем передачи информации от вращающегося объекта и решение различных исследовательских задач на его борту требует создания высокоскоростной системы передачи информации на наземный пункт приема [1, 7]. Особенностью условий функционирования аппаратуры передачи информации для некоторых типов носителей исследовательской аппаратуры является быстрое вращение объекта вокруг продольной оси, создаваемое с целью стабилизации летательного аппарата при движении по всей траектории полета. Также при движении объекта изменяется ориентация относительно наземного пункта приема. Все это усложняет работу системы передачи информации, требует согласования направленных свойств бортовой и наземной антенных систем с характером движения объекта, учета влияния изменения их взаимной ориентации на пропускную способность канала передачи информации.

В работе [7] излагаются проблемы, связанные с изменением ориентации объекта относительно наземного пункта приема, что приводит к снижению эффективности передачи информации. Эффект затенения антенны корпусом объекта при его вращении может привести даже к прерыванию передачи информации. Для повышения надежности работы используются несколько наземных

Исследованы варианты построения систем передачи информации от вращающегося объекта на наземный пункт приема с различными пространственными структурами антенн: SISO, SIMO, MISO, MIMO. Рассмотрены варианты построения бортовой антенной системы с различным числом антенн. Рассчитаны значения канальной матрицы на траектории движения летательного аппарата с учетом его вращения вокруг продольной оси. На основе точных значений канальных матриц рассчитана усредненная за период вращения пропускная способность системы передачи информации в зависимости от пространственной структуры наземного пункта приема. Особое внимание уделено влиянию пространственной структуры приемных и передающих антенн на пропускную способность. Рассмотрены вопросы технической реализации систем передачи информации с различной пространственной структурой.

пунктов приема, что приводит к значительным затратам.

В данной статье для повышения качества передачи информации используется MIMO технология передачи информации [5], получившая широкое распространение в мобильных системах передачи информации. Особенностью рассматриваемой задачи является наличие нескольких антенн на борту вращающегося объекта для передачи информации на наземный пункт приема. Поэтому актуальной задачей является оптимизация передающей части системы, что можно реализовать путем пространственно-временного кодирования [5,6].

Целью работы является получение предельных значений пропускной способности системы передачи информации от вращающегося объекта при различных пространственных структурах антенных систем и разработка рекомендаций по технической реализации исследуемой системы.

Постановка задачи

Пусть ось вращения объекта перпендикулярна плоскости расположения приемных пунктов. Вращение объекта производится с некоторой угловой скоростью Ω . Передающие антенны располагаются на поверхности объекта, имеющего вид цилиндра радиусом r , равномерно по окружности, угловые координаты антенн $\gamma_{Tm} = 2\pi(m-1)/N_T$, $m=1, \dots, N_T$ (рис. 1). Приемные антенны расположены в плоскости вращения объекта по дуге радиуса R равномерно в угловом секторе $\Delta\gamma_R$, угловые координаты антенн $\gamma_{Rn} = \Delta\gamma_R(n-1)/N_R$, $n=1, \dots, N_R$.

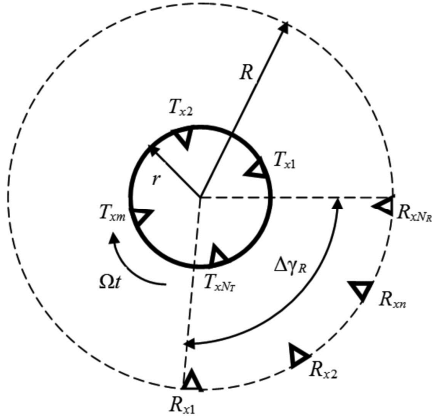


Рис. 1. Пространственное расположение передающих и приемных антенн

Направленные свойства передающих антенн определяются типом излучателей, расположением на поверхности цилиндра, влиянием растекания токов по поверхности цилиндра, а также взаимным влиянием излучателей [3, 7]. В простейшем случае можно считать, что корпус объекта полностью затеняет антенну, а диаграмма направленности излучателей передающей антенны в плоскости вращения задается выражением

$$D(\gamma) = \begin{cases} \cos \gamma, & \cos \gamma > 0 \\ 0, & \cos \gamma < 0 \end{cases}$$

При вращении ориентация m -й передающей антенны относительно n -й приемной антенны определяется углом поворота объекта и угловыми положениями передающей антенны на объекте и приемной антенны на местности: $\gamma_{nm} = \Omega t + \gamma_{Tm} - \gamma_{Rn}$. Коэффициенты канальной матрицы определяются в предположении прямой видимости между приемными и передающими антеннами и зависят от ориентации передающих антенн в направлении приемных антенн, а также от расстояния между фазовыми центрами R_{nm} каждой пары m -й передающей и n -й приемной антенн. Полагая, что $R_{nm} \gg r$ и изменение расстояния между антеннами при вращении объекта практически не влияет на амплитуду принимаемого сигнала, запишем канальный коэффициент в виде:

$$H_{nm} = D(\gamma_{nm}) e^{-j \frac{2\pi}{\lambda} R_{nm}},$$

$$\text{где } R_{nm} = \sqrt{(R \cos \gamma_{Rn} - r \cos(\Omega t + \gamma_{Tm}))^2 + (R \sin \gamma_{Rn} - r \sin(\Omega t + \gamma_{Tm}))^2}.$$

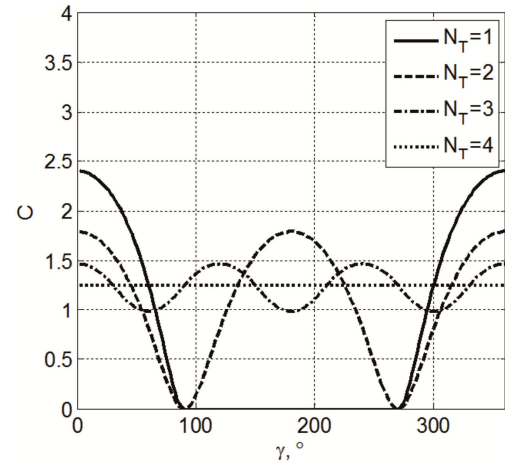
Пропускная способность канала передачи информации определяется выражением [5]:

$$C = \log_2 \det \left(\mathbf{I}_{N_R} + \frac{q}{N_T} \mathbf{H} \mathbf{H}^H \right), \text{ бит/символ,} \quad (1)$$

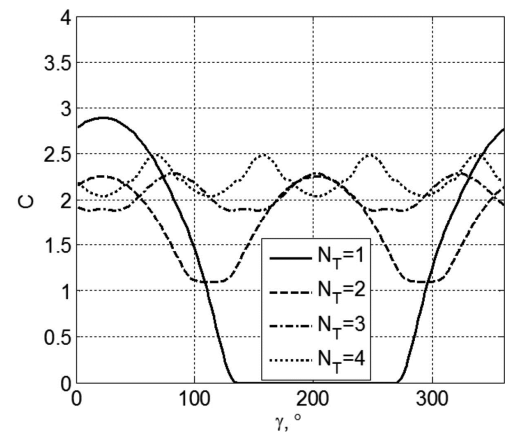
где \mathbf{I}_{N_R} – единичная диагональная матрица размерностью N_R , $q = D_X / D_N$ – энергетический потенциал системы, равный отношению мощности сигнала, излучаемой всеми передающими антеннами D_X , к мощности шума в каждой приемной антенне D_N , $\mathbf{H} = \{H_{nm}, n=1, \dots, N_R, m=1, \dots, N_T\}$ – матрица канальных коэффициентов.

Результаты анализа пропускной способности

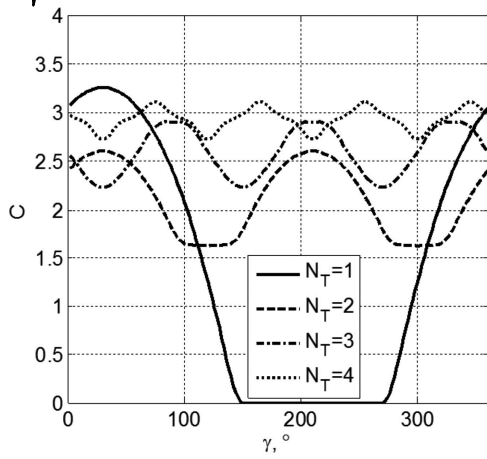
На рис. 2 а, 2 б, 2 в, 2 г приведены зависимости пропускной способности системы передачи информации от угла поворота летательного объекта по оси вращения за время, равное периоду вращения объекта, при различном числе приемных N_R , передающих антенн N_T и $R = 10^6 \lambda$, $r = 5 \lambda$ и расположении приемных антенн в секторе $\Delta\gamma_R = \pi/2$.



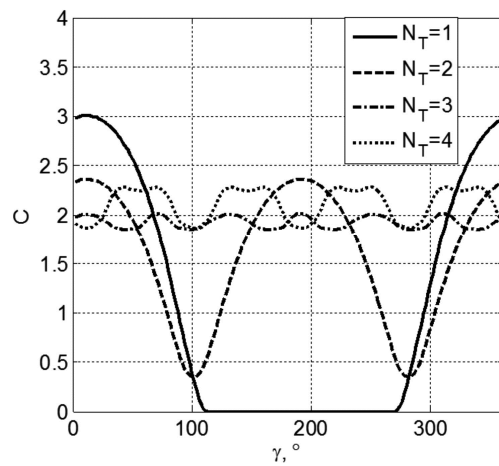
а)



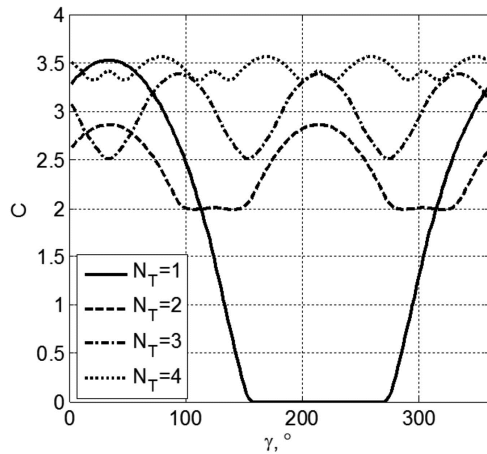
б)



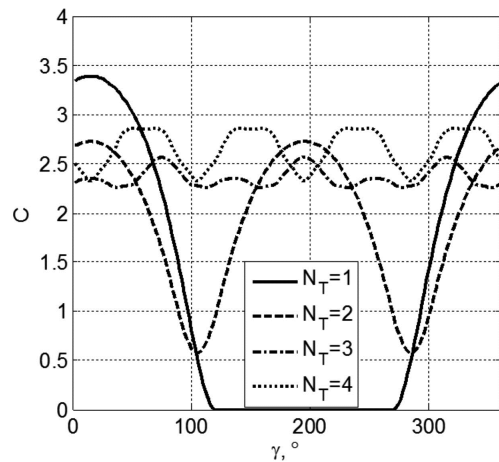
а)



б)



в)



г)

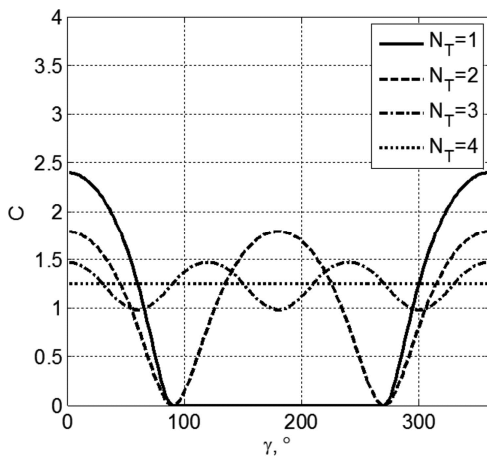
Рис. 2. Пропускная способность системы

передачи информации $\Delta\gamma_R = \pi/2$ а) $N_R = 1$; б) $N_R = 2$;

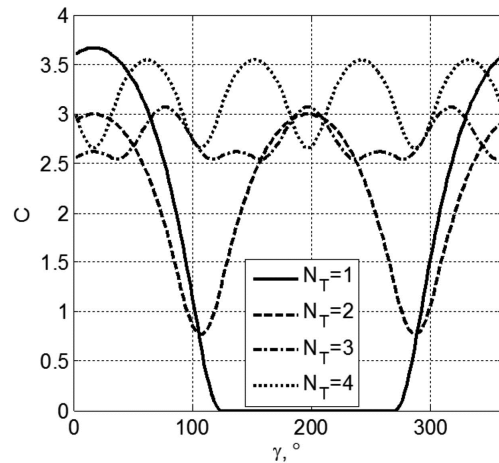
в) $N_R = 3$; г) $N_R = 4$

На рис. 3 а, 3 б, 3 в, 3 г приведены зависимости пропускной способности от угла поворота летательного объекта по оси вращения за время, равное периоду вращения объекта, при различном числе приемных N_R и передающих антенн N_T , $R = 10^6 \lambda$, $r = 5\lambda$, расположении приемных антенн в секторе $\Delta\gamma_R = \pi/4$.

При большой скорости вращения объекта практический интерес представляет средняя за период вращения пропускная способность канала передачи информации.



а)



б)

Рис. 3. Пропускная способность системы передачи информации $\Delta\gamma_R = \pi/4$ а) $N_R = 1$; б) $N_R = 2$;

в) $N_R = 3$; г) $N_R = 4$

В этом случае потенциальную пропускную способность можно получить путем использования перемежения в сочетании с эффективным помехоустойчивым кодированием. На рис. 4 а, 4 б, 4 в, 4 г, 4 д и 4 е приведены зависимости средней пропускной способности от углового сектора расположения приемных антенн при различном числе передающих антенн.

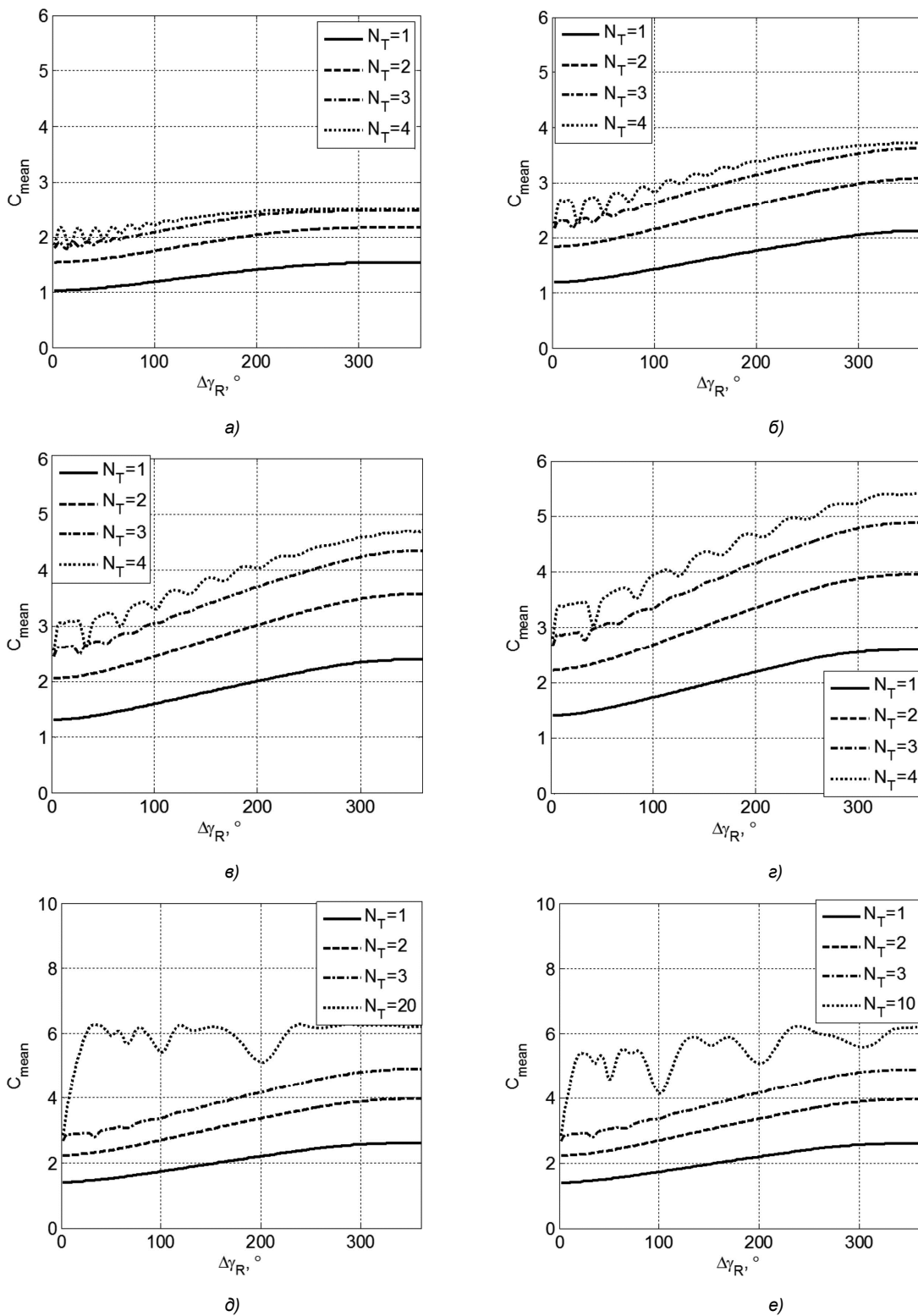


Рис. 4. Средняя за период вращения пропускная способность а) $N_R = 2$; б) $N_R = 3$; в) $N_R = 4$; г) $N_R = 5$; д) $N_R = 5$; е) $N_R = 5$

Заключение

На основании результатов проведенного исследования мгновенной и средней пропускной способности предлагаются два варианта технической реализации высокоскоростной системы передачи информации с вращающегося объекта на наземный пункт приема.

В первом случае предполагается размещение на борту летательного аппарата кольцевой заполненной антен-

ной решетки, с диаграммой направленности, свободной от глубоких провалов и дифракционных лепестков. В этом случае обеспечивается минимальная вариация мгновенной пропускной способности в зависимости от ориентации объекта, а также максимальная пропускная способность. Вместе с тем ограничения на механическую прочность корпуса объекта и сложность реализации передающего тракта, содержащего N_T усилителей мощности или СВЧ тракт фазирования и распределения мощности элемен-

тов антенной решетки от одноканального передатчика не позволяют реализовать данный вариант при большом числе антенн. Кроме того, реализация многоканального передатчика и СВЧ тракта фазирования и распределения мощности с необходимым коэффициентом полезного действия затруднена вследствие ограничений на массогабаритные параметры бортовой аппаратуры. Установка кольцевой заполненной антенной решетки в конструкцию объекта с диаметром, превышающим 1...2 длины волны, сопровождается изменениями в конструкции объекта, приводящими к уменьшению механической прочности, что не всегда допустимо.

Второй вариант высокоскоростной системы передачи информации предполагает размещение на борту объекта малозлементной кольцевой антенной решетки и многопозиционной наземной приемной системы. Высокая средняя пропускная способность достигается при использовании эффективного помехоустойчивого кодирования. При этом реализация кодирования и декодирования, обеспечивающего необходимую скорость передачи информации, может потребовать значительных вычислительных ресурсов. Одним из главных недостатков рассматриваемого метода являются чрезвычайно высокие требования к объему оперативной памяти кодирующего устройства, входящего в состав бортовой передающей аппаратуры. Объем требуемой памяти возрастает пропорционально квадрату периода вращения летательного аппарата. Рассматриваемый метод чувствителен к ширине и количеству глубоких провалов более 10 дБ в диаграмме направленности антенной решетки бортовой передающей аппаратуры. При наличии глубоких провалов в диаграмме направленности резко снижается достоверность передаваемой информации. Дополнительным ограничением второго метода является необходимость измерения коэффициентов канальной матрицы канала связи с частотой, зависящей от характерной скорости перемещения летательного аппарата и требование наличия соответствующего обратного канала связи.

Исследование выполнено при поддержке гранта Российского научного фонда (проект 14-19-01263) в Рязанском государственном радиотехническом университете.

Литература

1. Булгаков Н.Н., Альбин В.Г., Кривошеин А.А. Особенности построения бортовой аппаратуры командно-измерительной системы космического аппарата для управления им как в зоне его радиовидимости с наземной станцией, так и вне ее // *Материалы 24-й Международной Крымской конференции «СВЧ-техника и телекоммуникационные технологии» (КрыМиКо'2014)*. Севастополь: Вебер, 2014. – С.6-9
2. Паршин Ю.Н., Кудряшов В.И., Кудряшова А.А. Особенности разработки ММО канала связи с беспилотными аппаратами // *Материалы 24-й Международной Крымской конференции «СВЧ-техника и телекоммуникационные технологии» (КрыМиКо'2014)*. Севастополь: Вебер, 2014. – С. 253-254
3. Марков Г.Т., Сазонов Д.М. *Антенны: Учебник для студентов радиотехнических специальностей ВУЗов*. М.: Энергия, 1975
4. Паршин Ю.Н., Ксендзов А.В. Влияние пространственной корреляции на эффективность оптимизации пространственной структуры многоантенной системы при разнесенном приеме // *Вестник Рязанской государственной радиотехнической академии*, вып. 19. Рязань: Изд. РГРТУ, 2006. – С. 54-62.
5. Kuhn V. *Wireless Communications over MIMO channels. Applications to CDMA and multiple antenna systems* – John Wiley & Sons, Ltd. 2006.
6. Шлома А.М., Бакулин М.Г., Крейнделин В.Б., Шумов А.П. Новые алгоритмы формирования и обработки сигналов в системах подвижной связи / Под ред. проф. А.М. Шломы. – М.: Горячая линия-Телеком, 2008. – 244 с.
7. Колесников Е.П. Особенности расчета энергетики радиолиний измерительных систем при пусках ракет // *Радиотехника*, 2004, №6, с. 70-80.

ПРИМЕНЕНИЕ МНОГОПороГОВОГО ДЕКОДИРОВАНИЯ ДЛЯ КОРРЕКЦИИ ОШИБОК В БЕСПРОВОДНЫХ КАНАЛАХ СВЯЗИ

Овечкин Г.В., д.т.н., доцент кафедры ВПМ Рязанского государственного радиотехнического университета (РГРТУ), e-mail: g_ovechkin@mail.ru;

Шевляков Д.А., аспирант кафедры ВПМ РГРТУ, e-mail: dima-shevlyakov@yandex.ru.

USING OF MULTITHRESHOLD DECODING FOR ERROR CORRECTION IN WIRELESS CHANNELS

Ovechkin G.V., Shevlyakov D.A.

The article deals with multithreshold decoders (MTD) for self-orthogonal codes. The lower bounds for multithreshold decoders bit error probability in uncorrelated Rayleigh and Rician channels are presented. The simulation results show these bounds are tight enough in field of suboptimal MTD performance. A using MTD with OFDM and MIMO technologies with precoding over wireless fading channels is discussed. It's shown MTD with precoding provides significant performance improvement in such channels.

Key words: multithreshold decoders, self-orthogonal codes, bit error probability, precoding, wireless fading channels.

Ключевые слова: помехоустойчивое кодирование, самоортогональные коды, многопороговые декодеры, энергетический выигрыш кодирования, канал связи, прекодирование, MIMO, OFDM.

Введение

Помехоустойчивое кодирование или кодирование с прямым исправлением ошибок применяется в системах связи, в которых отсутствует или недоступен обратный канал для передачи запросов на повторную передачу, задержки в канале при запросах повторной передачи оказываются недопустимо большими или, наконец, уровень помех настолько велик, что количество повторных передач становится чрезвычайно большим. В настоящее время существуют множество кодов и алгоритмов их декодирования, например, блочные коды Боуза-Чоудхури-Хоквигема, коды Рида-Соломона, сверточные коды, коды с низкой плотностью проверок на четность (Low-Density Parity-Check, LDPC), каскадные коды и турбо коды [1]. Новое и очень эффективное решение проблемы высокой сложности декодирования при одновременной реализации высоких энергетических характеристик систем кодирования на базе многопороговых декодеров (МПД) самоортогональных кодов (СОК) предлагается российскими специалистами [2, 3].

В настоящее время характеристики МПД широко исследованы для каналов с независимыми ошибками, в которых данные методы обеспечивают близкое по эффективности к оптимальному декодирование даже очень длинных кодов всего лишь с линейной от длины кода сложностью реализации [2, 3]. Для таких каналов известен ряд способов улучшения эффективности МПД, позволяющих приблизить область эффективной работы МПД к пропускной способности канала. В результате, как аппаратные, так и программные версии МПД оказываются способными обеспечивать уровни энергетиче-

Рассмотрены многопороговые алгоритмы декодирования самоортогональных помехоустойчивых кодов. Представлены аналитические оценки их эффективности в некоррелированном релейском и райсовском канале и получены результаты компьютерного моделирования. Рассмотрены возможности применения многопороговых декодеров совместно с такими технологиями повышения эффективности систем передачи данных в беспроводных каналах с замираниями, как OFDM, MIMO, прекодирование. Показано, что применение прекодирования в таких условиях обеспечивает существенное улучшение энергетических характеристик системы передачи данных.

ского выигрыша, сопоставимые с выигрышем лучших методов декодирования турбо и низкоплотностных кодов при в десятки раз более высоком быстродействии [4]. Вместе с тем в соответствии с теорией кодирования, эффективность МПД в гауссовских каналах может быть еще несколько улучшена. Кроме того, перспективные сети связи работают в гораздо более сложных условиях, возникающих из-за многолучевого распространения сигналов, доплеровского сдвига и многих других причин. В результате возникающие в канале ошибки группируются в пакеты. В таких условиях эффект от применения кодирования оказывается много больше, чем в каналах с независимыми ошибками, поскольку здесь, в ряде случаев, только за счет повышения мощности передатчика вообще невозможно увеличить достоверность передаваемых данных. Следовательно, разработка алгоритмов повышения эффективности многопороговых декодеров в каналах с группирующимися ошибками, **являющаяся целью данной работы**, позволит повысить энергетический выигрыш от применения кодирования, который можно использовать для улучшения технических характеристик систем передачи данных и значительно расширит область применения данного метода. При этом особенно важным является сохранение или лишь незначительное увеличение сложности реализации исходного многопорогового декодера, поскольку только самые простые методы коррекции оши-

бок смогут обеспечить уже требуемые в настоящее время скорости декодирования цифровых потоков, составляющие десятки Гбит/с.

Многopороговые декодеры

Многopороговые декодеры используются для декодирования блочных или сверточных самоортogonalных кодов. Основные принципы работы МПД, используемого для декодирования блочного СОК с кодовой скоростью 1/2 и длиной 34 бита, заданного образующим полиномом $g(x) = 1 + x^3 + x^7 + x^8$, иллюстрируются схемой, представленной на рис. 1 [2]. Отметим, что в состав МПД входят регистры, полусумматоры и пороговый элемент, суммирующий свои входы и сравнивающий полученную сумму с порогом. Это делает МПД простейшим для реализации устройством, способным обеспечить предельно возможные скорости декодирования.

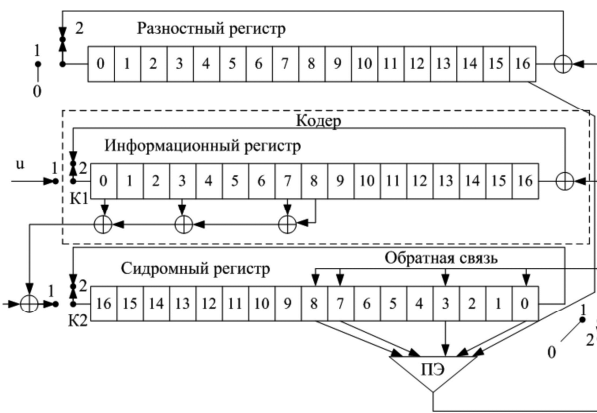


Рис. 1. Схема МПД блочного СОК

При работе МПД в двоичном симметричном канале на пороговом элементе для текущего декодируемого символа u_j вычисляется функция правдоподобия L_j , зависящая от элементов синдрома s_{jk} и соответствующего элемента разностного регистра d_j :

$$L_j = \sum_{s_{jk} \in \{S_j\}} s_{jk} + d_j, \tag{1}$$

где $\{S_j\}$ – множество проверок (элементов синдрома) относительно ошибки e_j в декодируемом символе u_j . При превышении функцией правдоподобия некоторого порога (в общем случае равного половине числа слагаемых) осуществляется изменение декодируемого символа, соответствующего элемента разностного регистра и участвующих в вычислении функции правдоподобия элементов синдрома.

При работе МПД в гауссовском канале для текущего символа u_j также вычисляется функция правдоподобия L_j . При этом элементы синдрома и элемент разностного регистра суммируются с некоторыми коэффициентами, отражающими их надежность:

$$L_j = \sum_{s_{jk} \in \{S_j\}} (2s_{jk} - 1)w_{jk} + (2d_j - 1)w_j, \tag{2}$$

где w_{jk} – коэффициент, отражающий надежность проверки s_{jk} ; w_j – коэффициент, отражающий надежность принятого символа u_j . В качестве оценок надежности принятых из канала символов можно использовать,

например, логарифм отношения правдоподобия

В [2, 3] показано, что МПД с каждым изменением декодируемого символа приближает собственное решение к решению оптимального (по максимуму правдоподобия) декодера. Это объясняется тем, что при изменении символа суммарный вес синдрома и разностного вектора обязательно уменьшается и, таким образом, происходит переход к кодовому слову, которое более правдоподобно, чем кодовое слово, находившееся в МПД в предыдущий момент времени. Однако ввиду возникновения эффекта размножения ошибок [2, 3] процесс перехода от одного кодового слова к другому, более правдоподобному, может прекратиться до того, как МПД достигнет решения оптимального декодера. Данный эффект заключается в том, что после совершения декодером ошибки через ветви обратной связи в синдром попадает большое число ошибок декодера, которые мешают правильному декодированию информационных символов на последующих итерациях. Это приводит к тому, что вероятность второй ошибки декодирования существенно возрастает, и появляются пакеты ошибок. Следовательно, для того чтобы вероятность исправления ошибок МПД была близка к решению оптимального декодера, необходимо использовать коды, в максимальной степени устойчивые к размножению ошибок [2, 3]. При построении таких кодов можно использовать методику оценки размножения ошибок, описанную в [2].

Аналитическая оценка вероятности ошибки МПД в релейских и райсовских каналах

Простейшими моделями каналов с замираниями являются модели каналов с релейскими и райсовскими замираниями, которые получаются при наличии множества путей распространения сигнала. При этом в релейском канале отсутствует прямая видимость между передатчиком и приемником, а в райсовском канале присутствует.

Для оценки возможностей МПД в таких каналах получим нижнюю границу вероятности ошибки декодирования. Для простоты предположим, что используется модуляция типа BPSK (или QPSK) и демодулятор формирует только жесткие решения относительно принятых битов. Отметим, что для МПД нижняя оценка вероятности ошибки в двоичном симметричном канале, в котором каждый бит независимо от других искажается с вероятностью p , определяется вероятностью ошибки оптимального декодера (ОД), вычисляемой в соответствии с выражением [2, 3]:

$$P_b = \sum_{i=(d+1)/2}^d C_d^i p^i (1-p)^{d-i}, \tag{3}$$

где d – кодовое расстояние используемого СОК (предполагается нечетным); p – вероятность ошибки в канале.

Для некоррелированного релейского канала известно выражение для вероятности ошибки [5]

$$p = \frac{1}{2} \left(1 - \sqrt{\frac{E_s/N_0}{1+E_s/N_0}} \right), \tag{4}$$

где E_s/N_0 – символьное (безразмерное) отношение сигнал/шум. Данная оценка справедлива, когда при пере-

даче используются модуляция типа BPSK и частота доплеровского сдвига $F_d = 0$. В случае канала с райсовскими замираниями вероятность ошибки в соответствии с [5] определяется как

$$p = 2Q\left(\sqrt{\frac{2kE_s/N_0}{k+E_s/N_0}}\right), \quad (5)$$

где k – коэффициент Райса, а функция $Q(x)$ определяется как

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-\frac{t^2}{2}} dt. \quad (6)$$

Учитывая, что МПД способен работать почти как ОД даже при достаточно высоком уровне шума в канале, подставляем (4) в (3) и получаем нижнюю оценку вероятности битовой ошибки для МПД в релейском канале:

$$P_b = \sum_{i=(d+1)/2}^d C_d^i \frac{1}{2} \left(1 - \sqrt{\frac{E_s/N_0}{1+E_s/N_0}}\right)^i \times \left(1 - \frac{1}{2} \left(1 - \sqrt{\frac{E_s/N_0}{1+E_s/N_0}}\right)\right)^{d-i}. \quad (7)$$

Аналогично вероятность битовой ошибки в райсовском канале определяется как

$$P_b = \sum_{i=(d+1)/2}^d C_d^i Q\left[\sqrt{\frac{2kE_s/N_0}{k+E_s/N_0}}\right]^i \times \left(1 - Q\left[\sqrt{\frac{2kE_s/N_0}{k+E_s/N_0}}\right]\right)^{d-i}. \quad (8)$$

Отметим, что в таком виде выражения позволяют получить оценку вероятности ошибки МПД при использовании СОК с нечетным кодовым расстоянием.

На рис. 2 кривыми 5 и 6 показаны оценки вероятности ошибки в некоррелированном релейском канале и вероятности ошибки после декодера для СОК с минимальным кодовым расстоянием $d=9$. Отметим, что по сравнению с гауссовским каналом в обсуждаемых условиях наблюдаются потери в энергетике порядка 1,5 дБ. На этом же рисунке кривыми 1 и 2 показаны оценки вероятности ошибки в канале после декодера, полученные путем компьютерного моделирования при использовании блочного СОК с длиной 20000 бит, кодовой скоростью $R=1/2$ и минимальным кодовым расстоянием $d=9$ и МПД с 15 итерациями декодирования. Отметим, что экспериментальная оценка вероятности ошибки в канале хорошо согласуется с аналитической, что позволяет использовать последнюю в полученных формулах. Кроме того, видно, что предложенная аналитическая оценка вероятности ошибки декодирования при использовании МПД в некоррелированном релейском канале оказывается достаточно точной для области эффективной работы МПД. Аналогичные результаты для райсовского канала при $k=5$ показаны на рис. 2 кривыми 7, 8 для аналитических оценок и 3, 4 для результатов моделирования. И в данном случае полученная оценка является хорошей для области эффективной работы МПД.

Отметим, что при наличии коррелированных замираний или использовании модуляции более высокого

порядка для большого шума получить аналитические оценки с приемлемой точностью не представляется возможным. Поэтому в таких условиях для оценки эффективности МПД следует использовать компьютерное моделирование.

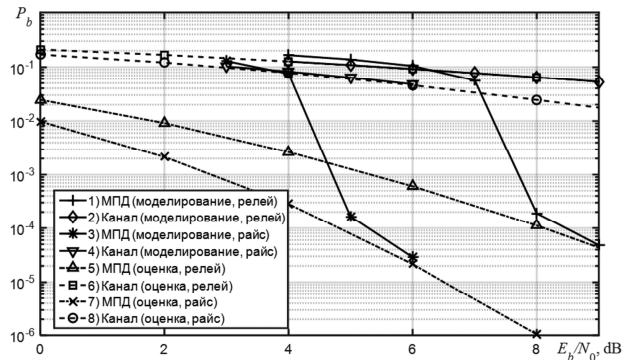


Рис. 2. Аналитические оценки вероятности ошибки и результаты моделирования

для МПД в релейском и райсовском канале

Все результаты в данном разделе получены в рамках гранта РФФИ №15-37-50568.

Результаты моделирования многопороговых декодеров в релейском и райсовском каналах

На рис. 3 кривыми 2, 4 и 6 показаны результаты моделирования для райсовских и релейских замираний при доплеровской частоте 100 и 150 Гц и использовании модуляции типа BPSK и демодулятора, формирующего жесткие решения. Здесь использовался МПД для того же кода, что и на ранее представленном рисунке. Отметим, что наличие коррелированных замираний существенно (для релейских замираний на 6 дБ) ухудшает рабочее отношение сигнал/шум. Это объясняется тем, что в данном случае использовался код с параллельным каскадированием и данные из кодера выдавались через модулятор в канал «по строкам» без использования перемежителя. В результате при возникновении замираний могли искажаться целые ветви с большей размерностью проверок, что приводило к появлению после декодирования блоков со значительным числом ошибок. Для улучшения эффективности МПД следует либо использовать дополнительный перемежитель, что не всегда удобно, либо организовать внутреннее перемежение, выдавая данные с кодера «по столбцам». Кривыми 1, 3 и 5 на рис. 3 представлены результаты моделирования, соответствующие предлагаемой параллельной передаче бит. Из анализа графиков видно, что в случае использования передачи бит «по столбцам» в канале с райсовскими замираниями уменьшение вероятности ошибки декодирования по сравнению с первоначальным вариантом, например, при отношении сигнал/шум равном 7 дБ при $k=5$ и $F_d=150$, оказывается большим двух десятичных порядков. Выигрыш наблюдается и для других условий моделирования. Это объясняется тем, что при такой параллельной передаче ошибки, вызванные замираниями сигнала, оказываются разнесенными по разным ветвям. При этом ветви с большой размерностью проверок искажаются гораздо меньше, и МПД оказывается способным обеспечить меньшую вероятность

ошибки декодирования.

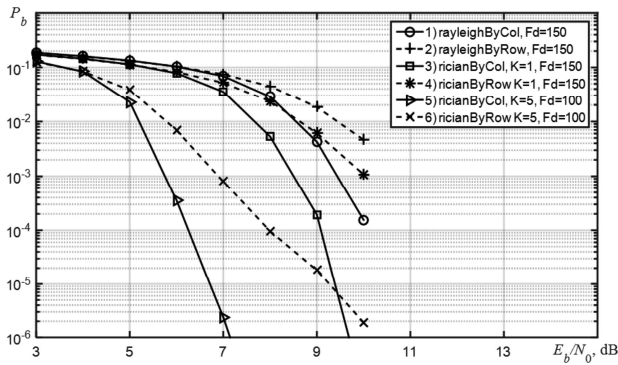


Рис. 3. Характеристики МПД в канале с замираниями при последовательной и параллельной передаче бит

Работа МПД в беспроводных системах связи с предварительным кодированием

Следует отметить, что рассмотренные случаи применения МПД позволяют успешно бороться только с замираниями, при которых нет межсимвольной интерференции. Если же присутствует межсимвольная интерференция, то для борьбы с такими замираниями необходимо использовать дополнительные средства, например мультиплексирование с ортогональным частотным разделением (OFDM). Кроме того, из анализа представленных в [6] графиков видно, что для каналов с замираниями свойственно значительно более медленное уменьшение вероятности ошибки с ростом отношения сигнал/шум по сравнению с гауссовским каналом. Это объясняется тем, что при длительном замирании даже при большом отношении сигнал/шум на входе декодера появляется большой пакет ошибок, с которым он справиться не в состоянии. Для уменьшения влияния данного эффекта часто используется технология пространственного разнесения, когда в системе связи используются несколько передающих и несколько приемных антенн (технология Multiple Input Multiple Output – MIMO). При правильном построении такой системы можно считать, что получающиеся при этом подканалы оказываются независимыми. Следовательно, вероятность того, что все подканалы одновременно будут подвержены замираниям будет много меньше по сравнению с одноканальной системой. Это позволяет существенно улучшить достоверность передачи данных. Отметим, что при использовании технологии MIMO появляется дополнительная возможность выполнять пространственно-временное кодирование (STC – Space-Time Coding) [7] – кодирование, выполняющее отображение передаваемых символов сигнального созвездия на передающие антенны, использование которого позволяет существенно улучшить характеристики системы, использующей МПД [8, 9].

Так же отметим, что полученные ранее результаты предполагали, что на приемной стороне известно точное состояние канала, т.е. известна мультипликативная составляющая шума для каждого передаваемого символа. На практике для оценки канала обычно применяются пилот символы, добавляемые в передаваемый сигнал. Демодулятор использовал эту точную инфор-

мацию при оценке принятых из канала символов. В некоторых случаях такая информация известна и на передающей стороне (например, в том случае, если канал меняется достаточно медленно, можно организовать передачу этой информации по обратному каналу). Передатчик может использовать информацию о состоянии канала для перераспределения энергии сигнала по пространственным каналам, загружая «хорошие» каналы и освобождая «плохие». Данные действия выполняются с помощью так называемого предварительного кодирования (прекодирования) передаваемого сигнала. Рассмотрим процесс прекодирования более подробно.

В случае использования MIMO архитектуры передатчика, когда каждая антенна передает свой символ, модель канала описывается выражением

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}, \quad (7)$$

где $\mathbf{x} \in \mathbb{C}^M$ – передаваемый вектор из M комплексных значений; $\mathbf{n} \in \mathbb{C}^N$ – вектор некоррелированного комплексного гауссовского шума; $\mathbf{H} = (\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_N) \in \mathbb{C}^{M \times N}$ – комплексная матрица канала; M – число передающих антенн; N – число приемных антенн. Отметим, что величины $h_{i,j}$ являются случайными, но для обсуждаемого варианта они точно известны и на приемной и на передающей стороне.

Для такой модели канала вектор информационных символов $\mathbf{s} \in \mathbb{C}^L$ из некоторого алфавита прекодируется с помощью линейной операции

$$\mathbf{x} = \mathbf{B}\mathbf{s}, \quad (8)$$

где $\mathbf{B} \in \mathbb{C}^{M \times L}$ – комплексная матрица прекодера; $L \leq \text{rank}(\mathbf{H})$ – число активных каналов.

Оптимальный MMSE приемник для такого канала и прекодера представляет собой винеровский фильтр с матрицей

$$\mathbf{G} = \mathbf{B}^* \mathbf{H}^* (\mathbf{H} \mathbf{B} \mathbf{B}^* \mathbf{H}^* + \sigma_n^2 \mathbf{I}_N)^{-1}, \quad (9)$$

где $(\cdot)^*$ – комплексное сопряжение; σ_n^2 – дисперсия гауссовского шума; \mathbf{I}_N – единичная матрица размером $N \times N$.

Данный фильтр на выходе дает оптимальную оценку переданных символов

$$\hat{\mathbf{s}} = \mathbf{G}\mathbf{y}. \quad (10)$$

Существует большое число различных алгоритмов прекодирования, различающихся, в том числе, сложностью оптимального приемника. Несколько интересных оптимальных по некоторому критерию прекодеров, приемник для которых имеет линейную сложность, предложены в работе [10].

Первый из этих алгоритмов минимизирует суммарную среднеквадратическую ошибку на выходе приемника. Для такого прекодера матрица прекодирования имеет вид

$$\mathbf{B} = \tilde{\mathbf{V}}\mathbf{\Phi}^{1/2}, \quad (11)$$

где $\tilde{\mathbf{V}} \in \mathbb{C}^{M \times L}$ – комплексная матрица, составленная из первых L столбцов унитарной матрицы, входящей в сингулярное разложение симметричной матрицы $\mathbf{H}\mathbf{H}^*$, в котором все собственные числа λ_{mm} диагональной матрицы $\Lambda = \text{diag}(\lambda_{11}, \lambda_{22}, \dots, \lambda_{NN})$ неотрицательны и упорядочены по убыванию; $\mathbf{\Phi} \in \mathbb{R}^{L \times L}$ – диагональная матрица, состоящая из L неотрицательных чисел

$$\phi_{ii} = \left(\frac{P_0 + \sigma_n^2 \sum_{n=1}^{\bar{L}} \lambda_{nm}^{-1}}{\sum_{n=1}^{\bar{L}} \lambda_{nm}^{-1/2}} \lambda_{ii}^{-1/2} - \frac{\sigma_n^2}{\lambda_{ii}} \right)^+, \quad i = 1, \dots, L, \quad (12)$$

где $(x)^+ = \max(x, 0)$, $\bar{L} \leq L$ выбирается так, что $\phi_{nn} > 0$ для $n \in [1, \bar{L}]$ и $\phi_{nn} = 0$ для всех других n ; P_0 – средняя мощность передаваемого сигнала.

Второй прекодер максимизирует взаимную информацию между переданными и принятыми данными. В отличие от первого варианта здесь элементы матрицы $\Phi \in \mathbb{R}^{L \times L}$ определяются следующим образом

$$\phi_{ii} = \left(\frac{P_0 + \sigma_n^2 \sum_{n=1}^{\bar{L}} \lambda_{nm}^{-1}}{\bar{L}} \lambda_{ii}^{-1} - \frac{\sigma_n^2}{\lambda_{ii}} \right)^+ \quad i = 1, \dots, L. \quad (11)$$

Далее рассмотрим эффективность применения данных алгоритмов совместно с МПД. При моделировании использовался МПД с 30 итерациями декодирования для построенного СОК с кодовой скоростью $R = 8/16$, кодовым расстоянием 17 и длиной 43200 битов. Совместно с МПД применялось OFDM мультиплексирование с 1024 несущими. Защитный интервал составлял 1/16 от длины OFDM символа. В качестве модуляции использовалась обычная QPSK. При получении результатов использовалась модель канала Spatiat Channel Model типа Urban micro [11]. Максимальная доплеровская частота F_d была равна 0. На рис. 4 показаны характеристики МПД в вышеописанных условиях при использовании различного числа передающих и приемных антенн и различных алгоритмов прекодирования.

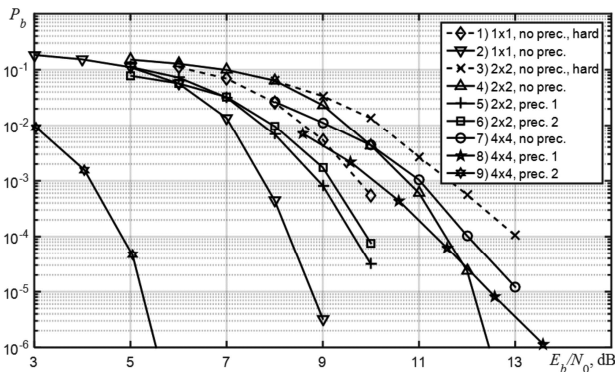


Рис. 4. Характеристики МПД для разного числа приемных и передающих антенн и различных алгоритмов прекодирования

На данном рисунке при получении кривых 1 и 3 демодулятор формировал только жесткие решения, а при получении остальных кривых оценивал и надежность решений. Отметим, что переход к мягким решениям демодулятора обеспечивает увеличение энергетического выигрыша кодирования до 2 дБ по сравнению с использованием только жестких решений. Также заметим, что 2x2 вариант по энергетике оказывается хуже одноканального на 3 дБ, но при этом спектральная эффективность системы оказывается в два раза лучше.

Использование прекодирования вместе с двумя передающими антеннами уменьшает проигрыш по энергетике до 1 дБ. Интересные результаты наблюдаются для четырех передающих антенн. Здесь при использовании второго прекодера результаты существенно улучшаются. Это оказывается возможным за счет того, что более эффективно по сравнению с первым прекодером отключаются «плохие» пространственные каналы (например, при отношении сигнал/шум 5 дБ используется в среднем 2,5 пространственных канала из четырех, а для первого прекодера 3,5 канала). В результате энергетические характеристики оказываются лучше одноканального варианта более, чем на 3,5 дБ при почти в 2,5 раза лучшей спектральной эффективности.

Таким образом, полученные результаты позволяют утверждать, что за счет применения ММО и прекодирования можно существенно улучшить эффективность МПД в многолучевых каналах с замираниями по сравнению с одноканальным вариантом. Отметим, что полученные в данном разделе результаты получены в рамках НИР, выполненной при поддержке Российского научного фонда (проект №14-19-01263).

Заключение

В работе получены новые результаты исследования эффективности МПД в ряде многолучевых каналах связи с Рейсовскими и Релеевскими замираниями. Для этих каналов даны рекомендации по наилучшему согласованию кодера и канала, для некоррелированного релеевского и райсовского каналов получены нижние оценки вероятности ошибки многопорогового декодирования. Выполнено исследование эффективности применения прекодирования совместно с многопороговыми декодерами и технологией ММО.

Анализ результатов, представленных в работе, показал, что МПД, обеспечивающие высокую эффективность в гауссовских каналах, оказываются способными бороться с ошибками и в значительно более сложных условиях применения. При этом эффект от применения помехоустойчивого кодирования в данных случаях многократно превышает эффект от применения кодирования в гауссовском канале.

Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (гранты №14-07-00859 и №15-37-50568), гранта Президента РФ (грант МД-639.2014.9) и Российского научного фонда в Рязанском государственном радиотехническом университете (проект №14-19-01263). Большой объем дополнительной информации о МПД можно найти на веб-сайтах [12].

Литература

1. Зубарев Ю.Б., Овечкин Г.В. Помехоустойчивое кодирование в цифровых системах передачи данных // Электросвязь. М., 2008. №12. С. 2–11.
2. Золотарёв В.В., Зубарев Ю.Б., Овечкин Г.В. Многопороговые декодеры и оптимизационная теория кодирования. – М.: Горячая линия – Телеком, 2012. 239 с.

3. Золотарёв В.В., Овечкин Г.В. Повышение надежности передачи и хранения данных с использованием многопороговых методов декодирования помехоустойчивых кодов // Цифровая обработка сигналов, 2012, №1, С.16–21.
4. Золотарёв В.В., Назиров Р.Р., Чулков И.В., Овечкин Г.В. Алгоритмы МПД // Российский космос. 2009. №1. С. 60–63.
5. Viswanathan M. Simulation of Digital Communication Systems Using Matlab [eBook] – Second Edition, 2013.
6. Овечкин Г.В., Шевляков Д.А. Эффективность многопороговых методов коррекции ошибок в каналах связи с замираниями // Успехи современной радиоэлектроники, М.: Радиотехника, 2014. №6, С.37–43.
7. Giosic S. Advanced wireless communications. 4G technologies // S. Giosic Wiley & Sons. 2004. – 878 p.
8. Овечкин Г.В., Шевляков Д.А. Эффективность применения многопороговых декодеров с пространственно-временным кодированием // Математическое и программное обеспечение вычислительных систем: Межвуз. сб. науч. тр./под ред. Пылькина – Рязань: РГРТУ, 2013. С.115–121.
9. Золотарев В.В., Овечкин Г.В., Шевляков Д.А. Исследование эффективности многопороговых декодеров при совместном использовании с пространственно-временным кодированием // Материалы 17-й Международной конференции «Цифровая обработка сигналов и ее применение». Москва, 2015. С. 83–87.
10. Scaglione A., Stoica P., Barbarossa S., Giannakis G.B., Sampath H. Optimal designs for space-time linear precoders and decoders // IEEE Trans. Signal Process., vol. 50, no. 5, pp. 1051–1064, May 2002.
11. 3GPP TR 25.996. Spatial channel model for Multiple Input Multiple Output (MIMO) Simulations.
12. Веб-сайты www.mtdbest.ru и www.mtdbest.iki.rssi.ru.

КОРРЕЛЯЦИОННЫЕ СВОЙСТВА КАНАЛЬНЫХ КОЭФФИЦИЕНТОВ МИМО СИСТЕМЫ СВЯЗИ С МАНЕВРИРУЮЩИМ ОБЪЕКТОМ

*Паршин Ю.Н., д.т.н., проф., заведующий кафедрой РТУ РГПТУ, e-mail: parshin.y.n@rsreu.ru;
Кудряшов В.И., аспирант кафедры РТУ РГПТУ, e-mail: yachmen@mail.ru.*

CORRELATION PROPERTIES OF MIMO COMMUNICATION SYSTEM CHANNEL COEFFICIENTS WITH MANEUVERING OBJECT

Parshin Y.N., Kudryashov V.I.

The analysis of the correlation coefficients of the MIMO channel communication system with maneuvering objects indispensable to the development of optimal algorithms for estimating channel coefficients for different models of maneuvering motion of the object is performed. The results allow reasonable estimation algorithm selected channel coefficients, which reduces the error of estimation.

Key words: MIMO channel, matrix of channel coefficients, maneuvering object, measurement error, correlation analysis.

Ключевые слова: МИМО канал, матрица канальных коэффициентов, маневрирующий объект, ошибка измерения, корреляционный анализ.

Введение

В настоящее время наблюдается интенсивный рост применения маневрирующих объектов, например, беспилотных аппаратов (БПА) для получения и передачи информации. В этом случае к пропускной способности канала связи предъявляются высокие требования. Для повышения скорости передачи данных в условиях замираний и многолучевости в системах связи с БПА применяют технологию МИМО. Канал передачи данных построенный на основе данной технологии обеспечивает высокую пропускную способность за счет пространственно-временного кодирования и декодирования с использованием нескольких антенн на приемной и передающей стороне.

Одной из проблем применения МИМО каналов для связи с маневрирующими объектами является большая вариабельность во времени матрицы канальных коэффициентов (МКК) при движении объекта. В работах [1, 2] проведен анализ влияния ошибок измерения МКК на пропускную способность МИМО канала связи. Установлено, что движение объекта вызывает дополнительную динамическую ошибку оценивания МКК. Поэтому важно определить характер изменения канальных коэффициентов МИМО системы передачи информации в различных типовых условиях.

Целью работы является анализ корреляции канальных коэффициентов МИМО системы связи с маневрирующим объектом, необходимый для разработки оптимальных алгоритмов оценивания МКК, при различных моделях движения маневрирующего объекта.

Постановка задачи. В статье используются различные модели движения маневрирующего объекта.

Модель № 1. В качестве одной из математических моделей движения может быть принята модель Дубинса [3].

Проводится анализ корреляции канальных коэффициентов МИМО системы связи с маневрирующим объектом, необходимый для разработки оптимальных алгоритмов оценивания канальных коэффициентов, при различных моделях движения маневрирующего объекта. Получены результаты, позволяющие обоснованно выбрать алгоритм оценивания канальных коэффициентов, что уменьшает ошибку оценивания.

са [3]. Данная модель описывает детерминированное движение объекта в непрерывном времени, обеспечивающее оптимальное по быстродействию движение и управление: кратчайший путь, соединяющий две точки в двумерной евклидовой плоскости с ограничением на кривизну траектории.

В работе [3] показано, что любой траектория Дубинса состоит из дуг окружности максимального радиуса кривизны поворота и (или) прямых отрезков. В случае двумерной модели, объект перемещается по плоскости и именуется машиной Дубинса, которая широко используется в качестве одной из базовых моделей движения маневрирующего аппарата. Движение такого объекта описывается следующей системой уравнений в дискретном времени:

$$\begin{aligned}x_n &= x_{n-1} + \Delta t V_{n-1} \cos \theta_{n-1}, \\y_n &= y_{n-1} + \Delta t V_{n-1} \sin \theta_{n-1}, \\ \theta_n &= \theta_{n-1} + \Delta t u,\end{aligned}\tag{1}$$

где $(x, y) \in \mathbb{R}^2$ – координаты базисной точки объекта на плоскости, θ_n – угол между осью абсцисс и вектором скорости объекта, u – угловая скорость объекта, V_n – линейная скорость объекта в собственной системе координат, Δt – дискрет изменения времени.

В работе [4] представлена дискретная модель Дубинса, в которой траектория движения БПА аппроксимируется отрезками прямой. Приводится доказательство того, что кратчайшие полигональные пути с ограниченной кривизной поворота объекта имеют такую же структуру, как и у путей Дубинса.

Модель № 2. Описанная выше модель является детерминированной, т.е. в ней отсутствует случайная составляющая движения. В работе [5] предлагается для учета флуктуаций управляющих воздействий и турбулентности среды движения ввести шумовую составляющую в некоторые параметры модели движения. Например, в системе уравнений такими параметрами могут быть линейная скорость V , угловая скорость u . Уравнения, описывающие флуктуации этих параметров в дискретном времени имеют вид:

$$V_n = \left(1 - \frac{\Delta t}{T_V}\right) V_{n-1} + \frac{k_1}{T_V} \delta_n, \quad (2)$$

где коэффициенты T_V, k_1 характеризуют скорость и интенсивность флуктуаций, δ_n – стандартный белый шум. Уравнение для угловой скорости имеет аналогичный вид с учетом обозначений.

Модель № 3. В работе [6] приведены уравнения движения маневрирующего объекта, записанных в виде переменных состояния. Невозмущенная траектория объекта задается уравнением в дискретном времени:

$$\mathbf{x}_n = \Phi \mathbf{x}_{n-1}, \quad (3)$$

где $\mathbf{x}_n = \left\{ x_n, \frac{dx_n}{dt}, y_n, \frac{dy_n}{dt} \right\}$ – координаты объекта и их первые производные,

$$\Phi = \begin{pmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Неманеврирующий объект имеет приблизительно постоянную скорость, которая незначительно флуктуирует. Уравнение движения неманеврирующего объекта на плоскости содержит случайные возмущения в виде белого шума $\boldsymbol{\eta}_n = \{\eta_1, \eta_2\}$:

$$\mathbf{x}_n = \Phi \mathbf{x}_{n-1} + \mathbf{B} \boldsymbol{\eta}_n, \quad (4)$$

где $\mathbf{B} = \begin{pmatrix} \Delta t / 2 & 0 \\ 1 & 0 \\ 0 & \Delta t / 2 \\ 0 & 1 \end{pmatrix}.$

Маневрирующий объект имеет приблизительно постоянное ускорение, которое незначительно флуктуирует:

$$\mathbf{x}_n = \left\{ x_n, \frac{dx_n}{dt}, y_n, \frac{dy_n}{dt}, \frac{d^2x_n}{dt^2}, \frac{d^2y_n}{dt^2} \right\}.$$

Коэффициенты уравнения для маневрирующей цели имеют вид:

$$\Phi = \begin{pmatrix} 1 & \Delta t & 0 & 0 & \Delta t^2 / 2 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & \Delta t & 0 & \Delta t^2 / 2 \\ 0 & 0 & 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

$$\mathbf{B} = \begin{pmatrix} \Delta t^2 / 4 & 0 \\ \Delta t / 2 & 0 \\ 0 & \Delta t^2 / 4 \\ 0 & \Delta t / 2 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Разработаны также модели движения [7], более точно отражающие движение объекта, учитывающие его свойства управляемости а также турбулентность воздушных масс. Однако, большой объем априорных сведений, необходимых для расчета траектории движения, затрудняют их применение и получение общих результатов.

Модель распространения сигнала. Описание распространения сигнала в MIMO канале основано на геометрической однокольцевой модели отражения (рис. 1) [8].

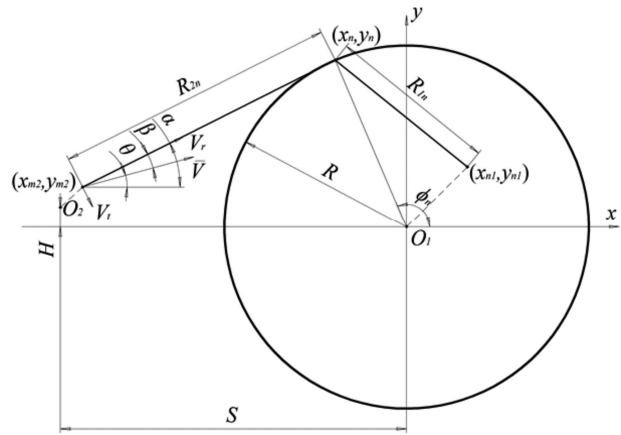


Рис. 1. Однокольцевая геометрическая модель

Прохождение сигнала от передающей антенны к приемной антенне можно представить в виде линейного преобразования амплитуды и фазы сигнала. Комплексные коэффициенты преобразования образуют канальную матрицу

$$\underline{\mathbf{H}} = \{ \underline{H}_{im} = H_{im} e^{j\psi_{im}}, i = 1, \dots, N_R, m = 1, \dots, N_T \},$$

где H_{im} – коэффициент передачи амплитуды сигнала, ψ_{im} – фазовый сдвиг сигнала, N_R, N_T – число приемных и передающих антенн соответственно.

Наземная антенная система работает в режиме приема и содержит N_R элементов с координатами x_{n1}, y_{n1} в системе координат O_1 , связанной с наземной антенной системой. Антенная система маневрирующего объекта работает в режиме передачи и содержит N_T элементов с координатами $x_{m2}, y_{m2}, m = 1, \dots, N_T$, в системе координат O_2 , связанной с маневрирующим объектом. Начало системы координат O_2 располагается в точке (S, H) , характеризующей положение маневрирующего объекта, а ее зависимость от времени задает траекторию его движения. Каждый из отражателей располагается на окружности радиусом R с центром в точке O_1 ; координаты отражателей равны: $x_n = R \cos \phi_n, y_n = R \sin \phi_n$. Угловая координата отражателей $\phi_n = n \frac{2\pi}{N}, n = 1, \dots, N$, соответствует

равномерному распределению отражателей по окружности, N – количество отражателей.

В соответствии с принятой моделью коэффициент канальной матрицы определяется следующим выражением:

$$\underline{H}(t) = \lim_{N \rightarrow \infty} \frac{1}{\sqrt{N}} \sum_{n=1}^N a_n b_n e^{j(2\pi f_n t + \theta_n)}, \quad (6)$$

где $a_n = e^{j2\pi \frac{R_{1n}}{\lambda}}$ – коэффициент, определяющий сдвиг фазы сигнала на участке от передающих антенн

до одного из отражателей; $b_n = e^{j2\pi \frac{R_{2n}}{\lambda}}$ – коэффициент, определяющий сдвиг фазы сигнала на участке от кольца отражателей до приемной антенны,

$R_{1n} = \sqrt{(x_n - x_1)^2 + (y_n - y_1)^2}$ – расстояние от приемной антенны до одного из отражателей,

$R_{2n} = \sqrt{(x_n - x_2 - S)^2 + (y_n - y_2 - H)^2}$ – расстояние от передающей антенны до одного из отражателей,

$\lambda = c / f_0$ – длина волны. Доплеровский сдвиг частоты определяется радиальной составляющей скорости движения БПА в направлении на НПУ $f_n = f_{\max} \cos \beta$ – значение доплеровского сдвига для n -го отражателя,

$$\cos \beta = \frac{(x_n - x_2 - S) \cos \theta - (y_n - y_2 - H) \sin \theta}{\sqrt{(x_n - x_2 - S)^2 + (y_n - y_2 - H)^2}},$$

$f_{\max} = Vf_0 / c$ – максимальное значение доплеровского сдвига частот, V – скорость движения маневрирующего объекта, $c = 3 \times 10^8$ м/с – скорость распространения радиоволн.

Сдвиги фазы θ_n , вносимые каждым отражателем, являются независимыми одинаково распределенными случайными величинами с равномерным законом распределения на интервале $[0, 2\pi)$, причем при расчете всех канальных коэффициентов значения фазы считаются постоянными. Затухание амплитуды сигнала при прохождении от передатчика к отражателю, от отражателя к приемнику, а также при отражении в при расчете нормированной корреляционной функции несущественны.

При оценивании канальных коэффициентов и пропускной способности ММО системы передачи информации важное значение имеет характер изменения канальных коэффициентов во времени. В наиболее общем виде эти изменения задаются корреляционной функцией каждого из коэффициентов, отражающей скорость изменения во времени и особенности изменений модуля и аргумента комплексных канальных коэффициентов:

$$R_{ijmn}(\tau) = \frac{\mathbf{M}\{\underline{H}_{ij}(t) \underline{H}_{mn}^*(t + \tau)\}}{\sqrt{\mathbf{M}\{|\underline{H}_{ij}(t)|^2\} \mathbf{M}\{|\underline{H}_{mn}(t + \tau)|^2\}}}. \quad (7)$$

Корреляционные функции определяются различным образом в зависимости от особенностей формирования канальных коэффициентов.

Экспериментальные исследования. Для оценки корреляционной функции канальных коэффициентов ММО системы связи $R_{ijmn}(\tau)$ проведено компьютерное моделирование. Условия проведения моделирования: радиус кольца отражателей $R = 50$ м, интервал дискре-

тизации во времени $\Delta t = 1$ мкс, количество отражателей $N = 60$, центральная частота передатчика $f_0 = 2,4$ ГГц, начальные расстояния от передающих до приемных антенн по осям $S = 1000$ м, $H = 100$ м, начальная скорость движения объекта $V = 50$ м/с. Усреднение при вычислении корреляционной функции (7) происходит по $N_{st} = 1000$ реализациям.

Рассчитаны корреляционные функции канальных коэффициентов при различных моделях движения объекта. Если движение объекта имеет детерминированный характер, например, описывается моделью Дубинса или более простыми полиномиальными моделями, а фазы отраженных от элементарных отражателей можно считать постоянными на заданном временном интервале, то характер изменения канальных коэффициентов также детерминированный. Источниками случайных возмущений канальных коэффициентов являются начальные фазы элементарных отражателей θ_n , которые изменяют свое значение при каждом эксперименте. Данная модель справедлива, если перемещение объекта за время τ незначительно, а ракурс визирования элементарных отражателей изменяется настолько незначительно, что не приводит к существенному изменению коэффициента рассеяния.

На рис. 2 представлен модуль корреляционной функции канального коэффициента при прямолинейном равномерном движении объекта. В данном случае корреляция канального коэффициента сохраняется на уровне 90% при $\tau = 7$ мс.

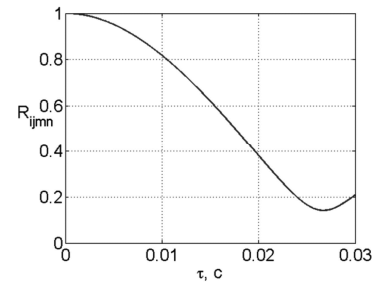


Рис. 2. Зависимость усредненного значения корреляции от времени движения объекта при прямолинейном равномерном движении

На рис. 3 приведен модуль корреляционной функции канального коэффициента при движении объекта по траектории Дубинса. По сравнению со случаем равномерного прямолинейного движения, показанным на рис. 2, при движении по траектории Дубинса корреляция на уровне 90% сохраняется на значительно меньшем временном интервале $\tau = 0,2$ мс.

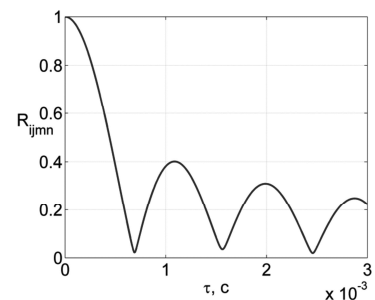


Рис. 3. Корреляционная функция при движении объекта по траектории Дубинса

Если движение объекта имеет детерминированный характер, но перемещение объекта за время τ достаточно велико, то можно считать фазы отраженных от элементарных отражателей случайными, независимыми при различных значениях τ . На рис. 4 представлен модуль корреляционной функции канального коэффициента при прямолинейном равномерном движении объекта и случайных фазах отражателей. В данном случае, из-за постоянного изменения фаз отражателей элементы МКК становятся некоррелированными между собой, что не позволяет применять экстраполяцию значений коэффициентов в последующие моменты времени.

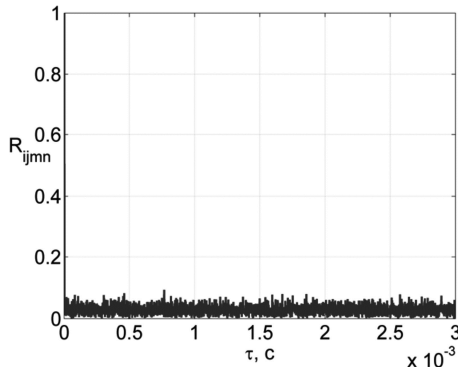


Рис. 4. Корреляционная функция

при прямолинейном равномерном движении объекта и случайных фазах отражателей

Если движение объекта имеет случайный характер, задаваемый возмущениями траектории в соответствии с моделями 1 и 2, то это необходимо учитывать при вычислении корреляционной функции методом статистического моделирования. В этом случае фазы отраженных от элементарных отражателей сигналов также могут быть постоянными при разных задержках во времени или независимыми при различных значениях задержки. На рис. 5 представлена модуль корреляционной функции канального коэффициента при прямолинейном движении объекта с флюктуацией скорости.

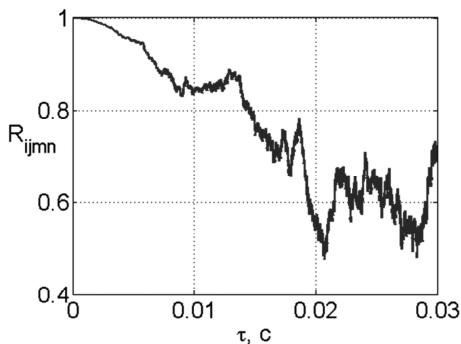


Рис. 5. Корреляционная функция

при прямолинейном движении объекта с флюктуацией скорости

На рис. 6 представлена модуль корреляционной функции канального коэффициента при движении объекта по траектории Дубинса с флюктуацией скорости.

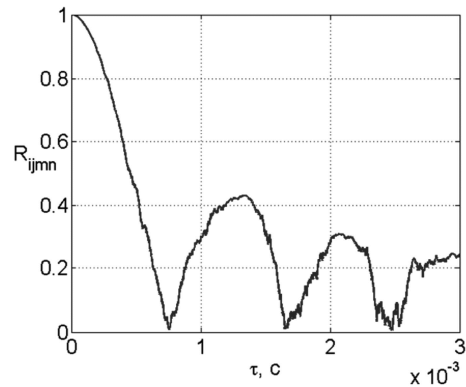


Рис. 6. Корреляционная функция при движении объекта по траектории Дубинса с флюктуацией скорости

По сравнению со случаями, показанными на рис. 2 и рис. 3, при движении объекта с флюктуацией скорости корреляция на уровне 90% сохраняется.

Исследования выполнены при поддержке гранта Российского научного фонда, проект № 14-19-01263 в Рязанском государственном радиотехническом университете.

Литература

1. Ermolayev V.T., Flaksman A.G., Kovalyov I.P., Averin I.M. Weight Error Loss in MIMO Systems Using Eigenchannel Technique // Proceedings of the 1st International Conference on Antenna Theory and Techniques (ICATT'03). Sevastopol, Ukraine. 2003. – P.333–336.
2. Паршин Ю.Н., Кудряшов В.И. Анализ пропускной способности канала передачи информации от беспилотного летательного аппарата при неточной канальной матрице // Вестник Рязанского государственного радиотехнического университета. – 2015. – № 52. – С. 19–24.
3. Dubins L. E. On curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents // American Journal of Mathematics. – 1957. – V. 79. – №. 3 – P. 497–516.
4. Eriksson-Bique, S.D., Kirkpatrick, D.G., Polishchuk, V. Discrete Dubins paths. CoRR, abs/1211.2365 (2012).
5. Кондратьев, В. С. Многопозиционные радиотехнические системы / В. С. Кондратьев, А. Ф. Котов, Л. Н. Марков. – М.: Радио и связь, 1986. – 264 с.
6. Кузьмин, С. З. Цифровая радиолокация. Введение в теорию / С. З. Кузьмин. – Киев. Издательство, 2000. – 428 с.
7. Cook, M. V. Flight dynamics principles. London, Arnold; New York, Wiley, 1997. – 379 p.
8. M. Patzold, B. O. Hogstad. A Space-Time Channel Simulator for MIMO Channels Based on the Geometrical One-Ring Scattering Model // Wireless Communications and Mobile Computing. Special Issue on Multiple-Input Multiple-Output (MIMO) Communications. – Nov. 2004. – V4. – №. 7. – P. 727–737.

ИССЛЕДОВАНИЕ ВЛИЯНИЯ СПЕКТРА СВЯЗАННОСТИ ЦИКЛОВ В ГРАФЕ ТАННЕРА НА ЭНЕРГЕТИЧЕСКИЙ ВЫИГРЫШ КОДИРОВАНИЯ ИЗВЕСТНЫХ LDPC КОДОВ

Овинников А.А., н.с. кафедры телекоммуникаций и основ радиотехники Рязанского государственного радиотехнического университета, e-mail: ovinnikov.a.a@tor.rsreu.ru.

ACE SPECTRUM INFLUENCE ON CHANNEL CODING GAIN OF KNOWN LDPC CODES

Ovinnikov A.A.

The construction of finite-length irregular LDPC codes with low error floors is currently an attractive research problem. In particular, for the binary erasure channel (BEC), the problem is to find the elements of selected irregular LDPC code ensembles with the size of their minimum stopping set being maximized. Due to the lack of analytical solutions to this problem, a simple but powerful heuristic design algorithm, the approximate cycle extrinsic message degree (ACE) constrained design algorithm, has recently been proposed. Building upon the ACE metric associated with a cycle in a code graph, we use the ACE spectrum of LDPC codes as a useful tool for evaluation of codes from selected irregular LDPC code ensembles. We justify the ACE spectrum approach through examples and simulation results.

Key words: LDPC, approximate cycle extrinsic, heuristic design algorithm, girth.

Ключевые слова: помехоустойчивое кодирование, итеративное декодирование, низкоплотностные коды, граф Таннера, обхват графа, связанность циклов графа, энергетический выигрыш от кодирования (ЭВК).

Введение

В связи с постоянным увеличением скорости передаваемой информации и, как следствие, ростом ёмкостей для её хранения повышаются требования к информационной достоверности. Увеличение этого показателя возможно при внедрении эффективных алгоритмов помехоустойчивого кодирования и декодирования, среди которых следует особенно отметить наиболее перспективные с точки зрения практической реализации турбо и LDPC коды. Широкое распространение в стандартах беспроводной связи [1] и передачи данных получил подкласс низкоплотностных кодов – двоичные квазициклические коды с малой плотностью проверок на чётность (QC-LDPC). Их основное преимущество заключается в возможности компактного хранения проверочных матриц, возможности реализации быстрых алгоритмов кодирования и параллельного вероятностного декодирования. Однако, как и для любых других итеративно-декодируемых кодов, открытым остаётся вопрос повышения энергетического выигрыша от кодирования в области малых шумов. Один из способов решения этой проблемы лежит в плоскости исследования свойств циклов и их взаимосвязей в графах Таннера низкоплотностных кодов. Этому вопросу и посвящена настоящая работа.

Теоретические аспекты

Информация, передаваемая от одних узлов графа Таннера к другим в процессе вероятностного декодирования, неизбежно попадает в замкнутые структуры с

Рассматривается исследовательская задача оценки ряда параметров, присущих графам Таннера нерегулярных кодов с низкой плотностью проверок на чётность, для определения степени их влияния на энергетический выигрыш от кодирования. Эффективность кодов оценивается с помощью имитационного моделирования по методу Монте Карло в канале с белым шумом и итеративным вероятностным декодером, работающим по алгоритму «распространение доверия».

наиболее простой конфигурацией в виде циклов. Известные по ряду работ [2-5] более сложные образования – множества «остановок» и «ловушек» фактически состоят из конечного числа связанных друг другом циклов различной длины. Для каналов со стираниями энергетический выигрыш от кодирования полностью определяется размером наименьшего множества «остановок» [4]. Для других каналов связи такая зависимость не столь очевидна и до настоящего времени однозначно не построена полная антология всех подграфов в структуре графа Таннера, которые негативно влияют на эффективность итеративных декодеров. С другой стороны интуитивно понятно, что уменьшение количества объектов, относящихся к множествам «ловушек» и «остановок», должно приводить к увеличению достоверности информации при переходе от одной итерации декодирования к другой. Однако, конфигурация рассматриваемых структур в графе Таннера, влияющих на ЭВК может значительно отличаться для двух разных кодов. В силу этого большой интерес представляет поиск и верификация критерия оценки меры внутренней связи циклов в графе Таннера без необходимости определения всех имеющихся конфигураций множеств «ловушек» и «остановок» с оценкой степени их влияния.

Рассмотрим несколько различных множеств символьных вершин в графе Таннера, содержащих в своём составе циклы. Пусть C_d – множество, состоящее из d

символьных вершин, таких, что оно содержит в своём составе минимум один цикл. Максимальная длина цикла в C_d при этом ограничена и равна $2d$. Если в дополнении к вышесказанному добавить условие, что каждая из вершин в составе C_d связана с остальными как минимум дважды, то из C_d получается множество «остановки» S_d . Пусть цикл длиной $g \leq 2d$ содержит в своём составе вершины из C_d , тогда связанность g с остальными вершинами определяется как:

$$\eta = \sum_i (d_i - 2), \quad i = 1, 2, \dots, g/2, \quad (1)$$

где d_i – вес i -й вершины графа Таннера. Метрика связанности η цикла с графом показывает общее количество ветвей, по которым может транслироваться информация, не подверженная влиянию цикла. Для регулярных LDPC кодов параметр η является постоянным для групп циклов одной длины и равен:

$$\eta_r = \frac{g}{2} (d_s - 2). \quad (2)$$

Таким образом, оценка влияния метрики связанности на эффективность декодирования регулярных кодов затруднительна, т.к. она является постоянной величиной для заданного графа Таннера. В тоже время использование параметра η может оказаться продуктивным при определении энергетической характеристики нерегулярных кодов. Проанализируем причины появления эффекта насыщения вероятности ошибки декодирования для заданного графа Таннера. Для этого воспользуемся распределением минимального значения метрик связанности – спектром $S(\eta)$, для циклов длиной от обхвата $g_{min}=g_0$ до некоторого максимального значения g_{max} . Таким образом, исследование ЭВК при вероятностном декодировании будет заключаться в оценке спектра метрик связанности и определении его корреляции с наличием эффекта насыщения, появление которого фиксируется по результатам моделирования в канале с АБГШ по методу Монте Карло. Для нахождения значения η требуется идентифицировать все циклы, имеющиеся в проверочной матрице вплоть до определённой длины. Поэтому первоначально требуется выбрать наиболее подходящий алгоритм поиска и нумерации замкнутых структур в двудольных разреженных графах.

Алгоритм идентификации коротких циклов в двудольных графах

Известно, что реализация алгоритмов подсчёта циклов в произвольных графах представляет значительную сложность [6]. Однако, разреженность структуры проверочной матрицы позволяет снизить вычислительные затраты традиционных алгоритмов [7] подсчёта циклов. В силу того, что наибольшее влияние на ЭВК оказывают короткие циклы, то целесообразно использовать алгоритмы, способные за ограниченное время оценивать длину циклов с длиной вплоть до $2g_0$. В литературе представлен ряд алгоритмов, способных оценивать длину цикла [8], однако большинство из них либо имеют ограничение на максимальную длину цикла (g_{max}) с привязкой к размеру обхвата, причём $g_{max} < 2g_0$ либо требу-

ют значительных затрат памяти, которые как минимум линейно растут с длиной кода. С учётом рассмотренных выше ограничений для оценки метрик связанности используется алгоритм «цепей» для идентификации всех циклов в графе Таннера вплоть до $2g_0$ включительно.

Суть алгоритма цепей заключается в последовательном построении дерева цепей из каждой символьной вершины V на глубину k , наиболее актуально для $k \leq 2g_0$, с последующим поиском всех циклов, проходящих через анализируемую вершину. Путём поиска в ширину из заданной вершины V строится дерево цепей высоты $k/2 - 1$, так что вершины дерева, образующие уровень l (нумерация – с нуля), соответствуют всевозможным цепям длиной l с началом V в исходном графе. Далее находятся все пары листьев построенного дерева, для каждой из которых соответствующие цепи образуют цикл при соединении их концов с какой-либо общей смежной вершиной V . Необходимо проверить, чтобы обе цепи не «пересекались», а также не проходили через вершину V . В [8] предписывается перебирать только пары листьев из разных поддеревьев с корнями, составляющими первый уровень дерева. В противном случае цепи заведомо «пересекаются».

Дерево цепей высоты $k/2 - 1$ позволяет найти не только циклы длиной k , но и более короткие циклы. Для поиска циклов длиной $l < k$ можно воспользоваться уровнем $l=2$ (вместо $l/2 - 1$) и рассматривать пары вершин этого уровня, соответствующих цепям с общим концом. С целью унификации процедуры поиска циклов различной длины дерево цепей удобно строить до высоты $k/2$.

Подсчёт всех циклов длиной не более k осуществляется путём запуска указанной процедуры для каждой вершины одной доли графа. Здесь также применим общий оптимизационный приём: не рассматривать вершину после того как найдены все циклы, проходящие через неё.

Таким образом, используя алгоритм «цепей» для идентификации циклов длиной вплоть до $2g_0$, можно подготовить исходные данные для алгоритма определения спектра $S(\eta)$ циклов в графе Таннера.

Исследования влияния спектра связанности циклов на ЭВК известных LDPC кодов

Исследование эффективности метрик связанности циклов для быстрой оценки наличия порога проводилось на псевдослучайных кодах [9, 10] и трёх ансамблях из стандартов [11, 12]. Параметры исследуемых кодов представлены табл. 1 во второй колонке. Аббревиатуры кодов, полученных на базе известных алгоритмов синтеза, имеют следующую расшифровку – Мак [9] (псевдослучайные коды Маккая) и PEG [10] (алгоритм последовательно расширения графа с максимизацией минимального локального цикла). К кодам из стандартов относятся: RAVIS – нерегулярные LDPC коды наименьшей длины из спецификации [12], Wi1 – коды, используемые на физическом уровне стандарта беспроводного широкополосного доступа WiMAX [11] и, наконец, Wi2 – помехоустойчивые коды из спецификации для беспроводных локальных вычислительных сетей 802.11n/ac [11]. В качестве алгоритма обнаружения и исправления ошибок использовался итеративный вероятностный декодер

ИВР с мягким входом и выходом [1] и максимальным количеством итераций равным 20. Сигнал на выходе кодера модулировался низкочастотной двоичной фазовой модуляцией и поступал в канал с двоичным входом и непрерывным выходом с АБГШ. При расчёте вероятностей ошибки на бит для получения достоверных оценок накапливалось не менее 1000 ошибок для построения каждой точки. Графические зависимости вероятность битовой ошибки от отношения сигнал шум представлены на рис. 1-5. Для определения эффективности ис-

пользования спектра метрик для каждого в отдельности взятого кода с помощью алгоритма «цепей» были идентифицированы циклы с длинами от g_0 до g_0+6 и составлена соответствующая табл. 2. На основании этих данных был выполнен расчёт метрик связности по каждой пронумерованной замкнутой структуре графа. В работе определено количество циклов ($n(\cdot)$) с длинами вплоть до $2g_0+2$. Также были определены минимальные значения метрик связности $S(\eta_i)$ для всех идентифицированных циклов.

Таблица 1. Параметры исследуемых LDPC кодов

N	Сокращение	R	g_0	$\lambda(x)$
1066	Mak	1/2	6	$0,206 \cdot x^7 + 0,293 \cdot x^2 + 0,5 \cdot x + 0,001$
	PEG			
	RAVIS			
1312	Mak	1/2	6	$0,206 \cdot x^7 + 0,29 \cdot x^2 + 0,5023 \cdot x + 0,0007$
	PEG			
	RAVIS			
1056	Mak	1/2	8	$0,2083 \cdot x^5 + 0,3333 \cdot x^2 + 0,4583 \cdot x$
	PEG		6	
	Wi1		6	
1248	Mak	1/2	6	$0,2083 \cdot x^5 + 0,3333 \cdot x^2 + 0,4583 \cdot x$
	PEG		8	
	Wi1		6	
1344	Mak	1/2	6	$0,2083 \cdot x^5 + 0,3333 \cdot x^2 + 0,4583 \cdot x$
	PEG		8	
	Wi1		6	
648	Mak	1/2	4	$0,125 \cdot x^{11} + 0,4167 \cdot x^2 + 0,4583 \cdot x$
		2/3		$0,125 \cdot x^7 + 0,0417 \cdot x^5 + 0,2083 \cdot x^3 + 0,3333 \cdot x^2 + 0,2917 \cdot x$
		3/4		$0,2083 \cdot x^5 + 0,25 \cdot x^3 + 0,3333 \cdot x^2 + 0,2083 \cdot x$
		5/6		$0,7917 \cdot x^3 + 0,0833 \cdot x^2 + 0,125 \cdot x$
	PEG	1/2	6	$0,125 \cdot x^{11} + 0,4167 \cdot x^2 + 0,4583 \cdot x$
		2/3		$0,125 \cdot x^7 + 0,0417 \cdot x^5 + 0,2083 \cdot x^3 + 0,3333 \cdot x^2 + 0,2917 \cdot x$
		3/4		$0,2083 \cdot x^5 + 0,25 \cdot x^3 + 0,3333 \cdot x^2 + 0,2083 \cdot x$
		5/6		$0,7917 \cdot x^3 + 0,0833 \cdot x^2 + 0,125 \cdot x$
	Wi2	1/2	4	$0,125 \cdot x^{11} + 0,4167 \cdot x^2 + 0,4583 \cdot x$
		2/3		$0,125 \cdot x^7 + 0,0417 \cdot x^5 + 0,2083 \cdot x^3 + 0,3333 \cdot x^2 + 0,2917 \cdot x$
		3/4		$0,2083 \cdot x^5 + 0,25 \cdot x^3 + 0,3333 \cdot x^2 + 0,2083 \cdot x$
		5/6		$0,7917 \cdot x^3 + 0,0833 \cdot x^2 + 0,125 \cdot x$
1296	Mak	1/2	6	$0,125 \cdot x^{10} + 0,0417 \cdot x^3 + 0,375 \cdot x^2 + 0,4583 \cdot x$
		2/3		$0,125 \cdot x^7 + 0,0833 \cdot x^6 + 0,5 \cdot x^2 + 0,2917 \cdot x$
		3/4		$0,2917 \cdot x^3 + 0,5 \cdot x^2 + 0,2083 \cdot x$
		5/6		$0,6667 \cdot x^3 + 0,2083 \cdot x^2 + 0,125 \cdot x$
	PEG	1/2	6	$0,125 \cdot x^{10} + 0,0417 \cdot x^3 + 0,375 \cdot x^2 + 0,4583 \cdot x$
		2/3		$0,125 \cdot x^7 + 0,0833 \cdot x^6 + 0,5 \cdot x^2 + 0,2917 \cdot x$
		3/4		$0,2917 \cdot x^3 + 0,5 \cdot x^2 + 0,2083 \cdot x$
		5/6		$0,6667 \cdot x^3 + 0,2083 \cdot x^2 + 0,125 \cdot x$
	Wi2	1/2	6	$0,125 \cdot x^{10} + 0,0417 \cdot x^3 + 0,375 \cdot x^2 + 0,4583 \cdot x$
		2/3	4	$0,125 \cdot x^7 + 0,0833 \cdot x^6 + 0,5 \cdot x^2 + 0,2917 \cdot x$
		3/4	6	$0,2917 \cdot x^3 + 0,5 \cdot x^2 + 0,2083 \cdot x$
		5/6	6	$0,6667 \cdot x^3 + 0,2083 \cdot x^2 + 0,125 \cdot x$
1944	Mak	1/2	6	$0,125 \cdot x^{10} + 0,0417 \cdot x^3 + 0,375 \cdot x^2 + 0,4583 \cdot x$
		2/3		$0,1667 \cdot x^7 + 0,0417 \cdot x^6 + 0,5 \cdot x^2 + 0,2917 \cdot x$
		3/4		$0,25 \cdot x^3 + 0,5417 \cdot x^2 + 0,2083 \cdot x$
		5/6		$0,4167 \cdot x^3 + 0,4583 \cdot x^2 + 0,125 \cdot x$
	PEG	1/2	6	$0,125 \cdot x^{10} + 0,0417 \cdot x^3 + 0,375 \cdot x^2 + 0,4583 \cdot x$
		2/3		$0,1667 \cdot x^7 + 0,0417 \cdot x^6 + 0,5 \cdot x^2 + 0,2917 \cdot x$
		3/4		$0,25 \cdot x^3 + 0,5417 \cdot x^2 + 0,2083 \cdot x$
		5/6		$0,4167 \cdot x^3 + 0,4583 \cdot x^2 + 0,125 \cdot x$
	Wi2	1/2	6	$0,125 \cdot x^{10} + 0,0417 \cdot x^3 + 0,375 \cdot x^2 + 0,4583 \cdot x$
		2/3	4	$0,1667 \cdot x^7 + 0,0417 \cdot x^6 + 0,5 \cdot x^2 + 0,2917 \cdot x$
		3/4	6	$0,25 \cdot x^3 + 0,5417 \cdot x^2 + 0,2083 \cdot x$
		5/6	6	$0,4167 \cdot x^3 + 0,4583 \cdot x^2 + 0,125 \cdot x$

Таблица 2. Оценка количества циклов и их метрик связанности в исследуемых графах Таннера

N	Сокр.	R	$n(g_0)$	$n(g_0+2)$	$n(g_0+4)$	$n(g_0+6)$	$S(\eta_0)$	$S(\eta_1)$	$S(\eta_2)$	$S(\eta_3)$
1066	Mak	1/2	2577	45712	895095	18151786	0	0	0	0
	PEG		383	38787	656038	12377406	7	6	6	2
	RAVIS		3047	58188	1208187	25906904	1	1	1	1
1312	Mak		2486	45268	881218	17839027	0	0	0	0
	PEG		96	37908	643199	12079373	7	6	6	6
	RAVIS		2972	55693	1143747	24338489	1	1	1	1
1056	Mak		663	7099	91191	1168366	1	0	1	0
	PEG		5042	74607	868060	10795584	1	1	1	0
	Wi1		440	6886	75758	967230	8	5	5	2
1248	Mak		631	7436	90667	1171857	1	0	0	0
	PEG		4016	75351	861152	10624572	1	1	1	1
	Wi1		312	7644	75244	974818	8	5	5	3
1344	Mak	649	7347	92005	1186190	1	1	0	0	
	PEG	3523	76346	866175	10650354	1	0	1	1	
	Wi1	280	7476	75712	974568	8	5	5	2	
648	Mak	1/2	86	8661	227711	6517545	2	0	0	0
		2/3	28	10306	279512	8508927	3	0	0	1
		3/4	203	21525	778058	30183786	2	1	0	0
		5/6	2522	180478	12708504	-	1	0	1	-
	PEG	1/2	4038	130027	2971601	76809500	1	1	1	1
		2/3	7259	200451	5511750	-	1	1	1	-
		3/4	13780	430972	14561200	-	1	1	1	-
		5/6	36105	1509866	69638364	-	2	1	2	-
	Wi2	1/2	3942	123012	2760507	70568721	11	11	10	2
		2/3	8046	193023	5288679	-	3	2	2	-
		3/4	54	13662	433431	14510070	3	3	2	3
		5/6	32346	1574775	68153562	-	2	1	3	-
1296	Mak	1/2	4787	105988	2551778	-	0	1	0	-
		2/3	10390	295511	9223157	-	1	0	1	-
		3/4	17274	588316	21731924	-	0	1	0	-
		5/6	6	31339	1297670	58085184	3	0	1	1
	PEG	1/2	22	82419	1528348	34502257	27	0	1	1
		2/3	4253	256305	7025399	-	1	1	1	-
		3/4	13609	510308	17851513	-	1	1	1	-
		5/6	26607	1096676	47002182	-	1	1	1	-
	Wi2	1/2	2754	68229	1470906	33283602	10	10	3	2
		2/3	108	7830	237627	6885540	11	6	6	6
		3/4	14040	512433	17523054	-	3	2	3	-
		5/6	22842	1141830	46268496	-	2	1	3	-
1944	Mak	1/2	4613	101626	2426214	60052447	1	1	1	1
		2/3	10621	297956	9305797	-	0	0	0	-
		3/4	12472	384127	12824133	-	0	1	0	-
		5/6	16711	573719	21080924	-	0	1	0	-
	PEG	1/2	77959	1518877	34302934	-	1	1	1	-
		2/3	160	273336	7193269	-	6	1	1	-
		3/4	4124	379411	11427397	-	1	1	1	-
		5/6	12644	549913	19465562	-	1	0	1	-
	Wi2	1/2	3321	70551	1465371	33089445	10	10	4	3
		2/3	81	6399	251667	7072515	12	7	5	5
		3/4	9558	346923	11269044	-	3	2	2	-
		5/6	14418	563193	19125801	-	3	1	2	-

В соответствии с результатами, представленными в табл. 2, можно обратить внимание на то, что коды, обозначенные аббревиатурами Wi1 и Wi2, обладают относительно высоким средним уровнем спектра $S(\eta_i)$. Кроме этого на общем фоне выделяются некоторые коды, синтезированные по алгоритму PEG, обладающие аналогичной особенностью. Объяснение относительно высокого среднего уровня спектральных составляющих только для 2-х кодов PEG из всего ансамбля заключено

в самом алгоритме синтеза, согласно которому циклы короткой длины будут появляться по мере увеличения общего количества вершин. Поэтому если начинать построение графа с вершин с меньшим весом, то короткие циклы будут по большей части содержаться в наименее разреженной области проверочной матрицы. Это в свою очередь способствует увеличению среднего значения спектра $S(\eta_i)$.

Рассмотрим кривые энергетической эффективности

исследуемых кодов с низкой плотностью проверок на чётность в канале с АБГШ и сопоставим графические и табличные результаты. На каждом из графиков представлены зависимости вероятности битовой ошибки (p_b) от отношения сигнал-шум для 3-х категорий кодов, обозначенных соответствующими аббревиатурами, значения которых описаны ранее. Целью сопоставления результатов декодирования и анализа кодовых конструкций является выявление корреляционных связей между значениями метрик связанности и уровнем порога насыщения битовой ошибки на кривых энергетической эффективности.

Анализируя полученные зависимости можно сделать ряд умозаключений, а именно:

1. Низкий уровень порога для рассмотренных ансамблей нерегулярных кодов наблюдается в том случае, когда минимальные значения метрик η_0, η_1, η_2 оказываются менее трёх для скорости кодирования $R = 0.5$. При этом играет роль именно совокупное значение метрик для циклов разной длины. В частности для кода PEG с параметрами $N = 1296, g_0 = 6, R = 0.5$ минимальное значение метрики связанности для цикла длиной с обхват графа равно 27, что является максимальной величиной для всех возможных кодов. Однако, это не приводит к существенному снижению эффекта дна согласно рис. 2, т.к. остальные отсчёты спектра $S(\eta_i), i \neq 0$ принимают значения нуля либо единицы.

2. Увеличение обхвата графа не приводит к снижению уровня порога насыщения вероятности битовой ошибки, что характерно для кодов PEG с длинами $N=1056, 1296$ и 1344 .

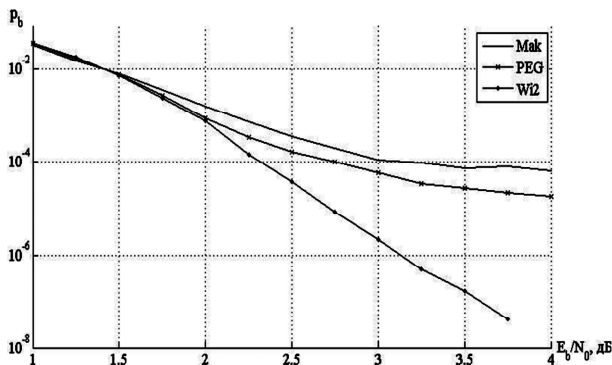


Рис. 1. Эффективность декодирования коротких ($N = 648$ бит) LDPC кодов с различными способами синтеза проверочных матриц при $R = 1/2$

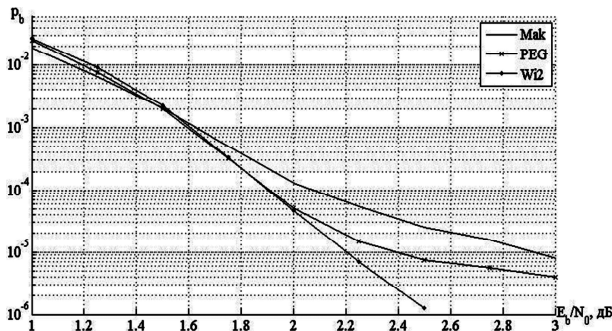


Рис. 2. Эффективность декодирования ($N = 1296$ бит) LDPC кодов с различными способами синтеза проверочных матриц при $R = 1/2$

3. Увеличение скорости спада характеристик $p_b(E_b/N_0)$ в области относительно малых шумов достигается из-за двух факторов: увеличения значений метрик связанности для циклов разной длины, а также уменьшения общего количества коротких циклов в графе Таннера.

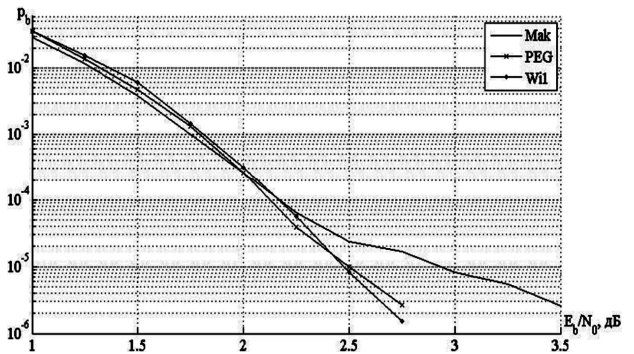


Рис. 3. Эффективность декодирования ($N = 1056$ бит) LDPC кодов с различными способами синтеза проверочных матриц при $R = 1/2$

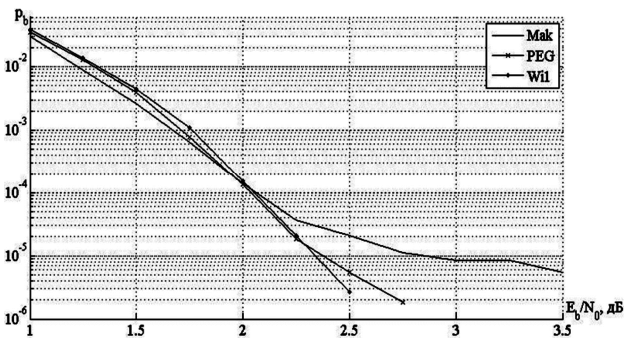


Рис. 4. Эффективность декодирования ($N = 1248$ бит) LDPC кодов с различными способами синтеза проверочных матриц при $R = 1/2$

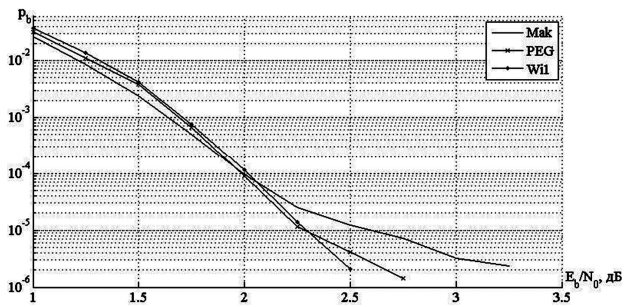


Рис. 5. Эффективность декодирования ($N = 1344$ бит) LDPC кодов с различными способами синтеза проверочных матриц при $R = 1/2$

Заключение

В работе проведено исследование эффективности декодирования нерегулярных квазициклических LDPC кодов, полученных из современных стандартов беспроводной передачи данных либо синтезированных на основе известных алгоритмов. Оценка энергетического выигрыша от кодирования в области малых шумов привязывалась к критерию минимизации спектра метрик связанности графа Таннера. Также был ограничен обхват графа на уровне 6 либо 8. Полученные результаты свидетельствуют о целесообразности использования предлагаемого подхода и требуют дополнительных под-

тверждений для кодов различной скорости и весовых распределений, что является одним из направлений дальнейших исследований.

Исследование выполнено за счет гранта Российского научного фонда (проект 14-19-01263) в Рязанском государственном радиотехническом университете.

Литература

1. Ryan W.E. and S. Lin. «Channel Codes. Classical and Modern», Cambridge University Press, 2009.
2. MacKay D. and Postol M.S., «Weaknesses of Margulis and Ramanujan-Margulis low-density parity-check codes», *Electronic Notes Theoretical Computer Science*, vol. 74, 2003.
3. Richardson T.J., «Error floors of LDPC codes», in Proc. 41th Allerton Conf. Commun., Computing Control, Oct. 2003.
4. Di C., Proietti D., Telatar E., Richardson T., and Urbanke R., «Finitelength analysis of low-density parity-check codes on the binary erasure channel», *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1570–1579, Jun. 2002.
5. Vasic B., Chilappagari S., Nguyen D., and Planjery S., «Trapping set ontology», Proc. 47th Annual Allerton Conf. on Commun., Control and Computing, Monticello, IL, Sept. 2009, pp. 1-7.
6. Harary F., Manvel B. On the Number of Cycles in a Graph. *Matematický časopis*, 1971, vol. 21, no. 1, pp. 55-63.
7. Halford T.R., Chugg K.M. An Algorithm for Counting Short Cycles in Bipartite Graphs. *IEEE Transactions on Information Theory*, 2006, vol. 52, no. 1, pp. 287–292.
8. Воропаев А.Н., Учёт обхвата при подсчёте коротких циклов в двудольных графах, *Информационные процессы*, Том 11, № 2, 2011, стр. 225–252.
9. MacKay D., «Good error-correcting codes based on very sparse matrices», *IEEE Trans. Inform. Theory*, vol. 45, pp. 399-431, Mar. 1999.
10. Hu X.-Y., Eleftheriou E., and Arnold D.-M., «Progressive edge-growth Tanner graphs», in Proc. IEEE GlobeCom, Nov. 2001, vol. 2, pp. 995-1001.
11. Declercq D., Fossorier M., Biglieri E., *Channel Coding. Theory, Algorithms, and Applications*. Academic Press Library in Mobile and Wireless Communications, 2014.
12. ГОСТ Р 54309-2011. «Аудиовизуальная система реального времени (РАВИС). Процессы формирования кадровой структуры, канального кодирования и модуляции для системы цифрового наземного узкополосного радиовещания в ОВЧ диапазоне. Технические условия».

БАНКИ ФИЛЬТРОВ И OFDM В СИСТЕМАХ ШИРОКОПОЛОСНОЙ ПЕРЕДАЧИ ДАННЫХ СО МНОГИМИ НЕСУЩИМИ

Витязев В.В., заведующий кафедрой телекоммуникаций и основ радиотехники Рязанского государственного радиотехнического университета, д.т.н., профессор, e-mail: vityazev.v.v@rsreu.ru;

Никушкин П.Б., магистрант Рязанского государственного радиотехнического университета.

FILTER BANKS AND OFDM IN BROADBAND DATA TRANSMISSION MULTICARRIER SYSTEMS

Vityazev V.V., Nikishkin P.B.

The methods of constructing broadband data transmission multicarrier system are considered. The way of adaptive broadband data transmission based on a combined approach is proposed. This way combines filter banks technology and OFDM. The spectral and energy efficiency system is shown in conditions of frequency fading, noise reduction and Doppler shifts.

Key words: filter banks, OFDM, data transmission, broadband, combine approach, spectral efficiency, adaptation, noise reduction.

Ключевые слова: банки фильтров, OFDM, передача данных, широкополосная, комбинированный способ, спектральная эффективность, адаптация, борьба с помехами.

Введение

Исторически первой областью применения методов анализа/синтеза сигналов на основе банка (набора) цифровых полосовых фильтров (ЦПФ) в системах телекоммуникаций является построение трансмультиплексоров – устройств преобразования вида уплотнения каналов: ВРК в ЧРК (временного в частотное разделение, на передающей стороне) и ЧРК в ВРК (частотного во временное разделение, на приемной стороне) [1, 2]. Последующее развитие теории многоскоростной обработки сигналов применительно к реализации систем анализа/синтеза сигналов на основе банков фильтров [3-6] нашло широкое отражение в работах, посвященных кодированию речи, звука и изображения.

Новым этапом на пути развития теории и применения банков ЦПФ стали широкополосные системы передачи данных со многими несущими, как альтернатива в ряде приложений технологии OFDM [7-9].

Известны [8] неоспоримые преимущества OFDM в системах высокоскоростной беспроводной передачи данных:

- значительное увеличение длительности символов (по отношению к методам модуляции с одной несущей), позволяющее вести эффективную борьбу с межсимвольной интерференцией (МСИ);
- простая генерация передаваемого сигнала с использованием обратного БПФ и не менее простое разделение принимаемого сигнала с применением прямого БПФ;
- высокая спектральная эффективность, как результат адаптивной модуляции на поднесущих;

Рассматриваются методы построения системы широкополосной передачи данных со многими несущими. Предлагается способ адаптивной широкополосной передачи данных на основе комбинированного подхода, сочетающий технологии банка фильтров и OFDM. Показана возможность повышения спектральной и вычислительной эффективности системы высокоскоростной передачи данных на основе предложенного способа в условиях частотных замираний, подавления сосредоточенными по спектру помехами и доплеровских смещений.

– существенное упрощение реализации адаптивного эквалайзера;

– эффективное сочетание технологий OFDM и MIMO.

Вместе с тем, как отмечено в работе [8], в некоторых областях, таких как когнитивное радио, системы беспроводной передачи с многопользовательским доступом на линии «мобильный абонент – базовая станция», технология OFDM трудно реализуема из-за проблемы синхронизации.

Повышенный интерес к использованию банков фильтров для систем передачи данных со многими несущими – FBMC (Filter Bank Multicarrier), был связан с поиском путей компенсации потерь спектральной эффективности и обеспечения работоспособности системы передачи данных со многими несущими в условиях частотно-временного рассеивания в многолучевых каналах связи, подверженных замираниям, доплеровскому смещению и расширению частот. Борьба с межсимвольной (МСИ) и межканальной интерференцией (МКИ) в условиях частотно-временного рассеивания – основная проблема на пути дальнейшего повышения эффективности широкополосной передачи данных со многими несущими.

Известен целый ряд подходов к решению этой проблемы как в рамках технологии OFDM, так и с применением банков фильтров, технологии FBMC [10]. В системах связи, использующих OFDM, для борьбы с МСИ применяют циклический префикс (защитный интервал), что приводит к потере спектральной эффективности. Но

главная проблема технологии OFDM – это МКИ, обусловленная частотным рассеиванием, высокая чувствительность к доплеровским смещениям и расширениям, неминуемо приводящим к потере ортогональности частотных каналов и, как следствие, срывам синхронизации [8, 10]. Введение защитных интервалов в виде «нулевых» поднесущих и спектральных окон не решает проблему в достаточной степени и приводит к потере спектральной эффективности. Одним из возможных походов к решению проблемы уменьшения МКИ и МСИ является использование хорошо локализованных в частотно-временной области сигнальных базисов, в частности базиса Вейля Гейзенберга [11].

Вместе с тем, неконтролируемое доплеровское смещение частот и в этом случае приводит к срывам синхронизации.

Таким образом, проблема повышения спектральной эффективности и работоспособности систем широкополосной передачи данных в многолучевых дисперсионных каналах связи с использованием технологии OFDM и банков фильтров по-прежнему остается актуальной. В настоящей работе предпринимается попытка объединения двух технологий в рамках двухступенчатой и, в более общем случае, многоступенчатой реализации. Предполагается, что весь диапазон рабочих частот широкополосного канала связи предварительно разбивается на отдельные поддиапазоны и внутри каждого поддиапазона частот формируется OFDM сигнал с последующим объединением всех частотных составляющих в общий групповой OFDM с ЧПК. Рассматриваются достоинства и недостатки данного подхода с позиции спектральной эффективности, работоспособности в

условиях частотных замираний и доплеровских искажений, а также вычислительных затрат.

Системы широкополосной передачи данных со многими несущими на основе банков фильтров и OFDM

На рис. 1 представлена классическая схема синтезатора группового широкополосного сигнала на основе банка цифровых полосовых фильтров-интерполяторов, реализующая одновременную передачу данных по N частотным каналам на несущих частотах ω_i , $i = \overline{1, N}$, где N – приведенная круговая частота. Предполагается, что предварительно высокоскоростной поток данных $x(nT_1)$ преобразуется в N параллельных последовательностей с понижением частоты дискретизации в N раз. Частота дискретизации каждой последовательности увеличивается экспандером в K раз ($K \geq N$) путем расстановки $K-1$ нулей между каждой парой соседних отсчетов. Низкочастотный фильтр с частотной характеристикой $H_0(\omega)$ интерполирует промежуточные отсчеты до частоты дискретизации группового сигнала, равной $2\pi K / N$, а умножитель $e^{j\omega_i m}$, $i = \overline{1, N}$, на выходе каждого низкочастотного фильтра-интерполятора трансформирует спектр выходного сигнала в окрестность i -ой несущей частоты ω_i . Параллельно-последовательный преобразователь формирует групповой широкополосный сигнал, суммируя выходы всех частотных каналов.

На приемной стороне реализуется дуальный процесс – преобразование принятого группового сигнала с ЧПК в N последовательностей с ВПК и понижением частоты дискретизации на выходе каждого канала в K раз. Па-

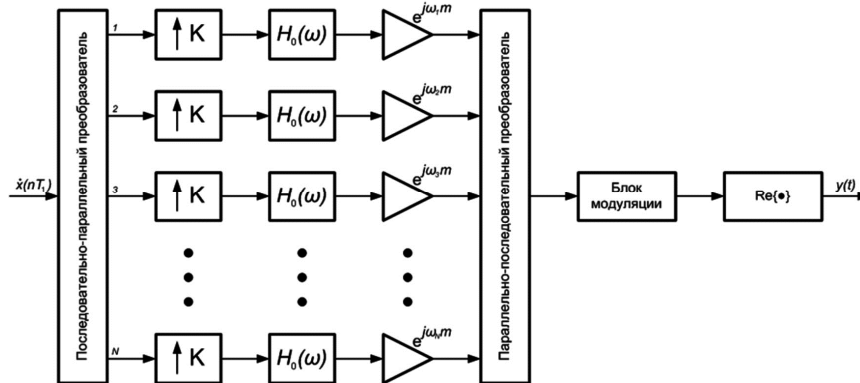


Рис. 1. Структурная схема синтезатора группового широкополосного сигнала на основе банка цифровых фильтров

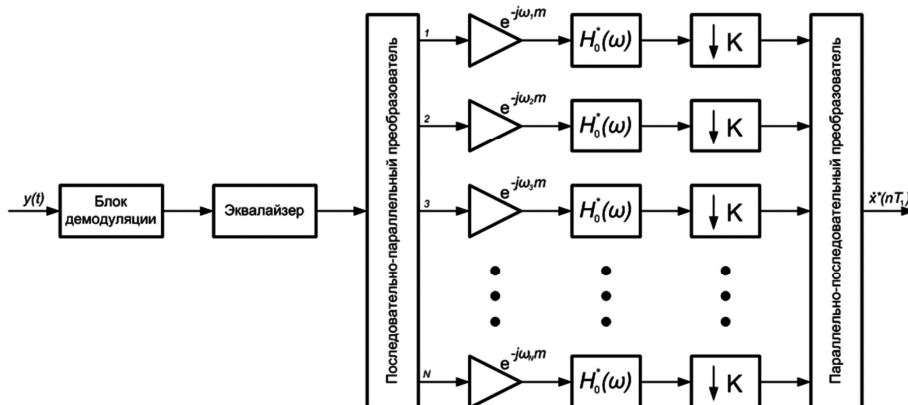


Рис. 2. Структурная схема анализатора группового широкополосного сигнала на основе банка цифровых фильтров

раллельно-последовательный преобразователь формирует высокоскоростной поток данных $x^*(nT_1)$, коммутируя выходы всех частотных каналов в один общий поток. Для борьбы с линейными амплитудно-частотными искажениями, вносимыми каналом связи, в общую цепь преобразования на приемной стороне включается адаптивный эквалайзер.

Основным элементом данной структуры является полосовой фильтр-интерполятор (на передающей стороне) и полосовой фильтр-дециматор (на приемной стороне). От частотной избирательности (формы спектрального окна), определяемой порядком фильтра интерполятора (дециматора), зависит уровень МКИ и устойчивость работы в условиях доплеровского смещения несущих частот. Увеличение порядка КИХ-фильтров приводит к значительному увеличению вычислительных затрат и памяти данных.

Однако рассматриваемый способ построения набора фильтров-интерполяторов (дециматоров) по прямой параллельной форме реализации, когда каждый фильтр реализуется независимо от других фильтров, является далеко не лучшим решением проблемы с позиции минимизации приведенных вычислительных затрат. Опираясь на теорию многоскоростной обработки сигналов, были разработаны многочисленные методы и алгоритмы построения банков цифровых фильтров частотной селекции [3-5], сравнительный анализ эффективности которых применительно к задаче построения системы передачи со многими несущими был рассмотрен в [10].

Вместе с тем, в условиях отсутствия заметных доплеровских искажений и глубоких замираний в канале связи, наилучшее решение проблемы высокоскоростной передачи данных со многими несущими с позиции спек-

тральной эффективности и минимума вычислительных затрат, как известно, дает OFDM, которую можно рассматривать как частный случай построения банка фильтров с прямоугольным окном импульсной характеристики длительностью NT_1 ($K = N$) и частотной характеристикой вида

$$\frac{\sin(N\omega/2)}{N \sin(\omega/2)}$$

При этом, если число частотных каналов N кратно степени двойки, то многократное уменьшение вычислительных затрат дает применение алгоритма ОБПФ на передающей стороне и алгоритма БПФ на приемной стороне (рис. 3).

Несмотря на простоту реализации и широкий спектр применения технологии OFDM, отмеченные выше недостатки этого популярного вида модуляции, требует проведения дальнейших исследований эффективности работы систем высокоскоростной передачи данных на основе банков фильтров и OFDM.

В первую очередь, это касается работы в условиях глубоких замираний, защиты от мощных сосредоточенных по спектру помех и доплеровских искажений. Одним из возможных подходов к решению поставленной задачи является сочетание достоинств и возможностей банков фильтров и OFDM.

Способ адаптивной широкополосной передачи данных со многими несущими

Предлагается комбинированный подход, предполагающий разбиение всего диапазона рабочих частот широкополосного группового сигнала на M поддиапазонов и реализацию внутри каждого поддиапазона более коротких OFDM сигналов ($OFDM_i, i = 1, M$).

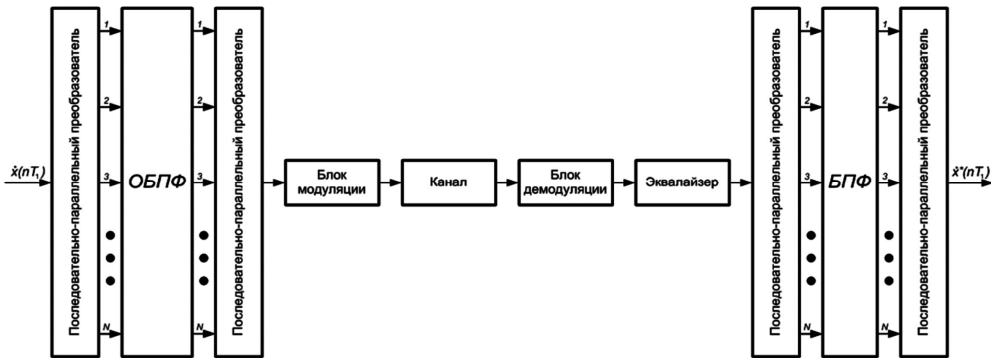


Рис. 3. Общая схема построения системы передачи данных с применением OFDM

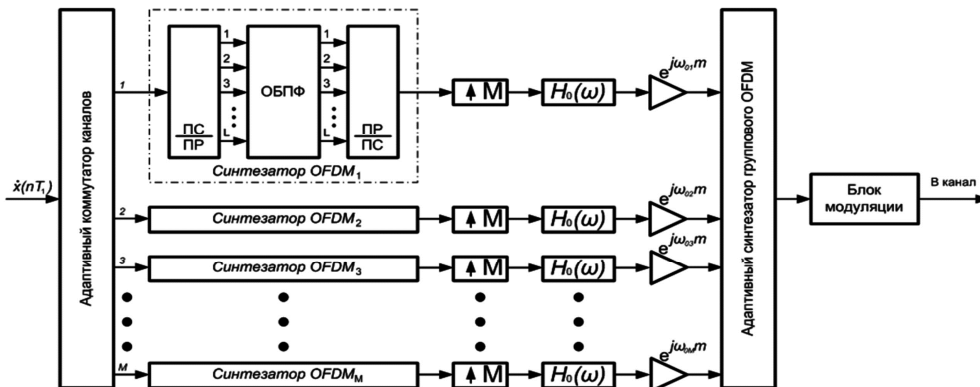


Рис. 4. Структурная схема синтезатора группового широкополосного OFDM сигнала

На рис. 4 представлена структурная схема синтезатора группового широкополосного OFDM сигнала. Высокоскоростной поток данных $x(nT_1)$ распределяется адаптивным коммутатором каналов на M частотных поддиапазонов с понижением частоты дискретизации (скорости потоков) в M раз. В каждом частотном поддиапазоне формируется i -ый OFDM сигнал, $i = \overline{1, M}$ по классической структуре с применением L -точечного ОБПФ. Далее частота дискретизации каждого $OFDM_i$ повышается экспандером частоты дискретизации в M раз, низкочастотный фильтр с функцией передачи (частотной характеристикой) $H_0(\omega)$ интерполирует промежуточные данные, а умножитель на комплексную функцию $e^{j\omega_0 m}$, $i = \overline{1, M}$ «транслирует» сигнал $OFDM_i$ в окрестность центральной частоты ω_{0i} (как по классической схеме трансмультиплексора). Адаптивный синтезатор группового OFDM формирует широкополосный групповой OFDM сигнал, суммируя выходы всех задействованных частотных каналов.

На приемной стороне реализуется дуальный процесс – преобразование принятого группового сигнала OFDM с ЧРК в M временных последовательностей с понижением частоты дискретизации в M раз на выходе каждого частотного канала. Эквалайзер, работающий на пониженной в M раз частоте дискретизации выравнивает ту часть частотной характеристики канала связи, которая соответствует i -му поддиапазону частот. Анализатор OFDM по схеме, дуальной синтезатору OFDM, формирует последовательность выходных данных по каждому частотному каналу. Адаптивный коммутатор каналов формирует общий высокоскоростной поток $x^*(nT_1)$, совпадающий с передаваемым потоком $x(nT_1)$ в пределах заданной точности.

Идея построения адаптивной системы заключается в том, что по результатам предварительного тестирования канала связи и определения зон затухания и подавления, можно управлять скоростью передачи данных, уменьшая или увеличивая ее в зависимости от числа пораженных частотных каналов. Предполагается, что в процесс формирования группового широкополосного OFDM сигнала

включаются только те частотные каналы, которые обеспечат устойчивую приемо-передачу i -го OFDM сигнала. Необходимо также отметить, что в рамках данной структуры проще обеспечить ортогональность более коротких OFDM сигналов, а адаптивный эквалайзер, имеющий существенно меньший порядок и работающий на пониженной частоте дискретизации, более эффективно решает задачу выравнивания частотных характеристик в каждом отдельном поддиапазоне частот.

Синтез оптимальной структуры системы широкополосной передачи данных со многими несущими по критерию минимума вычислительных затрат при заданном значении спектральной эффективности прежде всего ставит вопрос о минимизации затрат на реализацию банка цифровых фильтров. При увеличении требований к спектральной эффективности растет число частотных каналов, позволяя более точно локализовать зоны глубоких замираний и подавления сосредоточенными по спектру помехами. С увеличением числа частотных каналов вычислительные затраты могут расти по квадратичной зависимости, так как пропорционально числу каналов растет порядок КИХ-фильтров, используемых для интерполяции и децимации в структуре синтезатора и анализатора группового широкополосного OFDM сигнала. В этой связи целесообразно проведение сравнительного анализа вычислительной эффективности трех основных подходов к построению банка полосовых фильтров [4]: пирамидальной многоступенчатой структуры, полифазной формы с применением ДПФ, прямой параллельной формы на основе алгоритма быстрой свертки, применительно к системе широкополосной передачи со многими несущими. Нельзя не учитывать и влияние собственных шумов децимации, обусловленных «элайзингом» («заворачиванием» боковых лепестков спектра выходного сигнала в область частот OFDM сигнала) [12]. Поэтому альтернативный подход к построению банка фильтров может базироваться на применение цифровых гребенчатых фильтров [4]. В работе [13] предложен метод построения системы сбора и передачи данных с ЧРК на основе пирамидальной структуры цифровых гребенчатых фильтров, отличающийся высокой вычислительной эффективностью и отсутствием шума децимации.

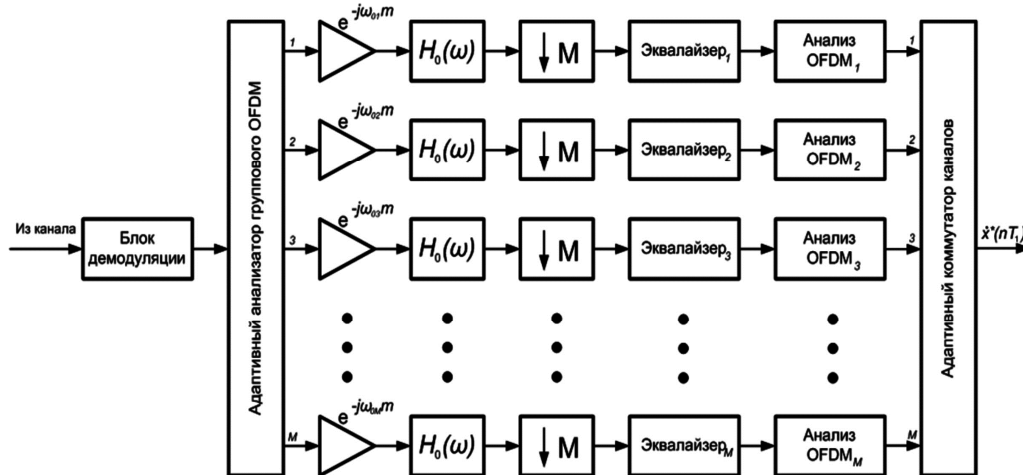
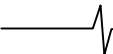


Рис. 5. Структурная схема анализатора группового широкополосного OFDM сигнала



Заключение

Отмечено, что в ряде приложений применение технологии FBMC (банков фильтров со многими несущими) и реализация системы широкополосной передачи данных со многими несущими на ее основе дает лучшее решение с позиции спектральной и энергетической эффективности.

Предложен способ адаптивной широкополосной передачи данных со многими несущими на основе комбинированного подхода, сочетающего технологии банка фильтров и OFDM. Банк из M полосовых фильтров берет на себя задачу разбиения всего диапазона рабочих частот на M поддиапазонов с последующим формированием OFDM сигнала внутри каждого частотного канала. Групповой широкополосный OFDM сигнал формируется адаптивным синтезатором на передающей стороне и разделяется на M частотных каналов с последующим формированием высокоскоростного потока данных адаптивным коммутатором каналов.

Показаны возможности предложенного способа по повышению спектральной эффективности в условиях частотных замираний и доплеровских смещений, а также подавления сосредоточенными по спектру помехами.

Работа выполнена при поддержке Российского научного фонда в Рязанском государственном радиотехническом университете (проект №14-19-01263).

Литература

1. Витязев В.В. Цифровая обработка сигналов: ретроспектива и современное состояние // Электросвязь. – 1997. – № 6.

2. Шойерманн Х., Геклер Х. Систематизированный обзор цифровых методов преобразования вида уплотнения каналов // ТИИЭР. 1981. Т. 69. № 11. С. 52–84.

3. Crochiere R.E., Rabiner L. Multirate Digital Signal Processing. Prentice Hall. Englewood Cliffs. – NJ, 1983.

4. Витязев В.В. Цифровая частотная селекция сигналов. М.: Радио и связь, 1993. 240 с.

5. Vaidyanathan P.P. Multirate Systems and Filter Banks. Prentice Hall. Englewood Cliffs. – NJ, 1993.

6. Mitra S.K. Digital Signal Processing: a computer-based approach. McGraw-Hill. Comp. Inc., 1998.

7. Farhang-Boroujenu B. Signal Processing Techniques for Software Radios// Lulu publishing house, 2010.

8. Behrouz Farhang-Boroujenu. OFDM Versus Filter Bank Multicarrier // IEEE Signal Processing Magazine, -2011, -Vol. 28, № 3, – P. 92-112.

9. Lin L. and Farhang-Boroujenu B. Cosine modulated multitone modulation for very high-speed digital subscriber lines // EURASIP J. Appl. Signal Processing, – 2006, Article ID 19329.

10. Овинников А.А. Методы анализа/синтеза сигналов в системах беспроводной связи со многими несущими // Электросвязь. 2013. № 9. – с. 28-32.

11. Волчков В.П. Новые технологии передачи и обработки информации на основе хорошо локализованных сигнальных базисов // Научные ведомости. 2009. № 15. – с. 181-189.

12. Витязев В.В., Горюшкин Р.С. Анализ шумов квантования многоскоростных структур узкополосных КИХ-фильтров // Цифровая обработка сигналов. 2015. № 4.

13. Витязев В.В., Никишкин П.Б. // Метод анализа/синтеза сигналов в системах передачи данных с частотным уплотнением каналов // Электросвязь. 2014. № 12. – с. 4-9.

УДК 621.372.542

АНАЛИЗ ШУМОВ КВАНТОВАНИЯ МНОГОСКОРОСТНЫХ СТРУКТУР УЗКОПОЛОСНЫХ КИХ-ФИЛЬТРОВ

Витязев В.В., д.т.н., зав. кафедрой телекоммуникаций и основ радиотехники Рязанского государственного радиотехнического университета (РГРТУ), e-mail: vityazev.v.v@rsreu.ru;

Горюшкин Р.С., м.н.с., аспирант кафедры телекоммуникаций и основ радиотехники РГРТУ.

ANALYSIS OF QUANTIZATION NOISES IN MULTIRATE STRUCTURES OF NARROWBAND FIR-FILTERS

Vityazev V.V., Gorushkin, R.S.

This article is devoted to the analysis problem of quantization noises in multirate structures of narrowband filters with finite impulse response (FIR-filters). The goal of this research is to minimize influence of such noises increasing the number of transformation stages and decreasing orders of series connected decimation and interpolation filters. The results of modeling in MATLAB+Simulink are also considered.

Keywords: multirate signal processing, narrowband FIR-filter, quantization noise, decimation, interpolation, modeling.

Ключевые слова: многоскоростная обработка сигналов, узкополосный КИХ-фильтр, шумы квантования, децимация, интерполяция, моделирование.

Введение

Многоскоростная обработка сигналов предполагает, что в процессе линейного преобразования цифрового сигнала возможно изменение частоты дискретизации (шага квантования по времени) в сторону уменьшения или увеличения и, как следствие, требуемой скорости обработки. Это приводит к более эффективной обработке сигнала, в первую очередь для систем узкополосной КИХ-фильтрации, так как открывается возможность значительного уменьшения требуемой вычислительной производительности проектируемой цифровой системы.

Многоскоростная фильтрация и особенности её применения стали предметом исследований многочисленных научных работ по цифровой обработке сигналов [1-9]. Внимание авторов было сосредоточено главным образом на проблеме минимизации приведенных вычислительных затрат, памяти данных и коэффициентов. Вместе с тем вопросы, связанные с анализом шумов квантования, прежде всего шума децимации, обусловленного эффектом прореживания отсчетов обрабатываемого сигнала, и последующей интерполяции (если требуется восстановление исходной частоты дискретизации), поднимались в сравнительно небольшом числе работ [10, 11] и не дали простого и эффективного решения проблемы уменьшения их влияния.

Как известно [6, 7], шумы квантования цифровых фильтров разделяются на две составляющие: шумы, обусловленные округлением при выполнении операций умножения и масштабирования, и шумы, источником которых является округление коэффициентов фильтра. Для многоскоростных систем дополнительным и основным источником шума становятся шумы децимации и интерполяции [3-5, 11].

Ставится и решается задача анализа шумов квантования многоскоростных структур узкополосных фильтров с конечной импульсной характеристикой (КИХ-фильтров) с целью последующей минимизации их влияния путем увеличения числа ступеней преобразования и уменьшения порядков последовательно соединяемых фильтров-дециматоров и фильтров-интерполяторов. Приводятся результаты моделирования в среде MATLAB+Simulink.

При относительно малоразрядном представлении коэффициентов фильтра не удастся достигнуть желаемой точности аппроксимации частотных характеристик в зоне непрозрачности (полосе непропускания). В этом случае можно воспользоваться последовательным соединением двух-трех КИХ-фильтров, каждый из которых обеспечивает «грубую» аппроксимацию желаемой частотной характеристики, достаточную для заданного уровня подавления боковых лепестков в зоне непрозрачности (в пределах 20 дБ – 60 дБ). Однако повышение уровня боковых лепестков приводит к значительному увеличению шума децимации, если каждый последовательно соединяемый КИХ-фильтр реализуется с использованием децимации и интерполяции преобразуемого сигнала [11].

В том, что касается шумов округления при умножении и масштабировании, то при реализации на современных цифровых сигнальных процессорах, он практически лежит в пределах младшего сохраняемого разряда (при передаче данных между последовательно соединенными фильтрами), поскольку внутри каждого отдельного КИХ-фильтра операция умножения с накоплением выполняется без округления промежуточных результатов. Таким образом, анализ шумов квантования многоскоростных систем КИХ-фильтрации сводится к анализу влияния шумов децимации и интерполяции.

В настоящей работе ставится и решается задача анализа шумов квантования многоскоростных структур узкополосных КИХ-фильтров с целью последующей минимизации их влияния путем увеличения числа ступеней

преобразования и уменьшения порядков последовательно соединяемых фильтров-дециматоров и фильтров-интерполяторов.

Анализ шумов квантования одноступенчатой структуры КИХ-фильтра

На рис. 1 представлена шумовая модель одноступенчатой структуры КИХ-фильтра, реализуемого на основе децимации и интерполяции с использованием дополнительного формирующего фильтра для обеспечения повышенной частотной избирательности в зоне его непрозрачности.

Предполагается, что на вход фильтра поступает сигнал $x(nT_1)$, где T_1 – период дискретизации, представляющий собой дискретный белый шум с нулевым математическим ожиданием и дисперсией σ_S^2 . Входной сигнал подвергается децимации – понижению частоты дискретизации в ν раз. Коэффициент децимации ν выбирается, как правило, по критерию минимизации приведенных вычислительных затрат. Амплитудно-частотная характеристика (АЧХ) фильтра-дециматора (ФД) $H_{\Phi Д}(\omega)$ характеризуется относительно широкой переходной зоной (малым коэффициентом прямоугольности АЧХ), что позволяет многократно уменьшить его порядок. При этом частота среза полосы пропускания ω_{C_1} определяется заданной шириной полосы пропускания проектируемого фильтра, а частота среза зоны непрозрачности ω_{C_2} – выбранным значением коэффици-

$$\text{ента децимации } \nu : \omega_{C_2} \leq \frac{2\pi}{\nu} - \omega_{C_1}.$$

Вследствие понижения частоты дискретизации на выходе фильтра-дециматора в ν раз, все частотные составляющие спектра выходного сигнала, расположенные выше частоты π/ν , «сворачиваются» в область низких частот от 0 до π/ν (эффект «элайзинга» [3-5]), что в представленной шумовой модели учитывается введением источника шума децимации $e_D(mT_2)$. Статистические свойства источника шума $e_D(mT_2)$, при заданной модели входного сигнала $x(nT_1)$, определяются в основном частотной избирательностью фильтра-дециматора (его импульсной $h_D(nT_1)$ и частотной $H_{\Phi Д}(\omega)$ характеристиками), но и зависят от степени перекрытия основной (низкочастотной) и боковых сос-

тавляющих спектра прореженного сигнала $x_D(mT_2)$.

При выполнении условия $\nu \leq \frac{\pi}{\omega_{C_2}}$ эту зависимость, с

позиции формы корреляционной и спектральной характеристик, можно не учитывать, так как степень перекрытия определяется только уровнем боковых лепестков в зоне непрозрачности АЧХ фильтра-дециматора. Вместе с тем, эффект «сворачивания» большого числа боковых лепестков спектра сигнала на его выходе в низкочастотную область полезного сигнала приводит к заметному повышению дисперсии шума децимации $e_D(mT_2)$.

Шум децимации $e_D(mT_2)$ включает в себя две составляющие:

$$e_D(mT_2) = e_{DS}(mT_2) + e_{DP}(mT_2).$$

Первая составляющая $e_{DS}(mT_2)$ определяет ту часть спектральной плотности мощности шума децимации, которая попадает в область частот полезного сигнала, в полосе от 0 до ω_{C_1} , вторая составляющая $e_{DP}(mT_2)$ – все, что лежит за полосой пропускания, т.е. в переходной зоне (если $\nu \geq \pi/\omega_{C_2}$) и зоне непрозрачности фильтра-дециматора.

Оценку дисперсии первой составляющей (в диапазоне частот от 0 до ω_{C_1}) можно представить в виде

$$\begin{aligned} \sigma_{DS}^2 &= \frac{\sigma_S^2}{\pi} \sum_{k=1}^{\nu-1} \int_0^{\omega_{C_1}} \left| H_{\Phi Д}(j\omega + j\frac{2\pi k}{\nu}) \right|^2 d\omega \leq \\ &\leq \frac{\nu-1}{\beta} \varepsilon_{2\text{донФД}}^2 \sigma_S^2, \end{aligned} \quad (1)$$

где $\beta = 2\pi/\omega_{C_1}$ – показатель узкополосности ФД,

$\varepsilon_{2\text{донФД}}$ – допустимый уровень боковых лепестков ФД (для равноволновой аппроксимации по Чебышеву).

Дисперсия второй составляющей (в диапазоне частот от ω_{C_1} до $(2\pi/\nu) - \omega_{C_1}$) определяется формой АЧХ фильтра-дециматора в переходной зоне и уровнем боковых лепестков в зоне его непрозрачности:

$$\begin{aligned} \sigma_{DP}^2 &= \frac{\sigma_S^2}{\pi} \sum_{k=1}^{\nu-1} \int_{\omega_{C_1}}^{\frac{\pi}{\nu} - \omega_{C_1}} \left| H_{\Phi Д}(j\omega + j\frac{2\pi k}{\nu}) \right|^2 d\omega \leq \\ &\leq \frac{\beta - 2(\nu-1)}{2\beta} \varepsilon_{2\text{донФД}}^2 \sigma_S^2, \end{aligned} \quad (2)$$

если $\nu \leq \pi/\omega_{C_2}$ и

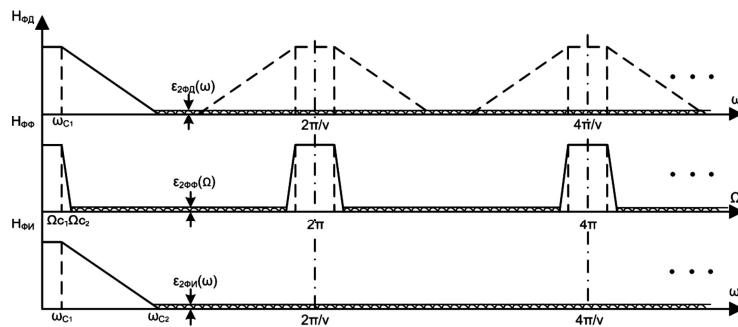


Рис. 1. Шумовая модель одноступенчатой структуры КИХ-фильтра

$$\sigma_{DP}^2 = \frac{\sigma_S^2}{\pi} \int_{\omega_{C1}}^{\pi - \omega_{C1}} \left| H_{\Phi D} \left(j\omega + j \frac{2\pi}{v} \right) \right|^2 d\omega + \frac{\sigma_S^2}{\pi} \sum_{k=2}^{v-1} \int_{\omega_{C1}}^{\pi - \omega_{C1}} \left| H_{\Phi D} \left(j\omega + j \frac{2\pi k}{v} \right) \right|^2 d\omega, \quad (3)$$

если $v > \pi / \omega_{C2}$.

Первая составляющая в (3) определяется степенью перекрытия основной и первой боковой составляющими, а вторая – влиянием остальных боковых полос. При этом первая составляющая, как правило, многократно превышает степень влияния второй составляющей.

Если далее (с целью формирования желаемой частотной характеристики и подавления шума децимации в переходной зоне) используется формирующий фильтр (ФФ) с передаточной функцией $H_{\Phi\Phi}(j\Omega)$, подавляющий всё, что лежит за полосой пропускания $\Omega_{C1} \cong \Omega_{C2}$ (рис. 1), то шум децимации на его выходе будет фактически определяться только первой составляющей $e_{DS}(mT_2)$ в форме «окрашенного» шума с дисперсией

$$\sigma_D^2 \approx \sigma_{DS}^2 \approx \frac{v}{\beta} \varepsilon_{2\text{донФД}}^2 \sigma_S^2.$$

На завершающем этапе преобразований входного сигнала $x(nT_1)$ в выходной сигнал $y(nT_1)$ может использоваться фильтр-интерполятор (ФИ) с передаточной функцией $H_{\Phi И}(j\omega)$, восстанавливающий исходную частоту дискретизации и форму сигнала с заданной точностью. Фильтр-интерполятор практически без искажений пропускает низкочастотную составляющую полезного сигнала и шум децимации в полосе пропускания. Но одновременно вносит ошибку интерполяции, связанную с недостаточным уровнем подавления боковых составляющих спектра сигнала $y_D(mT_2)$ на его входе. Приняв за основу статистическую модель шума интерполяции $e_{И}(nT_1)$, можно оценить его дисперсию в следующей форме [11]:

$$\sigma_{И}^2 = \frac{\sigma_S^2 v}{\pi} \sum_{k=1}^{v-1} \int_0^{\omega_{C1}} \left| H_{\Phi И} \left(j\omega + j \frac{2\pi k}{v} \right) \right|^2 d\omega \leq \frac{v(v-1)}{\beta} \varepsilon_{2\text{донФИ}}^2 \sigma_S^2.$$

Предполагается, что для согласования уровней сигналов (спектральной плотности мощности) на входе и выходе ФИ сигнал $y(nT_1)$ усиливается в \sqrt{v} раз. Таким образом, дисперсия шума интерполяции увеличивается пропорционально квадрату коэффициента децимации v^2 , зависит от уровня боковых лепестков АЧХ в зоне непрозрачности $\varepsilon_{2\text{донФИ}}$ и показателя узкополосности β . Для однотипных характеристик ФИ и ФД дисперсия шума интерполяции в v раз больше дисперсии шума децимации. Вместе с тем, их влияние на последующую обработку и сам полезный сигнал принципиально различно. Шум децимации «поражает» полезный сигнал, который не подлежит точному восстановлению, в то время как шум интерполяции может сглаживаться на последующих этапах обработки. Кроме того, требования к точности интерполяции, как правило, значительно меньше по отношению к шуму децимации. Поэтому в дальнейшем наше внимание будет сосредоточено только на анализе шумов децимации и уменьшении их влияния в многоскоростных структурах ФД.

Анализ шумов квантования двухступенчатой структуры фильтра-дециматора

На рис. 2 представлена шумовая модель двухступенчатой структуры ФД с использованием дополнительного формирующего фильтра для обеспечения повышенной частотной избирательности в переходной зоне и зоне непрозрачности ФД. Предполагается, что на вход первого фильтра-дециматора ФД1 поступает сигнал $x(nT_1)$, представляющий собой, как и ранее, дискретный белый шум с нулевым математическим ожиданием и дисперсией σ_S^2 . Входной сигнал подвергается двухступенчатой децимации с понижением частоты дискретизации в v_1 раз, на первой ступени, и в v_2 раз на второй ступени преобразований. В результате преобразований частота дискретизации сигнала $x_D(mT_3)$ на выходе второго фильтра-дециматора ФД2 уменьшается в $v = v_1 v_2$ раз. Появление шумов децимации, обусловленных понижением частоты дискретизации, учитывается введением в шумовую модель (рис. 2) источников шума: $e_{D1}(mT_2)$ – на выходе первого ФД и $e_{D2}(lT_3)$ – на выходе второго ФД.

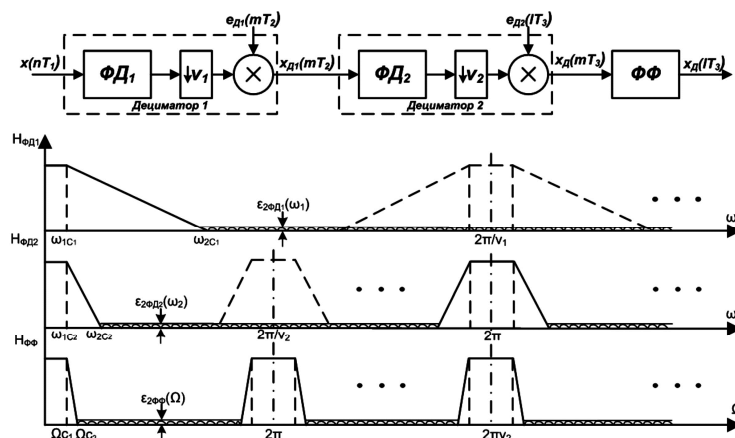


Рис. 2. Шумовая модель двухступенчатой структуры фильтра-дециматора

Формирующий фильтр с передаточной функцией $H_{\Phi\Phi}(j\Omega)$, подавляющий всё, что лежит за полосой его пропускания, как и ранее, обеспечивает не только формирование заданной высокой прямоугольности АЧХ проектируемого фильтра, но и подавляет шум децимации, попадающий в переходную зону и полосу непрозрачности первого и второго ФД. Поэтому в дальнейшем (при условии выполнения принятых ранее ограничений на выбираемые значения коэффициентов децимации ν_1 и ν_2) для оценки дисперсии шума децимации на выходе первого ФД воспользуемся полученной ранее оценкой

$$\sigma_{D_1}^2 \approx \frac{\nu-1}{\beta} \varepsilon_{2\text{донФД}_1}^2 \sigma_S^2. \quad (4)$$

Дисперсия составляющей шума децимации, обусловленной вторым ФД, зависит не только от передаточной функции $H_{\Phi D_2}(j\omega_2)$ самого ФД, но и от передаточной функции $H_{\Phi D_1}(j\omega_1)$ первого ФД, определяющей форму спектральной плотности мощности сигнала на его входе:

$$\begin{aligned} \sigma_{D_2}^2 &= \frac{\sigma_S^2}{2\pi\nu_1} \sum_{k=1}^{\nu_2-1} \int_{-\omega_{2C_1}}^{\omega_{2C_1}} \left| H_{\Phi D_2} \left(j \frac{\omega_2}{\nu_1} + j \frac{2\pi k}{\nu_2} \right) \right|^2 \times \\ &\times \left| H_{\Phi D_1} \left(j\omega_2 + j \frac{2\pi k}{\nu_2} \right) \right|^2 d\omega_2 \approx \\ &\approx \frac{\sigma_S^2}{2\pi\nu_1} \left[\sum_{k=1}^{\nu_2-1} \int_{-\omega_{2C_1}}^{\omega_{2C_1}} \left| H_{\Phi D_1} \left(j \frac{\omega_2}{\nu_1} + j \frac{2\pi k}{\nu_2} \right) \right|^2 d\omega_2 \right] \varepsilon_{2\text{донФД}_2}^2 = \\ &= \frac{\sigma_S^2}{2\pi} \left[\sum_{k=1}^{\nu_2-1} \int_{-\omega_{1C_1}}^{\omega_{1C_1}} \left| H_{\Phi D_1} \left(j\omega_1 + j \frac{2\pi k}{\nu_1\nu_2} \right) \right|^2 d\omega_1 \right] \varepsilon_{2\text{донФД}_2}^2. \end{aligned} \quad (5)$$

Заметим, что если выполняется условие $\nu_1 \leq \pi / \omega_{1C_2}$, то сумму интегралов в (5) можно аппроксимировать выражением вида:

$$\begin{aligned} \sum_{k=1}^{\nu_2-1} \int_{-\omega_{1C_1}}^{\omega_{1C_1}} \left| H_{\Phi D_1} \left(j\omega_1 + j \frac{2\pi k}{\nu_1\nu_2} \right) \right|^2 d\omega_1 \approx \\ \approx 2\omega_{1C_1} \sum_{k=1}^{\nu_2-1} \left| H_{\Phi D_1} \left(j \frac{2\pi k}{\nu_1\nu_2} \right) \right|^2. \end{aligned}$$

Пусть $\frac{2}{\nu_2} \sum_{k=1}^{\nu_2-1} \left| H_{\Phi D_1} \left(j \frac{2\pi k}{\nu_1\nu_2} \right) \right|^2 = \delta_{\Phi D_1}^2$ – оценка среднего квадратичного значения передаточной функции $H_{\Phi D_1}(j\omega_1)$ первого фильтра-дециматора в переходной зоне.

Тогда

$$\sigma_{D_2}^2 \approx \frac{\nu_2 \delta_{\Phi D_1}^2}{\beta} \varepsilon_{2\text{донФД}_2}^2 \sigma_S^2.$$

Поскольку, как правило, $\delta_{\Phi D_1}^2 \ll 1$, можно сделать вывод, что $\sigma_{D_2}^2 \ll \sigma_{D_1}^2$ (при соизмеримых значениях коэффициентов децимации ν_1 и ν_2), а дисперсия сум-

марного шума децимации уменьшается по отношению к одноступенчатой реализации пропорционально отношению $\nu / \nu_1 = \nu_2$.

Очевидно, что дальнейшее увеличение числа ступеней преобразования позволит уменьшить коэффициент децимации ФД первой ступени ν_1 и, как следствие, составляющей дисперсии шума децимации $\sigma_{D_1}^2$ на выходе многоступенчатой реализации. А учитывая, что дисперсия составляющих шума децимации на выходе системы всех последующих ступеней меньше по отношению к первой ступени (как показал пример двухступенчатой реализации), можно сделать общий вывод о возможности уменьшения влияния шума децимации путем увеличения числа ступеней преобразования. Этот вывод объясняется тем фактом, что дисперсия шума децимации зависит не только от коэффициентов децимации, т.е. числа наложений боковых составляющих АЧХ фильтра-дециматора в полосе пропускания, но и от мощности сигнала на его входе, уровень которого уменьшается от ступени к ступени.

Экспериментальные исследования

С целью подтверждения полученных аналитических зависимостей и выводов было проведено моделирование шумовой модели одно- и двухступенчатой структур узкополосного КИХ-фильтра для следующих параметров частотной избирательности. Входной сигнал $x(nT_1)$ – дискретный белый шум с нулевым математическим ожиданием и дисперсией $\sigma_S^2 = 1$. Частота дискретизации входного сигнала – 10 кГц, частота среза полосы пропускания $f_{C_1} = 10$ Гц, частота среза полосы непропускания $f_{C_2} = 13$ Гц. Допустимая неравномерность АЧХ фильтра в полосе пропускания 0,5 дБ, допустимый уровень боковых лепестков в зоне непрозрачности от 20 дБ до 60 дБ.

Как показал расчет параметров КИХ-фильтра, обеспечивающего воспроизведение АЧХ с заданной частотной избирательностью, требуемый порядок достигает значения $N = 3500$ для 20дБ, $N = 5500$, для 40 дБ и $N = 7300$, для 60 дБ. Повышенное значение порядка КИХ-фильтра прежде всего определяется высоким показателем узкополосности $\beta = f_{\text{кв}_1} / f_{C_1} = 10^3$, а также по-

казателем прямоугольности АЧХ $\alpha = \frac{f_{C_1}}{f_{C_2} - f_{C_1}} = 3,33$. В

меньшей степени порядок фильтра зависит от требуемой точности аппроксимации.

С целью уменьшения приведенных вычислительных затрат и памяти данных воспользуемся одноступенчатой реализацией проектируемого узкополосного КИХ-фильтра, представленной на рис. 1. Поскольку цель исследований – определить зависимость уровня (дисперсии) шума децимации от коэффициента децимации ν , будем считать, что значение коэффициента ν задается в пределах от $\nu_{\text{min}} = 2$ до $\nu_{\text{max}} = 64$, с шагом, кратным двойки.

С тем, чтобы исключить влияние самого полезного сиг-

нала в полосе пропускания фильтра (уровень которого, как правило, существенно выше уровня шума децимации) на результаты исследований, сигнал с выхода генератора дискретного белого шума предварительно подается на высокочастотный фильтр с полосой режекции от 0 до 13 Гц. Предполагается, что для формирования АЧХ фильтра с заданным высоким коэффициентом прямоугольности ($\alpha \geq 3$) и подавления шума децимации в переходной зоне фильтра-дециматора используется формирующий фильтр с уровнем подавления боковых лепестков в зоне непрозрачности (более 13 Гц) не менее 80 дБ.

Результаты оценки дисперсии шума децимации $e_D(mT_2)$ (среднеквадратичного значения σ_D) в зависимости от степени затухания (уровня боковых лепестков $\epsilon_{2\text{донФД}}$, в действующих значениях) и коэффициента децимации ν сведены в табл. 1.

Таблица 1.

ν							
$\epsilon_{2\text{донФД}}$ дБ	1	2	4	8	16	32	64
20	99,8	48,3	45,1	41,3	37,7	35,3	32,5
40	114,4	66,2	63,3	60,2	57,5	53,6	50,7
60	116,1	85,8	83,0	79,3	76,2	73,3	70,3

Анализ представленных в табл. 1 результатов моделирования отражает полное совпадение экспериментальных данных и оценок дисперсии шума децимации одноступенчатой структуры, рассчитываемых по формуле (1).

С увеличением коэффициента децимации ν в два раза оценка дисперсии шума децимации увеличивается на 6 дБ (на 3 дБ в действующих значениях), а при увеличении уровня затухания АЧХ в зоне непрозрачности на 20 дБ, примерно на такую же величину уменьшается среднеквадратичное значение шума децимации!

Дальнейшие экспериментальные исследования были связаны с анализом шумов децимации двухступенчатой структуры фильтра-дециматора, представленной на рис. 2, на частных примерах. Предполагалось, что суммарный коэффициент децимации $\nu = \nu_1 \nu_2 = 32$ при различных сочетаниях значений ν_1 и ν_2 . Уровень подавления боковых лепестков в зоне непрозрачности фильтров-дециматоров, как и ранее, менялся от 20 дБ до 60 дБ. Результаты моделирования двухступенчатой структуры фильтра-дециматора представлены в табл. 2.

Таблица 2.

ν			
$\epsilon_{2\text{донФД}}$ дБ	4,8	8,4	16,2
20, σ_D	45,54	40,92	39,41
20, σ_{D_1}	45,45	40,2	38,75
40, σ_D	62,61	60,38	57,0
40, σ_{D_1}	61,63	60,36	56,74
60, σ_D	77,43	75,26	73,32
60, σ_{D_1}	76,48	75,2	73,06

Как показали рассмотренные примеры, уровень

(среднеквадратичное значение σ_D) суммарного шума децимации от первого и второго ФД незначительно (от 0,1 дБ до 1 дБ) отличается от уровня σ_{D_1} шума децимации первого ФД. Поэтому можно сделать общий вывод – среднеквадратичное значение σ_D многоступенчатой структуры фильтра-дециматора не превышает удвоенного значения среднеквадратичной оценки σ_{D_1} шума децимации первого ФД.

Заключение

Проведенные теоретические и экспериментальные исследования влияния шумов децимации в многоскоростных системах узкополосной КИХ-фильтрации показали, что дисперсия шума децимации увеличивается пропорционально коэффициенту децимации ν зависит от уровня боковых лепестков АЧХ в зоне непрозрачности $\epsilon_{2\text{донФД}}$ и показателя узкополосности β .

Для узкополосных систем увеличение числа ступеней преобразования позволит уменьшить коэффициент децимации ФД первой ступени ν_1 и, как следствие, составляющей дисперсии шума децимации $\sigma_{D_1}^2$ на выходе многоступенчатой реализации. А учитывая, что дисперсия составляющих шума децимации на выходе системы всех последующих ступеней меньше по отношению к первой ступени (как показал пример двухступенчатой реализации), можно сделать общий вывод о возможности уменьшения влияния шума децимации путем увеличения числа ступеней преобразования.

Работа выполнена при поддержке Российского научного фонда в Рязанском государственном радиотехническом университете (проект №14-19-01263).

Литература

1. Витязев В.В. Цифровая обработка сигналов: ретроспектива и современное состояние // Электросвязь. – 1997. – № 6.
2. Зубарев Ю.Б., Витязев В.В., Дворкович В.П. Цифровая обработка сигналов – информатика реального времени // Цифровая обработка сигналов. 1999. № 1. С. 5-17.
3. Crochiere R.E., Rabiner L. Multirate Digital Signal Processing. Prentice Hall. Englewood Cliffs. - NJ, 1983.
4. Витязев В.В. Цифровая частотная селекция сигналов. М.: Радио и связь, 1993. 240 с.
5. Vaidyanathan P.P. Multirate Systems and Filter Banks. Prentice Hall. Englewood Cliffs. - NJ, 1993.
6. Mitra S.K. Digital Signal Processing: a computer-based approach. McGraw-Hill. Comp. Inc., 1998.
7. Айфичер Э.С., Джервис Б.У. Цифровая обработка сигналов: практический курс: Пер. с англ. М.: Изд. дом «Вильямс», 2004. 992 с.
8. Витязев В.В. Многоскоростная адаптивная обработка сигналов // Радиотехника. - 2012. - № 3. - С. 17–29.
9. Витязев В.В. Многоскоростная обработка сигналов в системах телекоммуникаций // Электросвязь. – 2013. – № 11. – С. 49- 56.
10. Витязев В.В., Демашов В.С., Степашкин А.И. Шум квантования цифрового фильтра с прореживанием и интерполяцией отсчетов выходного сигнала // Известия вузов – Приборостроение. – 1979. - № 5. – С.3-7.
11. Fred Mintzer, Bede Liu. Aliasing Error in the Design of Multirate Filters // IEEE Trans. on Acoustics, Speech, and Signal Processing, vol. ASSP-26, No. 1, February – 1978, p. 76-88.

НОВОЕ СРЕДСТВО ВЕРИФИКАЦИИ ПЛИС ДЛЯ ЗАДАЧ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ

*Романов А.М., к.т.н., научный сотрудник кафедры проблем управления Института кибернетики МИРЭА,
e-mail: AlexRomashka@yandex.ru;*

*Слащёв Б.В., программист кафедры проблем управления Института кибернетики МИРЭА,
e-mail: slaschov.bogdan@yandex.ru.*

NOVEL FPGA VERIFICATION TOOL FOR DIGITAL SIGNAL PROCESSING APPLICATIONS

Romanov A.M., Slashev B.V.

The article introduces a novel software tool Vmodel toolbox, that allows to increase FPGA verification performance in digital signal processing applications. The Vmodel toolbox speciality is the ability to simulate FPGA as a part of Simulink model without external HDL-simulators. The concept, functionality and key advantages are described. The results of successful implementation in FPGA design projects are described.

Key words: FPGA, verification, digital signal processing, Vmodel toolbox, HDL-simulators.

Ключевые слова: ПЛИС, верификация, цифровая обработка сигналов, Vmodel toolbox, HDL-симулятор.

Введение

На заре применения программируемых логических интегральных схем (ПЛИС) их основным назначением была реализация нестандартных логических преобразований, для которых выпуск отдельных микросхем был экономически невыгоден. После, к этому добавилась потребность в реализации логики для сопряжения нескольких различных интерфейсов между собой [1]. Эти задачи определили основные средства отладки, которые присутствуют в современных пакетах моделирования аппаратно-программного обеспечения (АПО) ПЛИС: временные диаграммы изменения сигналов, временные диаграммы изменения отдельных битов сигнала, возможность просмотра состояния сигналов в бинарном, восьмеричном, десятичном и шестнадцатеричном виде и т.д. Тесты для модулей ПЛИС, как правило, пишутся на одном из HDL-языков, которые идеально подходят для описания дискретных логических преобразований происходящих при обработке того или иного цифрового интерфейса. С развитием технологии ПЛИС и появлением ресурсоемких микросхем основной вектор их применения начал смещаться в сторону сложной цифровой обработки сигналов (ЦОС), однако способы отладки, представленные в современных средствах моделирования, не претерпели существенных изменений. В тоже время потребности разработчика ЦОС на базе ПЛИС в инструментах проектирования и отладки существенно отличаются от аналогичных потребностей разработчика интерфейсной логики. Начать стоит с того, что первый этап проектирования блоков ЦОС проходит в пакетах математического моделирования (MATLAB, SciLab и т.д.). Стандартом на данный момент де-факто является MATLAB. На этом

Предлагается новое программное средство Vmodel toolbox, позволяющее существенно повысить производительность верификации ПЛИС в задачах цифровой обработки сигналов. Особенностью Vmodel toolbox является возможность моделирования ПЛИС как части Simulink модели без использования внешних HDL-симуляторов. Описывается принцип работы, функциональность и основные достоинства нового программного обеспечения. Приводятся примеры удачного внедрения в реальных проектах.

этапе еще до описания на HDL проверяется верность самого математического преобразования, которое будет реализовано на ПЛИС, и соответствие его требуемым характеристикам. Для этого доступно множество инструментов. Так, в пакете Simulink легко можно смоделировать цифровой фильтр произвольной конфигурации, подать на него сигнал с заданными спектральными характеристиками, а потом убедиться в том, что спектр выходного сигнала удовлетворяет требования спецификации. На следующем этапе проектирования происходит реализация синтезированного ранее блока ЦОС на одном из HDL-языков, а для отладки полученного описания приходится использовать HDL-симуляторы, инструментов которых заточен под отладку интерфейсной логики. В результате для получения в процессе верификации входных сигналов с заданными спектральными характеристиками требуется описание процесса их формирования на HDL-языке, что сопоставимо по трудозатратам с созданием описания самого модуля ЦОС. В дополнение к этому такой тестовый HDL-модуль потребует отдельного тестирования. Можно сформировать требуемые сигналы при помощи MATLAB и загрузить их в HDL-симулятор из файла, но такой подход вряд ли можно считать удобным. Аналогично обстоят дела с анализом результатов. В ЦОС практически не востребован побитный анализ отдельных сигналов или представление сигналов в двоичной или восьмеричной форме. В тоже время часто применяется представление сигналов в

форме дробных чисел с фиксированной или плавающей точкой, которая не может быть корректно отражена на временных диаграммах в большинстве средств моделирования. Так же в HDL-симуляторах отсутствуют возможность построения многокоординатных графиков и поверхностей, средства частотного анализа сигналов и т.д. В конечном счете результаты моделирования блока ЦОС приходится сохранять и анализировать в MATLAB или аналогичных пакетах.

Понимая недостаточность имеющегося функционала для успешной отладки модулей ЦОС, разработчики наиболее продвинутых HDL-симуляторов добавили в них специализированный режим со-симуляции с MATLAB. В этом режиме HDL код моделируется в специализированном симуляторе, а остальные элементы системы моделируются в Simulink. Связь двух программ между собой осуществляется по сетевому протоколу TCP/IP непосредственно в процессе моделирования. Такой подход оказался особенно востребован при реализации ЦОС в системах управления, когда для верификации необходимо моделировать функционирование АПО ПЛИС совместно со сложными динамическими аналоговыми объектами, которые описываются дифференциальными уравнениями высокого порядка, а на входы блока ЦОС помимо управляющих сигналов поступают сигналы обратной связи от объекта управления. Один из примеров успешного применения данного подхода при создании нелинейной системы управления двигателем на базе ПЛИС описан в [2]. Главным недостатком со-симуляции является низкая скорость моделирования, связанная с постоянным обменом данными между разными программами. Помимо этого сохраняется, хоть и в меньшей степени, необходимость работы в ходе верификации с двумя различными программами, а также необходимость приобретения отдельных лицензий для каждой из них.

Для того, чтобы исключить использование внешнего HDL-симулятора в компании MathWorks предложили оригинальный подход к аппаратной реализации ЦОС на базе ПЛИС с использованием их продукта HDL Coder, который позволяет автоматически без участия разработчика создавать HDL описание блоков Simulink на языках Verilog и VHDL. При использовании HDL Coder разработчик работает только с моделью, а все этапы её преобразования в файл конфигурации ПЛИС проходят автоматически. Это позволяет в простых проектах обходиться вообще без моделирования HDL кода, ограничиваясь только отладкой функциональной модели в Simulink. Однако в сложных проектах такой подход не применим, так как степень оптимизации кода, созданного при помощи HDL Coder, на сегодняшний день уступает коду профессионального разработчика. И если на стадии прототипирования это не так важно, то в серийных изделиях, где размер АПО ПЛИС напрямую влияет на стоимость и характеристики будущего устройства, HDL-код пока приходится писать вручную. Пример успешного применения HDL Coder для решения задач ЦОС описан в [3].

При разработке сложных устройств на базе ПЛИС активно используются готовые модули (IP-ядра), кото-

рые были созданы ранее в ходе других проектов или куплены у сторонних разработчиков. Это позволяет существенно снизить трудозатраты и сократить сроки выхода на рынок. Для использования стороннего (не созданного автоматически) HDL кода, в HDL Coder есть механизм подключения внешних IP-ядер в формате «черного ящика». Суть его заключается в том, что разработчик самостоятельно создает функциональную модель IP-ядра на Simulink, которой ставит в соответствие имеющийся у него HDL код. В ходе верификации используется именно функциональная модель, а в момент генерации кода HDL Coder автоматически использует код соответствующего ей IP-ядра. На практике такой подход имеет существенные недостатки. Помимо дополнительных трудозатрат на создание функциональной модели для каждого подключаемого IP-ядра, использование «черных ящиков» в HDL Coder создает опасность того, что какие-то из особенностей функционирования IP-ядра не будут учтены в замещающей его модели. Это может привести к тому, что функционирование HDL кода, синтезированного в ПЛИС, будет отличаться от результатов, полученных в ходе верификации. Поэтому для проверки соответствия разработанной функциональной модели используемому IP-ядру нельзя ограничиться использованием MATLAB и приходится применять специализированные HDL-симуляторы.

Описанные выше недостатки существующих программных пакетов для верификации АПО ПЛИС обосновывают актуальность создания нового инструмента, который бы был более приспособлен для задач ЦОС. На основании имеющегося опыта разработки блоков ЦОС для ПЛИС можно сформулировать следующие требования к новому программному обеспечению:

- 1) высокая скорость моделирования HDL-кода;
- 2) кроссплатформенность;
- 3) наличие возможности:
 - проводить верификацию из интерфейса MATLAB;
 - формировать тестовые воздействия средствами MATLAB;
 - проводить анализ моделирования средствами MATLAB;
 - проводить автоматическую верификацию без участия человека;
 - на основе HDL-кода автоматически создавать блок Simulink;
 - использовать полученные Simulink модели на компьютерах без специализированных средств моделирования HDL кода;
 - автоматически создавать модели типа «черный ящик» для HDL Coder;
- 4) отсутствие необходимости покупать дополнительные лицензии, кроме MATLAB;

Всем этим требованиям удовлетворяет новый программный пакет Vmodel toolbox, созданный на кафедре проблем управления Института кибернетики МИРЭА [4].

Принцип работы Vmodel toolbox

При разработке данного инструмента моделирования сразу был отвергнут вариант использования внешнего симулятора, который бы связывался в процессе моде-

лирования с MATLAB. По сути это бы являлось попыткой повторить режим со-симуляции, производительность которого на примере коммерческих симуляторов оказалась недостаточной. Вместо этого был выбран путь создания моделей в виде прекомпилированных библиотек MATLAB. Такой подход имеет сразу несколько достоинств: в процессе компиляции код модели оптимизируется, что увеличивает производительность симуляции; обмен между MATLAB и моделью происходит через оперативную память, что также сказывается на быстродействии; созданную модель можно запустить на любом компьютере с MATLAB, что позволяет передать её специалистам работающим над проектом, но не занятым в проектировании АПО ПЛИС.

Основой разработанного инструмента стал симулятор с открытым кодом Verilator [5]. Он позволяет на основе HDL-кода на языке Verilog создавать поведенческие модели на языках C++ или SystemC, выполненные в виде объектов. Эти модели могут быть впоследствии скомпилированы и подключены к внешнему программному обеспечению. Verilator нашел широкое применение при разработке микропроцессорных системы. По результатам сравнения с другими симуляторами отмечается, что скорость моделирования моделей, созданных в Verilator, выше в несколько раз [6, 7].

Алгоритм работы Vmodel toolbox представлен на рис. 1. Пользователь передает в программу HDL-код, который необходимо промоделировать, и конфигурацию будущей модели. Конфигурация включает в себя данные о расположении файлов проекта, типах моделей, которые надо создать, параметрах тактовых сигналов и т.д.

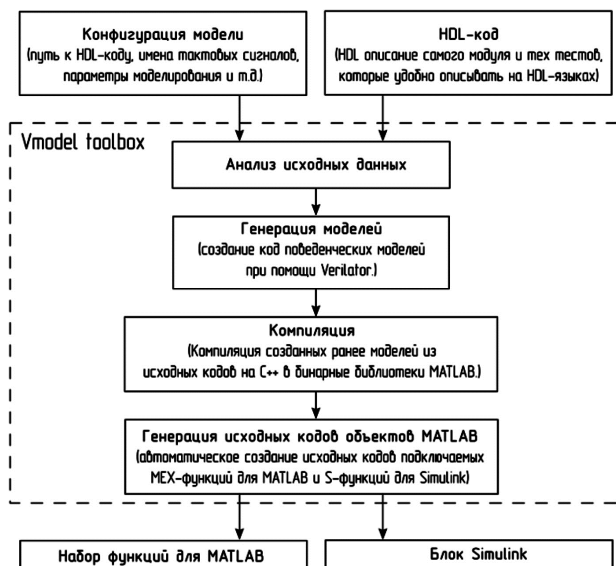


Рис. 1. Алгоритм работы пакета Vmodel toolbox

На основании заданной конфигурации Vmodel toolbox автоматически генерирует исходные коды объектов MATLAB на языке C++, подключает к ним поведенческие модели АПО ПЛИС, которые генерируются при помощи Verilator, и компилирует их в бинарные MEX-файлы и S-функции для MATLAB и Simulink соответственно. В дальнейшем пользователь использует в процессе моделирования только скомпилированные файлы, никак не задействуя Verilator.

Процедура установки

Vmodel toolbox свободно распространяется с открытым кодом по лицензии LGPL[8]. Дистрибутив может быть скачан с официального сайта проекта [9]. Vmodel toolbox требует для своей работы установленный пакет MATLAB с дополнением Simulink версии 2011a или старше. Никакие другие дополнения не требуются. Установка производится автоматически путем запуска скрипта `install_vmodel.m` из MATLAB. Если на компьютере не был установлен Verilator, то он будет автоматически установлен с официального сайта. На сегодняшний день известно об успешном использовании Vmodel toolbox в операционных системах Windows 7, Windows 8, Ubuntu Linux 12.04 LTS, Ubuntu Linux 14.04 LTS. Потенциально программное обеспечение Vmodel toolbox должно работать на любой операционной системе, которая позволяет запустить MATLAB.

Конфигурация модели

Для конфигурации модели необходимо создать в MATLAB структуру конфигурации и передать её функции `vmodel()`, которая станет доступна в пакете MATLAB после установки Vmodel toolbox. Основные параметры конфигурации представлены в табл. 1.

При использовании нескольких тактовых сигналов для каждого из них можно задать частоту, фазовый сдвиг и величину джиттера. В процессе создания модели эти величины пересчитываются относительно указанной в конфигурации частоты ПЛИС. Таким образом если, например, в модели Simulink поднять частоту ПЛИС в 2 раза, то и все внутренние частоты возрастут в 2 раза так, как если бы они были сформированы при помощи блока синтеза частоты в ПЛИС.

Одной из уникальных возможностей Vmodel toolbox является моделирование случайных отказов. Достаточно задать требуемую вероятность отказа в параметр `rchg_probability` при создании модели, и с этой вероятностью будут моделироваться случайные изменения сигналов, выходов регистров и элементов памяти. Данная функция оказывается крайне полезной при разработке блоков ЦОС для систем с высокими требованиями к отказоустойчивости.

Для оценки репрезентативности тестовой выборки в Vmodel toolbox реализована поддержка подсчета покрытия кода тестами, которая доступна как при моделировании в Simulink, так и в MATLAB. Благодаря имеющемуся функционалу, пользователь получает информацию о том, сколько с момента начала моделирования было вхождений в ту или иную строку HDL-кода. При этом существует возможность гибкой настройки того, какие модули участвуют в анализе, какое количество вхождений является минимально-допустимым и какие результаты анализа будут доступны на выходе.

Так как параметры HDL модулей не могут быть изменены в ходе их работы, они должны быть заданы до компиляции модели. Это можно сделать как в ручную, изменив исходный код модуля, так и автоматически при помощи функции `insert_with_params`, которая создает Verilog файл, содержащий экземпляр указанного модуля с заданными из M-языка параметрами. Это открывает возможность автоматизации процесса верификации параметризованных IP-ядер.

Таблица 1. Параметры конфигурации модели Vmodel toolbox

Параметр	Описание
src_filename	Путь к исходному коду главного HDL модуля
Dirlist	Список директорий, где необходимо искать исходные коды
Output	Директория, куда будут помещены созданные поведенческие модели
no_matlab_model	Не создавать модель для MATLAB
no_simulink_model	Не создавать модель для Simulink
constr_name	Имя функции конструктора
sim_name	Имя функции моделирования заданного периода времени
sim_name2	Имя функции моделирования до заданного условия
break_condition	Условие остановки на языке C++ для функции моделирования до заданного условия
fpga_freq	Частота ПЛИС
sample_time	Период расчета модели Simulink
clk_name	Имя тактового сигнала в режиме с одним тактовым сигналом
multiclock	Включить режим с несколькими тактовыми сигналами
Clocks	Структура описания тактовых сигналов в режиме с несколькими тактовыми сигналами
random_seed	Положительное 32-битное число, которым будут проинициализированы генераторы случайных чисел, которые используются при моделировании джиттера и случайных отказов. Параметр необходим для обеспечения повторяемости экспериментов
Signals	Параметр определяет какие сигналы будут отображаться пользователю по результатам моделирования: 'all' – все доступные сигналы; 'top' – только выходы главного модуля; 'public' – выходы главного модуля и сигналы, помеченные в коде комментарием /*verilator public*/; 'blackbox' – аналог 'top', но создается «черный ящик» для HDL Coder
clk_to_out	Добавить состояние тактовых генераторов в результаты моделирования
inputs_to_out	Добавить состояние входов в результаты моделирования
reinterpret_floats	Параметр определяет будут ли значения переданные из MATLAB в формате с плавающей точкой округлены и присвоены как целые числа или побитно присвоены в формате IEEE754
rchg_probability	Вероятность случайного отказа. Если более 0, то при моделировании HDL будут имитироваться случайные отказы элементов ПЛИС с этой вероятностью.
rchg_full_mem	Определяет каким образом обеспечивается вероятность отказа памяти: для блока памяти целиком или для каждой ячейки в отдельности
rchg_show_param	Добавить служебные данные об общем количестве произошедших отказов в результаты моделирования
verilator_keys	Опциональные ключи для Verilator
coverage	Включить подсчет покрытия кода тестами
coverage_keys	Опциональные ключи конфигурации покрытия кода тестами

Процедура моделирования в MATLAB

По результатам компиляции модели vmodel создает в заданной папке 3 функции:

1) конструктор (создает структуру, через которую пользователь передает модели текущее состояние входов HDL модуля);

2) функцию моделирования заданного периода времени, при фиксированном состоянии входов модели (аналог команды # из несинтезируемого подмножества языка Verilog);

3) функцию моделирования до заданного условия (эта функция создается только в том случае, если соответствующее условие было задано в конфигурации модели).

Имена функций и папки их расположения настраи-

ваются при помощи структуры конфигурации (табл. 1).

Используя созданные vmodel функции в скрипте на М-языке, пользователь может изменять состояние входов модели (через структуру, созданную конструктором) и проводить моделирование. Для создания тестовых воздействий и анализа результатов моделирования доступен весь широкий арсенал средств MATLAB, включающий в себя функции матричных и скалярных операций, статистического и частотного анализа, двумерного и трехмерного графического отображения.

Для иллюстрации процесса моделирования при помощи Vmodel toolbox рассмотрим скрипт на М-языке, представленный на листинге 1. Подразумевается, что к моменту выполнения данного скрипта функция vmodel уже создала в текущей папке функцию конструктора с именем «constructor», функцию моделирования заданного

периода времени (`sim_step`) и функцию моделирования до заданного условия (`sim_tcond`). При это в качестве условия остановки моделирования задано равенство единице сигнала об окончании расчета в HDL модуле.

По умолчанию время в параметрах функций моделирования указывается в тактах задающего генератора ПЛИС. Это удобно для отладки HDL-модулей с одним входом тактового генератора. В случае моделирования в режиме с несколькими тактовыми генераторами, временные параметры указываются в тактах главного генератора. Если же требуется указать время моделирования в системе СИ, то это возможно с использованием функции «`t2f`». В строке 6 листинга 1 представлен пример применения этой функции для установки времени моделирования в 30 мкс.

Листинг 1. Пример моделирования при помощи Vmodel_toolbox в MATLAB

```

1: uut=constructor;           %Создаем объект uut
2: uut.rst=1;                %Устанавливаем вход rst
                               HDL модуля в 1
3: sim_step(uut,1);         %Моделируем 1 такт
                               расчета ПЛИС
4: uut.rst=0;                %Устанавливаем вход rst
                               HDL модуля в 0
5: uut.in=-24;               %Устанавливаем вход in
                               HDL модуля в -24
6: sim_step(uut,t2f(30,'us')); %Моделируем 30 мкс
                               работы ПЛИС
7: uut.start=1;              %Устанавливаем вход
                               start HDL модуля в 1
8: sim_step(uut,1);         %Моделируем 1 такт
                               расчета ПЛИС
9: uut.start=0;              %Устанавливаем вход
                               start HDL модуля в 0
10: res=sim_tcond(uut,5000); %Моделируем работу
                               ПЛИС до заданного
                               условия%или в течении
                               5000 тактов в зависимо-
                               стей от того, % что
                               наступит ранее. Резуль-
                               тат помещаем в res.
11: display(res.out);        %Отображаем значение
                               выхода out HDL модуля
                               в %консоль

```

Несмотря на то, что MATLAB представляет широкие средства для визуального отображения данных, в нем отсутствуют некоторые привычные для HDL разработчика

способы визуализации изменения сигналов. Восполняя этот пробел, Vmodel toolbox добавляет в MATLAB функцию «`digital`» для отображения временных диаграмм изменения цифровых сигналов (рис. 2) и функцию «`bin_stairs`» для отображения побитной временная диаграммы (рис. 3).

В отличие от многих средств моделирования временные диаграммы, построенные Vmodel toolbox, позволяют отображать числовые значения сигналов не только в различных целочисленных формах, но и в виде дробных чисел с фиксированной или плавающей точкой. При этом есть поддержка систем исчисления с нестандартными размерами мантиссы и экспоненты.

Процедура моделирования в Simulink

Функция `vmodel` создает Simulink блок поведенческой модели HDL кода. Он имеет всего два параметра: частота работы ПЛИС и период расчета Simulink. Из соотношения этих параметров определяется то, как часто меняются значения на входах модели HDL кода и сколько тактов тактового генератора ПЛИС приходится на один период расчета среды Simulink. С точки зрения типовых задач ЦОС период расчета Simulink можно рассматривать как период обновления сигналов на выходах АЦП, подключенных к ПЛИС.

На этапе конфигурации модели можно указать, что необходимо создать не просто Simulink блок, а модель «черный ящик» для HDL Coder. В этом случае в процессе генерации кода модель будет заменена на соответствующий ей HDL код. Такой способ создания «черных ящиков» для HDL Coder обладает существенными преимуществами по сравнению с ручным созданием их из базовых блоков Simulink, так как весь процесс происходит автоматически, а на выходе разработчик получает точную модель, которая учитывает все особенности функционирования HDL-кода.

Для обеспечения совместимости среды Simulink с сигналами, имеющими нестандартную разрядность, в состав `vmodel` входит специальный Simulink блок `Signer`. Он преобразует целочисленный знаковый сигнал произвольной разрядности с выхода модели в стандартный для Simulink тип `Double`.

На рис. 4 представлен процесс моделирования IP-блока обработки синусно-косинусного датчика в составе модели Simulink.

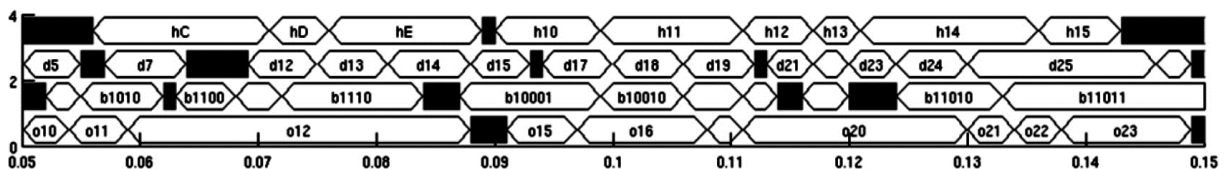


Рис. 2. Временных диаграмм изменения цифровых сигналов, построенная при помощи ПО vmodel

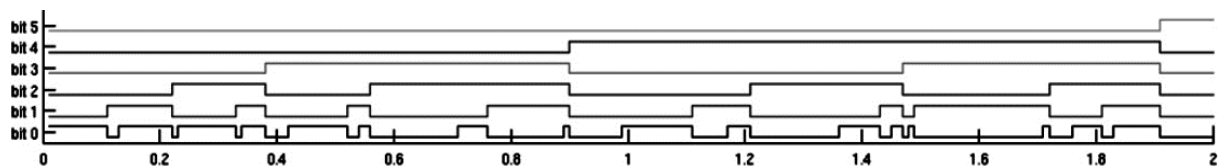


Рис. 3. Побитная временная диаграмма, построенная при помощи ПО vmodel

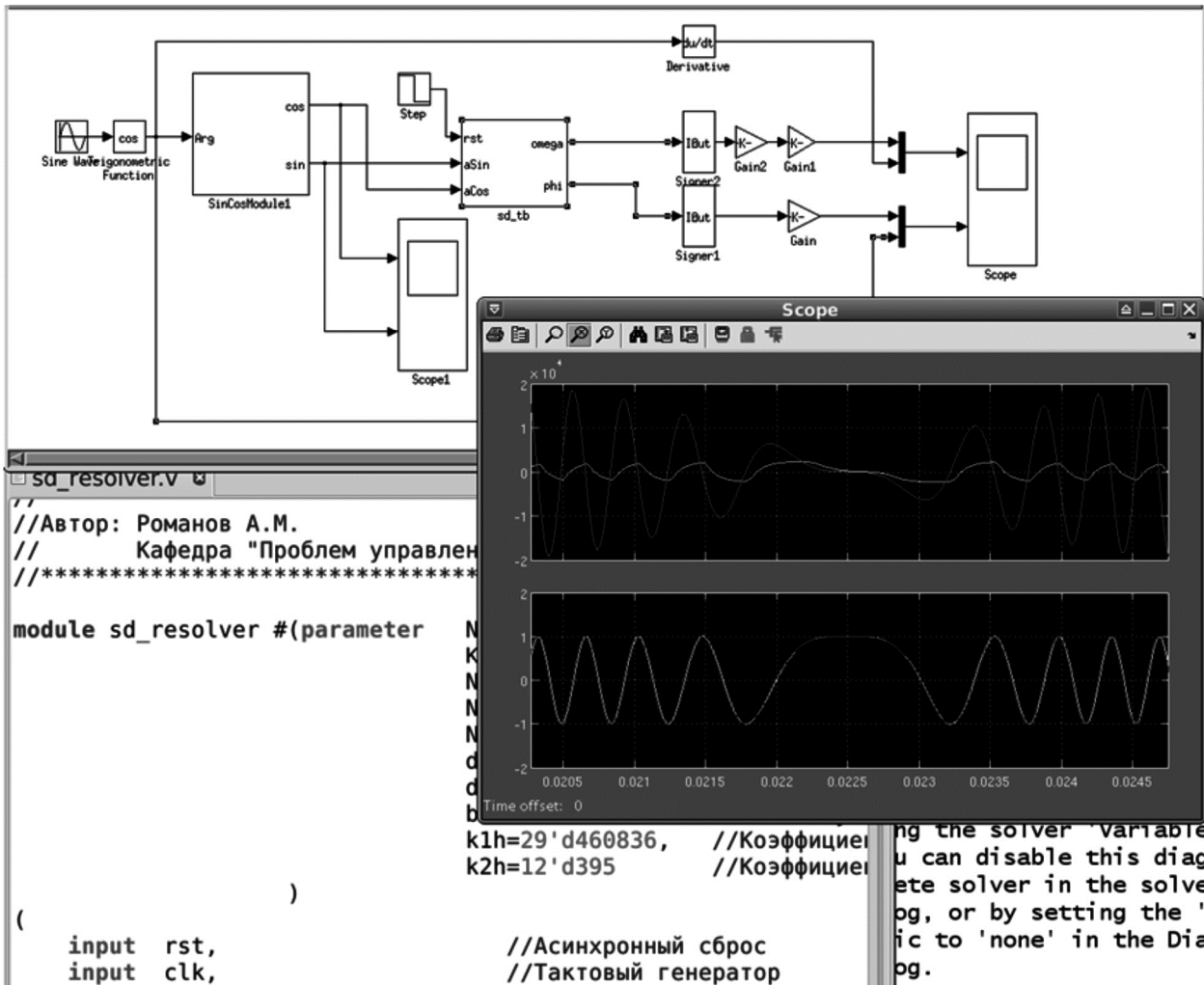


Рис. 4. Процесс моделирования IP-блока обработки синусно-косинусного датчика в составе модели Simulink

Так как на выходе Vmodel toolbox создается готовый Simulink блок, то существенных отличий по сравнению с обычным порядком моделирования в Simulink нет. Пользователю необходимо скопировать созданный блок в отдельную рабочую область Simulink и подать на него тестовые сигналы с требуемыми характеристиками. Для формирования сигналов сброса удобно использовать блок Step из стандартной библиотеки Simulink. По умолчанию выходы Simulink модели, созданной Vmodel tool-box, представлены в виде 32-битных беззнаковых чисел стандарта «uint32». В то время как стандартным типом в Simulink являются числа с плавающей точкой формата Double. Для сопряжения выходов модели с другими сигналами можно использовать как стандартные блоки «Data Type Conversion» из библиотеки Simulink, так и блоки Signer, предоставляемые Vmodel toolbox. В отличие от выходных сигналов, для входных сигналов приведение типов не требуется. Вне зависимости от типа, любой поданный на модель сигнал будет округлен и подан на HDL модуль в целочисленном виде.

Передача моделей другим разработчикам

Популярный сейчас модельно-ориентированный подход проектирования [10, 11] подразумевает работу различных специалистов над общей моделью системы.

Каждый из них работает над своей частью модели, используя остальные её компоненты для верификации. При таком подходе в случае использования со-симуляции специализированные HDL-симуляторы должны быть установлены на компьютерах всех участников проекта. Помимо ощутимых затрат на лицензии, это требует от специалистов, не имеющих отношения к разработке АПО ПЛИС, дополнительных компетенций. Альтернативой является использование упрощенных моделей на основе базовых блоков Simulink, которые могут быть неточными и не учитывать особенности реализации на ПЛИС.

Все модели, созданные при помощи vmodel, после компиляции могут быть использованы на любом компьютере с установленным пакетом MATLAB. Размер созданных моделей, как правило, не превышает 1 Мбайта, что позволяет без проблем распространить их между всеми участниками проекта.

В процессе создания поведенческой модели и её последующей компиляции она несколько раз оптимизируется. Таким образом, восстановить исходный код модуля по модели, созданной при помощи Vmodel toolbox, нельзя. Это позволяет беспрепятственно передавать модели IP-ядер в третьи организации, не боясь, что заложенное в них ноу-хау, будет утеряно.

Результаты внедрения

Программа Vmodel toolbox прошла государственную регистрацию и была успешно апробирована в ряде проектов, связанных с ЦОС, на основе которых можно судить об её эффективности.

Верификация матричного сопроцессора

Целью проекта было создание набора IP-ядер ПЛИС, предназначенного для ускорения матричных операций таких как: транспонирование, сложение, умножение и инверсия матриц, формирование диагональных и треугольных матриц, декомпозиция Холецкого и т.д. В результате был создан математический сопроцессор оригинальной архитектуры [12]. Для его полноценной верификации потребовалось создать большое количество тестов, включающих в себя множество матричных операций. Использование Vmodel toolbox позволило существенно упростить этот процесс, так как MATLAB уже содержит мощный аппарат матричных вычислений. В то же время функции построения временных диаграмм, входящие в состав vmodel позволили успешно отлаживать сложные конвейеры при помощи привычных для HDL-разработчиков средств, избежав применения других симуляторов.

Верификация устройств

с семисегментными индикаторами

В рамках проекта требовалось провести верификацию устройства на базе ПЛИС, одним из интерфейсов которого являлся семисегментный индикатор на 8 знаков. Применение Vmodel toolbox и встроенных функций отображения графики в MATLAB позволили вывести результаты индикации в том виде, в котором они будут отображаться пользователю конечного устройства. Такой подход существенно упростил работу инженера-тестировщика и сократил сроки верификации. Пример тестового скрипта на М-языке и отображения результатов в MATLAB представлен на рис. 5.

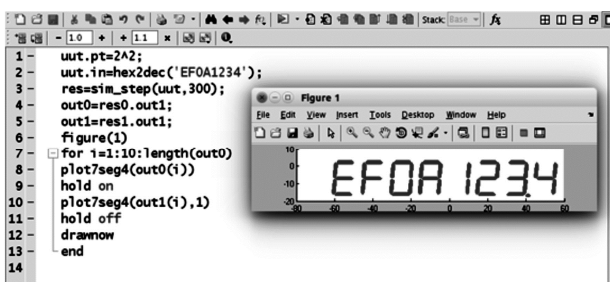


Рис. 5. Процесс моделирования IP-блока обработки синусно-косинусного датчика в составе модели Simulink

Верификация библиотеки IP-ядер для управления двигателями

Задачей проекта была реализация библиотеки IP-ядер для систем управления двигателями различного типа. Библиотека включает в себя траекторные задатчики, регуляторы для двигателей постоянного и переменного тока, а также блоки обработки датчиков. Одним из требований проекта была совместимость IP-ядер с бюджетными ПЛИС, обладающими малой логической ёмкостью. Для выполнения этого требования большая часть ЦОС в рамках проекта была реализована при по-

мощи «импульсных» математических операций [13]. Особенностью данного метода является представление информационных сигналов внутри блоков ЦОС в форме высокочастотной сигма-дельта модуляции. Просмотр временных диаграмм для таких сигналов не имеет смысла, а для восстановления информации из них требуется применение фильтров низких частот. Благодаря использованию Vmodel toolbox, весь анализ результатов проводился средствами MATLAB, поэтому реализация низкочастотной фильтрации не потребовала дополнительных трудозатрат. Еще одна сложность в рамках проекта заключалась в отсутствии возможности заранее сформировать тестовые воздействия: в связи с большим количеством обратных связей в системах управления двигателями входы IP-ядер на каждом такте расчета зависели от значений на выходе ядра на предыдущем такте. При этом длительность переходных процессов, которые требовалось моделировать, измерялась секундами, что соответствует сотням миллионов тактов расчета ПЛИС. Благодаря высокой скорости расчета моделей Simulink, созданных при помощи Vmodel toolbox, удалось выполнить весь объем работ по верификации проекта в установленные сроки. Пример моделирования IP-ядра обработки синусно-косинусного датчика в Simulink представлен на рис. 4.

Верификация цифровой системы обработки видео потока

При реализации на ПЛИС цифровой системы видеообработки, построенной на базе применения двумерного дискретного преобразования Фурье, стояла задача выбора оптимального значения разрядностей видеопамати и коэффициентов преобразования. Применение Vmodel toolbox позволило автоматически промоделировать тракт видеообработки при различных значения параметров HDL-модулей, определяющих требуемые разрядности. А при помощи средств визуализации MATLAB удалось отобразить содержимое видеопамати на экране оператора без использования дополнительного программного обеспечения. В результате разработчик имеет возможность с минимальными трудозатратами оценить качество изображения в каждом случае и выбрать наиболее подходящие значения параметров модуля. Примеры результатов моделирования тракта видеообработки, полученные при различных разрядностях коэффициентов преобразования Фурье представлены на рис. 6.

Заключение

Функционал существующих средств моделирования HDL-кода не достаточен для решения задач ЦОС. Для формирования тестовых воздействий и анализа результатов моделирования разработчики используют пакеты математического моделирования, наиболее популярным из которых является MATLAB. Для задач управления, где необходимо моделировать взаимодействие блоков ЦОС на базе ПЛИС с аналоговыми объектами, единственным приемлемым решением до недавнего времени являлось использование MATLAB в режиме симуляции с внешним HDL-симулятором. Однако такой режим имеет низкую скорость моделирования.

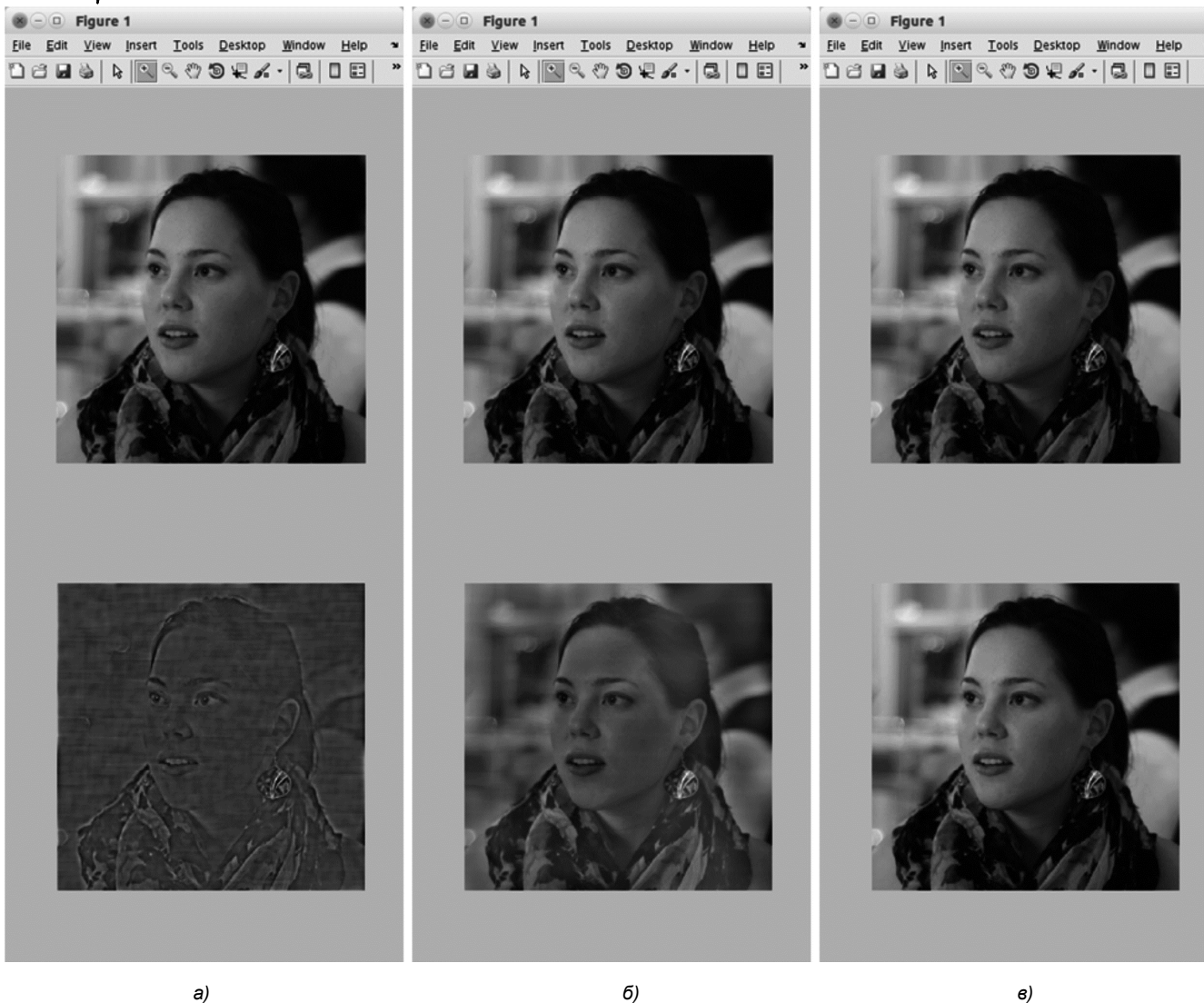


Рис. 1. Результаты моделирования тракта видеобработки в MATLAB при различных разрядностях коэффициентов преобразования Фурье; верхний ряд – эталонное изображение, нижний ряд — результат обработки при помощи IP-ядра;

а) 2x16 бит на точку б) 2x20 бит на точку в) 2x28 бит на точку

Предложенный в данной работе инструмент Vmodel toolbox позволяет проводить отладку HDL-кода без использования внешних HDL-симуляторов, только при помощи интерфейса MATLAB. Предварительная компиляция поведенческих моделей, которая используется в Vmodel toolbox, обеспечивает не только высокую скорость моделирования, но и возможность их использования на компьютерах без специализированного программного обеспечения для работы с HDL-кодом. Благодаря этому существует возможность передать точную поведенческую модель IP-ядра третьим лицам без раскрытия исходного кода.

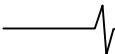
Использование средств MATLAB для формирования тестовых воздействий, анализа и визуализации результатов моделирования позволяет поднять верификацию блоков ЦОС на новый уровень. Тестовое окружение, созданное на этапе прототипирования алгоритма ЦОС, может быть без изменений использовано и для отладки HDL-кода, а для визуализации результатов моделирования больше не требуется выгружать данные в стороннее программное обеспечение.

Совместимость Vmodel toolbox с HDL Coder позво-

ляет повысить эффективность работы с этим инструментом, за счет полной автоматизации создания функциональных моделей существующих IP-ядер. Опыт успешного применения программного обеспечения Vmodel toolbox в реальных проектах показал, что оно не только удовлетворяет всем требованиям, предъявляемым к средствам верификации АПО ПЛИС в задачах ЦОС, но и позволяет обойтись без использования каких-либо других симуляторов при отладке HDL-кода.

Литература

1. Попов А.Ю. Проектирование цифровых устройств с использованием ПЛИС: Учеб. пособие. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2009. – 80 с.
2. Sandre-Hernandez, O.; Rangel-Magdaleno, J.J.; Morales-Caporal, R., «Simulink-HDL cosimulation of direct torque control of a PM synchronous machine based FPGA,» Electrical Engineering, Computing Science and Automatic Control (CCE), 2014 11th International Conference on, vol., no., pp.1, 6, Sept. 29 2014-Oct. 3 2014
3. Besbes, M.; Said, S.H.; M'Sahli, F., «FPGA implementation of high gain observer for induction machine using Simulink HDL



coder,» in Control, Engineering & Information Technology (CEIT), 2015 3rd International Conference on , vol., no., pp.1-6, 25-27 May 2015

4. Свидетельство о государственной регистрации программы для ЭВМ №2015612744 (Vmodel toolbox)

5. Wilson S. «Introduction to Verilator» [Электронный ресурс] = Intro-Verilator-Veripool – Режим доступа: <http://www.veripool.org/wiki/verilator>, свободный. – Загл. с экрана.

6. Wilson S. «Verilog Simulator Benchmarks» [Электронный ресурс] = Verilog Simulator Benchmarks-Veripool – Режим доступа: http://www.veripool.org/wiki/veripool/Verilog_Simulator_Benchmarks, свободный. – Загл. с экрана.

7. Ahmad, T.B.; Ciesielski, M., «Parallel Multi-core Verilog HDL Simulation Using Domain Partitioning,» in VLSI (ISVLSI), 2014 IEEE Computer Society Annual Symposium on, vol., no., pp.619-624, 9-11 July 2014

8. T.A. Alspaugh «GNU Lesser General Public License, version 3» [Электронный ресурс] = LGPLv3 numbered – Режим

доступа: <http://www.thomasalspaugh.org/pub/osl-sps/gpl3.0.html>, свободный. – Загл. с экрана.

9. Романов А.М. Vmodel toolbox repository [Электронный ресурс] = amromanov/vmodel – Режим доступа: <https://github.com/amromanov/vmodel>, свободный. – Загл. с экрана.

10. Sudhir Sharma, Wang Chen, «Using Model-Based Design to Accelerate FPGA Development for Automotive Applications», in 2009 SAE World Congress, January 1-15, 2009.

11. Kelemenová, Tatiana, et al. «Model Based Design and HIL Simulations» American Journal of Mechanical Engineering 1.7 (2013): 276-281.

12. A. Romanov, B.Slaschov FPGA-based Kalman filtering for motor control //NSCE2014 Transactions, CRC Press, 2014 г.

13. А.М. Романов Анализ и синтез элементов устройств управления мехатронно-модульными системами на базе ПЛИС с использованием сигма-дельта модуляции // Естественные и технические науки. – М.: «Компания Спутник+», 2013. – № 6.

УДК 621.382

СПОСОБЫ РЕАЛИЗАЦИИ НА ПЛИС МНОГОКАНАЛЬНЫХ АДАПТИВНЫХ КИХ-ФИЛЬТРОВ ДЛЯ СИСТЕМ АКТИВНОГО ПОДАВЛЕНИЯ АКУСТИЧЕСКОГО ШУМА*

Ушенина И.В., к.т.н., доцент кафедры вычислительных машин и систем Пензенского государственного технологического университета, e-mail: ivl23@yandex.ru.

FPGA IMPLEMENTATION TECHNIQUES FOR MULTI-CHANNEL ADAPTIVE FIR FILTERS OF ACTIVE NOISE CONTROL SYSTEMS

Ushenina I.V.

Adaptive filters implementation for active noise control (ANC) systems may require high-performance DSP devices. This paper shows a dependency of ANC system performance required from ANC system's parameters, and analyses multi-channel adaptive filters' structures possible for FPGA sources. Some FPGA-implementation results of a structure recognized as the most suitable are also represented.

Key words: active noise control, multi-channel adaptive filtering, FPGA.

Ключевые слова: активное подавление шума, многоканальная адаптивная фильтрация, ПЛИС.

Введение

Способ подавления акустического шума противофазным компенсирующим сигналом был предложен еще в 1933 году, однако его использование для решения многих практических задач стало возможным лишь с появлением высокопроизводительных устройств цифровой обработки сигналов [1, 2].

Наиболее распространены системы активного подавления шума (АПШ) с опорным сигналом [1-4], которые рассматриваются и в данной работе. В состав таких систем входят микрофоны или датчики другого типа, регистрирующие шум в выбранной точке (опорный сигнал x) и результат взаимодействия шума с компенсирующим сигналом (сигнал ошибки e). На основе x и e контроллер системы АПШ вырабатывает компенсирующий сигнал y , для излучения которого чаще всего используется динамик [1].

Контроллер большинства систем активного подавления шума (АПШ) представляет собой адаптивный фильтр (АФ), реализующий алгоритм наименьших средних квадратов с предварительной фильтрацией опорного сигнала (FXLMS, filtered-x least mean square [5]).

На рис. 1 представлена структура системы АПШ с опорным сигналом. Здесь: $P(z)$ – передаточная функция акустического пути распространения шума от опорного микрофона до точки суммирования шума и компенсирующего сигнала; $W(z)$ – передаточная функция АФ; $S(z)$ – передаточная функция пути компенсации, включающего

Реализация адаптивных фильтров систем активного подавления акустического шума (АПШ) может требовать высокопроизводительных устройств цифровой обработки сигналов. Показана зависимость требуемой производительности системы АПШ от ее параметров и проанализированы возможные способы структурной организации многоканальных адаптивных фильтров на ресурсах ПЛИС FPGA. Для структуры фильтра, признанной по итогам анализа наилучшей, приведены некоторые результаты ее реализации на FPGA.

динамик – от выхода АФ до точки суммирования; $E(z)$ – передаточная функция пути сигнала ошибки, включающего микрофон ошибки – от точки суммирования до входа блока адаптации; $\hat{S}(z)$ – оценка передаточных функций пути компенсации и пути сигнала ошибки, используемая алгоритмом FXLMS для предварительной фильтрации опорного сигнала; $x(n)$, $x'(n)$, $e(n)$, $y(n)$ – действующие на шаге дискретизации n опорный сигнал, результат предварительной фильтрации опорного сигнала, отсчет компенсирующего сигнала и сигнал ошибки соответственно. Фильтр $W(z)$, блок адаптации и фильтр-оценка $\hat{S}(z)$ входят в состав контроллера системы АПШ.

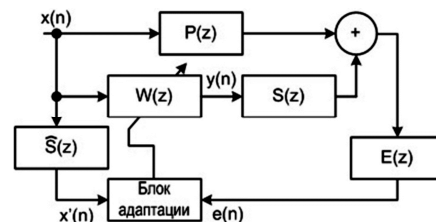


Рис. 1. Система АПШ с опорным сигналом

Параметры систем АПШ и требуемая производительность контроллера

Производительность, требуемая от контроллера системы АПШ, в том числе требуемое количество операций умножения с накоплением в секунду, зависит от спектрального состава подавляемого шума и условий его распространения (геометрия области распростране-

*Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта №14-07-31091 мол_а

ния шума, свойства отражающих шум поверхностей, особенности $P(z)$ и $S(z)$). Данные факторы определяют требования к количеству каналов системы, а также к порядкам АФ (N) и фильтров-оценок (L).

Система АПШ в общем случае содержит J опорных микрофонов, M микрофонов ошибки и K динамиков, компенсирующих шум. Каждая пара «опорный микрофон-динамик» имеет собственный АФ (рис. 2). Таким образом, количество АФ в многоканальной системе АПШ равно $J \cdot K$. На каждый k -й динамик поступает сумма Y_k результатов обработки всех J опорных сигналов системы (1):

$$Y_k(n) = \sum_{j=1}^J y_{kj}(n). \quad (1)$$

Каждый $y_{kj}(n)$ является результатом фильтрации j -го опорного сигнала k -м АФ (2):

$$y_{kj}(n) = \sum_{i=0}^{N-1} x_j(n-i) \cdot w_{kj}^i(n), \quad (2)$$

где $Y_k(n) = \sum_{j=1}^J y_{kj}(n)$ – i -й из N коэффициентов k -го АФ на n -м шаге дискретизации.

В многоканальной системе должны присутствовать $M \cdot K$ фильтров-оценок, соответствующих $M \cdot K$ парам «динамик-микрофон ошибки». В результате фильтрации векторов $X_j(n)$ отсчетов опорного сигнала фильтрами-оценками (3) в каждом интервале дискретизации получаем $M \cdot K \cdot J$ значений $x'_{mkj}(n)$:

$$x'_{mkj}(n) = \sum_{i=0}^{L-1} x_j(n-i) \cdot \hat{s}_{mk}^i, \quad (3)$$

где \hat{s}_{mk}^i – i -й из L коэффициентов mk -го фильтра-оценки с передаточной функцией $\hat{S}_{mk}(z)$.

В формулах (2) и (3) предполагаем, что порядки N_{kj} всех АФ одинаковы и равны N ; порядки L_{mk} всех фильтров-оценок путей компенсации одинаковы и равны L .

Критерием оптимизации коэффициентов АФ является минимум суммы квадратов сигналов ошибки. Блок адаптации каждого k -го АФ принимает сигналы $e_m(n)$ от всех имеющихся в системе микрофонов ошибки, а также результаты обработки j -го опорного сигнала оценками передаточных функций между k -м динамиком и каждым из микрофонов ошибки. Расчет коэффициентов k -го АФ осуществляется по формуле (4):

$$w_{kj}^i(n+1) = w_{kj}^i(n) + \mu \cdot \sum_{m=1}^M e_m(n) \cdot x'_{mkj}(n-i), \quad (4)$$

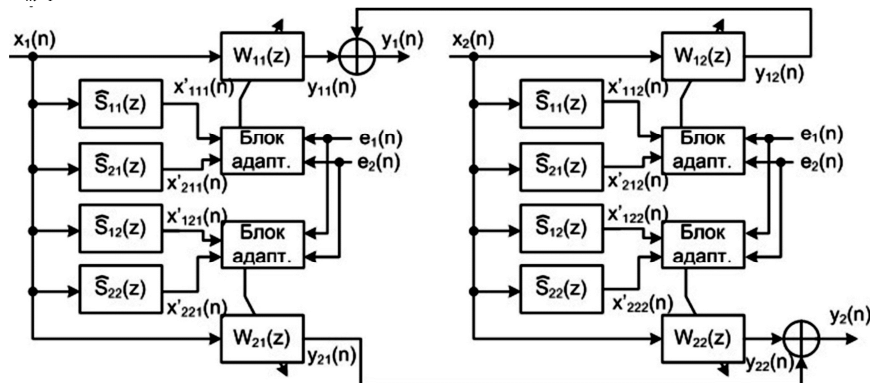


Рис. 2. Структура контроллера системы АПШ при $J=K=M=2$

где $\mu \ll 1$ – шаг сходимости адаптивного алгоритма [5].

Таким образом, на каждом интервале дискретизации должно быть выполнено $J \cdot K \cdot N$ умножений с накоплением для расчета y_{kj} , $J \cdot K \cdot M \cdot L$ умножений с накоплением для расчета набора $x'_{mkj}(n)$ и $M \cdot K \cdot J \cdot N$ умножений с накоплением для расчета новых коэффициентов АФ. Количество P операций умножения с накоплением, выполняемое контроллером в секунду, должно составлять (5):

$$P = F_s \cdot [(N+L) \cdot M \cdot K \cdot J + N \cdot K \cdot J], \quad (5)$$

где F_s – частота дискретизации системы АПШ, превышающая частоту подавляемого шума приблизительно в 10 раз [1].

Значения J , K , M на практике составляют несколько десятков, а в отдельных случаях достаточно $J=K=M=1$ [1, 4]. Выбор N зависит от спектрального состава подавляемого шума, а также характеристик акустического пути и пути компенсации. АФ совместно с передаточной функцией пути компенсации должны как можно точнее воспроизводить передаточную функцию акустического пути распространения шума на частотах подавляемого шума. Если спектр шума содержит несколько дискретных составляющих, на каждую из них АФ должен содержать по 4-20 коэффициентов [1, 3]. Для различных задач подавления случайного шума со сплошным спектром N может составлять от нескольких сотен до одной-двух тысяч коэффициентов [2, 3, 6, 7]. L выбирается с учетом того, что ФЧХ фильтра-оценки не должна отличаться от ФЧХ оригинала более чем на 45° , чтобы избежать нестабильности АФ [1]. При невысоких порядках АФ L часто выбирается равным N [2]. Далее будем полагать, что $L \leq N$.

Контроллеры систем АПШ чаще всего реализуются на цифровых сигнальных процессорах (ЦСП) [1, 2]. Подстановка в (5) значений M , K , J , N , L , F_s , соответствующих многоканальной системе АПШ для подавления случайного низкочастотного шума, показывает, что производительность контроллера должна составлять десятки или сотни GMACs (миллиардов операций умножения с накоплением в секунду). Это соответствует максимуму возможностей наиболее производительных современных многоядерных ЦСП [8] или превышает его. Существует практика реализации контроллеров систем АПШ на нескольких ЦСП [1, 3], однако с повышением требований к производительности целесообразно рассмотреть возможность использования ПЛИС FPGA. Эти устройства содержат до нескольких тысяч ячеек цифро-

вой обработки сигналов, и обеспечивают производительность порядка 10^{12} операций умножения с накоплением в секунду даже при более низкой тактовой частоте, чем у ЦСП.

Количество ЦОС-ячеек, сравнимое с порядками АФ, и программируемые соединения между ресурсами FPGA дают возможность выбора структурной организации контроллера. В оставшейся части статьи выполняется анализ возможностей структурной организации контроллера многоканальной системы АПШ на базе FPGA последнего поколения [9], с учетом количества, ресурсов и структуры ЦОС-ячеек этих устройств.

Критерии выбора структуры контроллера на ЦОС-ячейках FPGA

Можно выделить два подхода к структурной организации контроллера:

- контроллер в виде единой структуры из N ЦОС-ячеек, выполняющей параллельную фильтрацию с временным разделением ресурсов [10];

- контроллер в виде массива независимых друг от друга ЦОС-ячеек, каждая из которых последовательно выполняет N или L умножений с накоплением, обрабатывая собственный набор сигналов.

В рамках каждого из подходов можно выделить несколько вариантов структур контроллера. Выберем в каждом из подходов наилучшие варианты и сравним их.

Критериями выбора являются: 1) независимость максимальной тактовой частоты структуры c_{max} от порядков фильтров и количества каналов; 2) соответствие структуры возможностям современных FPGA по ресурсоемкости и быстродействию.

Соответствие первому критерию обеспечивает регулярная конвейеризированная структура, которую возможно разделить на одинаковые, взаимозаменяемые блоки. При этом увеличение N , L , M , K , J в идеале не должно вносить изменений в блоки, а приводить только к увеличению их количества.

Соответствие второму критерию отчасти достигается за счет нахождения баланса между объемами параллельных и последовательных вычислений, выполняемых в каждом интервале дискретизации. С другой стороны, требуемый объем памяти для хранения отсчетов входных сигналов, коэффициентов фильтров и результатов промежуточных вычислений определяется параметрами N , L , M , K , J , а не структурой контроллера. Так, при 16-разрядном представлении чисел для хранения коэффициентов АФ требуется $16 \cdot N \cdot K \cdot J$ бит памяти; для хранения коэффициентов фильтров-оценок – $16 \cdot L \cdot M \cdot K$ бит; для хранения J векторов из N отсчетов опорных сигналов – $16 \cdot N \cdot J$ бит. Для хранения N наборов из $M \cdot K \cdot J$ значений результатов предварительной фильтрации опорных сигналов – $16 \cdot N \cdot M \cdot K \cdot J$ бит, т.е. порядка половины от общего требуемого объема памяти. Для экономии памяти целесообразно сохранять не сами значения $x'_{mkj}(n-i)$, а результат их умножения на μ . Если μ является отрицательной степенью двойки, требуется сохранить несколько старших разрядов каждого из значений $x'_{mkj}(n-i)$.

Возможности реализации контроллера как единой структуры из N ЦОС-ячеек

Многоканальный адаптивный FXLMS-фильтр может быть построен добавлением к классической структуре параллельного КИХ-фильтра блоков адаптации и предварительной фильтрации опорных сигналов и обеспечением временного разделения ресурсов.

Из трех вариантов структур параллельных фильтров – систолической, транспонированной или «бинарного дерева» [10, 11] – для доработки и последующего анализа выбран первый. Транспонированная структура при адаптивной фильтрации будет выполнять расчет выходного сигнала с использованием коэффициентов АФ, рассчитанных на N последних интервалах дискретизации [11], тогда как алгоритм FXLMS предписывает использовать набор коэффициентов, рассчитанных только на интервале, предшествующем текущему. Кроме того, у транспонированной структуры один и тот же отсчет $x_j(n)$ подается сразу на все умножители, что может растянуть переходные процессы при переключении уровня сигнала. Структура «бинарное дерево» не соответствует первому из сформулированных выше критериев, т.к. помимо регулярной последовательности ЦОС-ячеек требует дополнительных внешних сумматоров, количество которых составляет половину порядка фильтра.

Систолическая структура состоит из одинаковых, последовательно соединенных блоков, которые в случае реализации одноканального неадаптивного фильтра полностью отображаются на ресурсы ЦОС-ячеек FPGA, не требуя дополнительно программируемых элементов кристалла. Каждый отсчет входного сигнала последовательно поступает на все ЦОС-ячейки по очереди, продвигаясь по цепочке регистров, поэтому зависимость максимума тактовой частоты от N отсутствует.

При реализации многоканального КИХ-фильтра параллельной систолической структуры вычислительное ядро каждого отвода (сумматор, умножитель) реализуется на ЦОС-ячейке, а для формирования и хранения последовательности обрабатываемых отсчетов, хранения коэффициентов фильтров и продвижения отсчетов входных сигналов должны использоваться программируемая логика и блочное ОЗУ [10].

Структура контроллера многоканальной системы АПШ, основанная на систолической структуре параллельного КИХ-фильтра, приведена на рис. 3. Элементы контроллера, входящие в состав ЦОС-ячеек, выделены. Исходя из предположения, что $N \geq L$, структура должна иметь N ЦОС-ячеек. Каждая i -я из N ячеек умножает $x_j(n-i)$ на коэффициенты $w_{kj}^i(n)$ и s_{mk}^i , а также выполняет перерасчет набора i -х коэффициентов $K \cdot J$ адаптивных фильтров.

С каждой i -й ЦОС-ячейкой связаны блоки ОЗУ, хранящие i -е коэффициенты каждого из АФ и фильтров-оценок. Сигналы ошибки поступают на все ЦОС-ячейки от общего блока памяти. На один из входов каждого i -го умножителя от элементов задержки $Z^{-J(KM+1)}$ подаются $x_j(n-i)$ и $\mu \cdot x'_{mkj}(n-i)$; на другой вход через мультиплексор 1 – коэффициенты $w_{kj}^i(n)$, s_{mk}^i или отсчеты сигналов ошибки $e_m(n)$. Через мультиплексор 2 на один

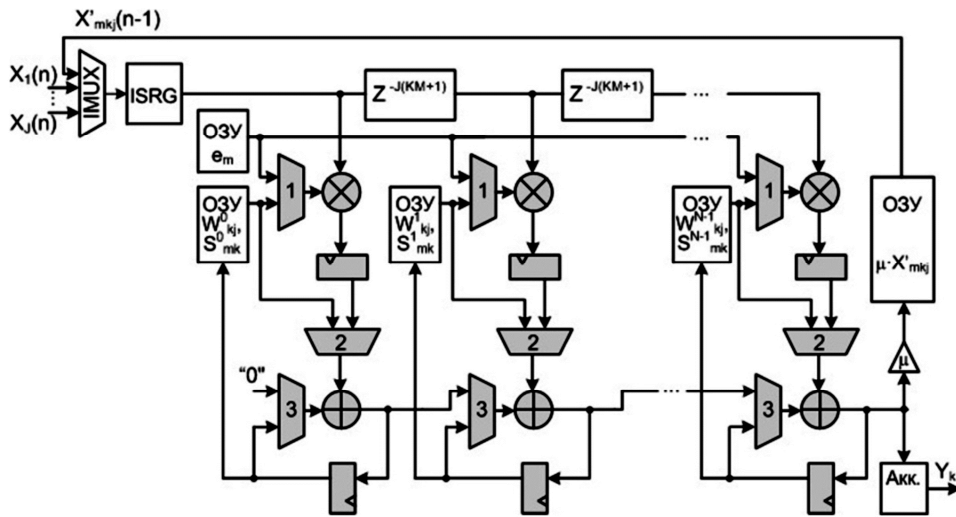


Рис. 3. Единая структура контроллера многоканальной системы АПШ

из входов каждого i -го сумматора при фильтрации поступает результат умножения $x_j(n-i)$ на $w_{kj}^i(n)$ или на s_{mk}^i ; при вычислении коэффициентов АФ – коэффициент $w_{kj}^i(n)$ или очередное произведение $e_m(n)$ на $\mu \cdot x'_{mkj}(n-i)$. Через мультиплексор 3 на другой вход каждого из сумматоров при фильтрации поступает накопленная предыдущими ЦОС-ячейками сумма произведений $x_j(n-i)$ на $w_{kj}^i(n)$ или на s_{mk}^i ; при вычислении коэффициентов АФ – сумма произведений $e_m(n)$ на $\mu \cdot x'_{mkj}(n-i)$, накопленная аккумулятором ЦОС-ячейки на предыдущих шагах вычисления.

Операции в каждом интервале дискретизации выполняются в следующем порядке. С приходом очередного набора $x_j(n)$ и $e_m(n)$ мультиплексором IMUX и сдвиговым регистром ISRГ формируется последовательность данных (рис. 4), поступающих на ЦОС-ячейки и элементы задержки структуры.



Рис. 4. Последовательность поступления данных на контроллер (первым поступает $\mu \cdot x'_{111}(n-1)$)

В сформированной последовательности первыми стоят $M \cdot K \cdot J$ значений $\mu \cdot x'_{mkj}(n-1)$, рассчитанных на предыдущем шаге дискретизации, а затем – поступившие от опорных микрофонов J значений $x_j(n)$. Это позволяет сформировать всю последовательность обрабатываемых данных, не дожидаясь расчета нового набора $\mu \cdot x'_{mkj}(n)$. Таким образом, в начале каждого интервала дискретизации рассчитываются и перезаписываются новые коэффициенты АФ, затем выполняется фильтрация опорных сигналов и получение набора $Y_k(n)$ и $\mu \cdot x'_{mkj}(n)$. При этом значения $\mu \cdot x'_{mkj}(n-1)$ используются по одному разу; значения $x_j(n)$ используются по K раз при расчете $y_{kj}(n)$ и по $M \cdot K$ раз при расчете $x'_{mkj}(n)$.

На расчет очередного набора значений $y_{kj}(n)$ требуется $K \cdot J$ тактов; на расчет набора значений $x'_{mkj}(n)$ – $M \cdot K \cdot J$ тактов; на перерасчет коэффициентов АФ – $K \cdot J \cdot (M+1)$ тактов. Итого, цикл вычислений требует $2 \cdot K \cdot J \cdot (M+1)$ тактов, а тактовая частота c структуры, реализуемой на ПЛИС, должна составлять не менее:

$$c = 2 \cdot K \cdot J \cdot (M+1) \cdot F_s.$$

Максимальная тактовая частота c_{max} ЦОС-ячеек FPGA – около 600 МГц [9]; F_s для случайного низкочастотного шума – единицы кГц [1,3], что создает достаточный запас по быстродействию для типичных значений M , K и J . Однако для реализации многоканального FXLMS-алгоритма к регулярной систолической структуре, использующей только ЦОС-ячейки и работающей с частотой дискретизации 600 МГц, добавляются мультиплексор, регистр сдвига и элементы задержки. Ресурсы для реализации этих устройств у ЦОС-ячеек отсутствуют, поэтому должна быть использована программируемая логика. Это обстоятельство и само по себе может снизить c_{max} ; кроме того, c_{max} начинает зависеть от M , K , J . Таким образом, первому из сформулированных выше критериев единая структура многоканального контроллера не соответствует.

Данная структура имеет и другие недостатки. Во-первых, ЦОС-ячейки не имеют достаточного для многоканальной адаптивной фильтрации количества мультиплексоров. Поэтому приходится компоновать $w_{kj}^i(n)$ и s_{mk}^i в общие блоки памяти (рис. 3), а $x_j(n-i)$ и $\mu \cdot x'_{mkj}(n-i-1)$ в общий поток (рис. 4). Это приводит к усложнению схемы управления структурой, т.к. каждый отсчет $x_j(n-i)$ требуется умножить на K коэффициентов АФ $w_{kj}^i(n)$, а затем – на $M \cdot K$ коэффициентов фильтров-оценок, в общем случае расположенных в блоке памяти не сразу вслед за $w_{kj}^i(n)$. Кроме того, поскольку $\mu \cdot x'_{mkj}(n-i-1)$ используются в каждом интервале дискретизации по одному разу, а $x_j(n-i)$ – по $K(M+1)$ раз, сдвиг в элементах задержки должен выполняться с различной частотой. Во-вторых, если N и/или L неодинаковы от канала к каналу, количество ЦОС-ячеек в структуре соответствует максимальному порядку фильтра. При этом появляется непроизводительный расход памяти (хранение коэффициентов фильтров, равных нулю) или дополнительное усложнение схемы управления, позволяющее выборочно заполнять элементы задержки. Съём результатов фильтрации во втором случае будет осуществляться в нескольких точках.

Возможности реализации контроллера как массива ЦОС-ячеек

ЦОС-ячейки, независимо друг от друга последовательно выполняющие умножение с накоплением, могут быть организованы в массив по-разному. Наиболее удачный в смысле соответствия сформулированным выше критериям вариант представлен на рис. 5.

Массив состоит из $K \cdot J$ вычислительных блоков, содержащих по две ЦОС-ячейки. К первой ячейке каждого из блоков относятся устройства умножения с накоплением MAC1 и мультиплексоры (рис. 5); ко второй – устройства умножения с накоплением MAC2. Режимы работы каждой из ЦОС-ячеек показаны на рис. 6. В состав каждого k_j -го блока входят также ОЗУ для хранения коэффициентов k_j -го АФ (W_{kj_RAM}) и последних N результатов фильтрации опорных сигналов (X'_{mki_RAM}).

К каждой j -й «строке» массива подводится вектор $X_j(n)$ от блоков памяти X_j_RAM ; каждый k -й «столбец» формирует выходной сигнал Y_k и использует при предварительной фильтрации опорных сигналов одни и те же наборы коэффициентов S_{mk} , содержащиеся в блоках памяти S_k_RAM . Отсчеты сигналов ошибки хранятся в отдельном блоке памяти e_m_RAM . Организация блоков памяти представлена на рис. 7.

Перед умножителями, после умножителей и после сумматоров ЦОС-ячеек с целью повышения c_{max} подключены регистры (рис. 6). MAC1 на фильтрацию векторов $X_j(n)$ коэффициентами АФ требуется $N+3$ такта, плюс $J-1$ тактов «столбцу» MAC1 требуется на вычисление Y_k . На предварительную фильтрацию $X_j(n)$ требуется $M \cdot (L+3)$ тактов. MAC2 выполняет пересчет коэффициентов АФ в течение $N(M+4)$ тактов.

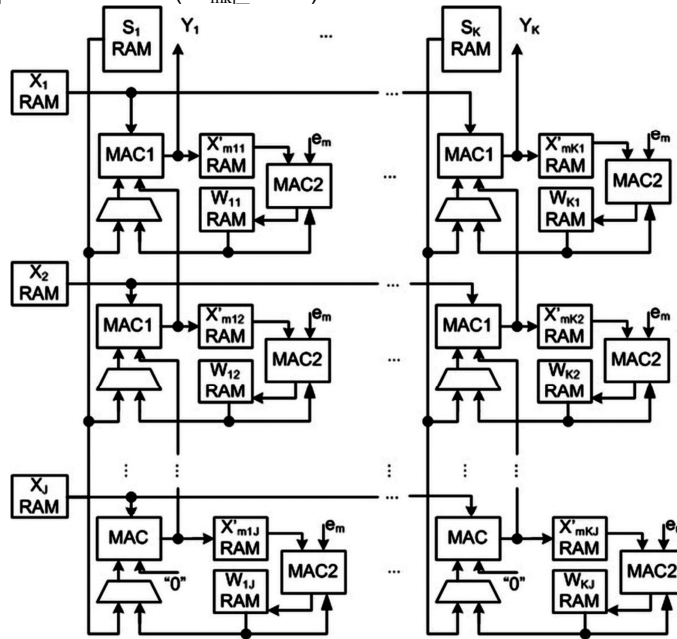


Рис. 5. Структура контроллера системы АГШ как массива независимых блоков

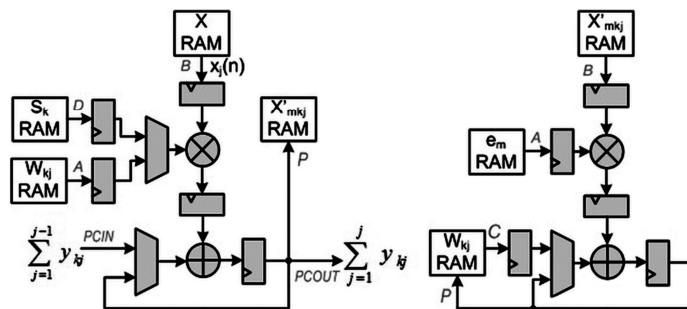


Рис. 6. Режимы работы ЦОС-ячеек

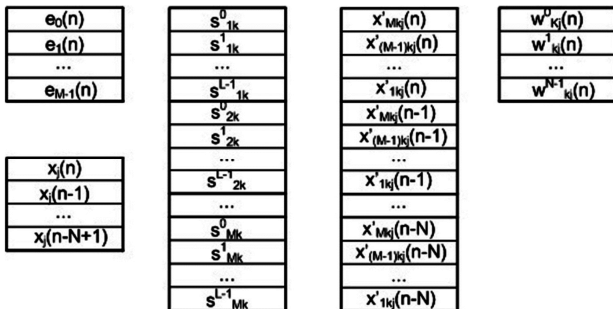


Рис. 7. Организация блоков памяти контроллера

Параллельную работу MAC1 и MAC2 целесообразно организовать во время вычисления наборов $x'_{mkj}(n)$ и векторов коэффициентов $w'_{kj}(n)$ соответственно. Это возможно, если каждый интервал дискретизации начинается с данных операций, и вычисленные в текущем такте наборы $x'_{mkj}(n)$ при расчете $w'_{kj}(n)$ не используются. Вычисленные $w'_{kj}(n)$ используются АФ в этом же интервале дискретизации. Поскольку $M \cdot (L+3)$ меньше $N \cdot (M+5)$, все вычисления займут приблизительно $N \cdot (M+6)$ тактов.

Основные характеристики ресурсоемкости и быстро-

действия двух рассмотренных вариантов реализации контроллера системы АПШ приведены в табл. 1. Сравнение данных вариантов позволяет заключить, что массив независимых блоков в большей степени соответствует сформулированному выше критерию. Максимальная тактовая частота массива независимых блоков менее зависима от параметров системы АПШ, т.к. увеличение K и J не вносит изменений в блоки, увеличивая только их количество, а с увеличением M, N, L увеличивается только требуемый объем памяти W_{ki_RAM} и X'_{mki_RAM} . Вся структура реализуется на специализированных ресурсах FPGA с фиксированными скоростными характеристиками – ЦОС-ячейках и блочном ОЗУ. Максимальная тактовая частота единой структуры более зависима от J, K, M, N из-за потребности в мультиплекторе и сдвиговых регистрах на $J(KM+1)$ ячеек, реализуемых на программируемой логике. Кроме того, массив независимых блоков требует меньшего количества ЦОС-ячеек, и при этом сохраняет достаточный запас по производительности.

В табл. 2 приведены результаты исследования зависимости c_{max} отдельного вычислительного блока от N при $M=2$, полученные при использовании временного анализатора среды WebPACK ISE после реализации проекта. Снижение c_{max} при $N=512$ и $N=1024$ объясняется тем, что для реализации блоков памяти X'_{mki_RAM} и W_{ki_RAM} одного блочного ОЗУ объемом 18 кбит [9] становится недостаточно.

Таблица 1. Характеристики структур контроллера многоканальной системы АПШ

	Кол-во ЦОС-ячеек	Объем памяти (хранится 5 разрядов $\mu X'$)	Кол-во тактов на цикл вычислений
Единая	N	$16N(KJ+MK+J)+5NJMK$	$2KJ(M+1)$
Массив независимых блоков	$2KJ$	$16N(KJ+MK+J)+5(N+1)JMK$	$N(M+6)$

Таблица 2. Зависимость c_{max} вычислительного блока от N

N	16	32	64	128	256	512	1024
c_{max} , МГц	447	447	447	447	447	411	411

Реализация на FPGA контроллера в виде массива независимых блоков

Входы и выходы ЦОС-ячеек делятся на внутренние, т.е. доступные только другим ЦОС-ячейкам при их объединении в цепочку, и внешние, т.е. доступные программируемой логике [9]. На умножитель ЦОС-ячейки могут одновременно поступать два из трех внешних входных сигналов ЦОС-ячейки – B и A или B и D ; непосредственно на вход сумматора может поступать внешний входной сигнал C (рис. 6). В MAC1 на входы B подаются отсчеты $X_j(n)$; на входы A и D – коэффициенты АФ и фильтров-оценок. Внешний выход P MAC1 используется для со-

единения с блоком памяти X'_{mki_RAM} . Внутренние вход и выход PCIN и PCOUT используются для приема и передачи сумм y_{kj} . В MAC2 на входы B подаются значения $\mu \cdot x'_{mkj}$, на входы A – отсчеты сигналов ошибки. На вход C подаются коэффициенты АФ (4). Новые значения коэффициентов АФ поступают на блоки памяти W_{ki_RAM} от внешних выходов P MAC2.

Контроллер содержит J блоков ОЗУ X_i_RAM по N ячеек; $K \cdot J$ блоков W_{ki_RAM} по N ячеек; K блоков S_k_RAM по $M \cdot L$ ячеек; $K \cdot J$ блоков X'_{ki_RAM} по $M(N+1)$ ячеек; блок ОЗУ на M ячеек для хранения сигналов ошибки. Блоки памяти X_i_RAM и X'_{ki_RAM} работают по принципу циклических буферов. W_{ki_RAM} и X'_{ki_RAM} являются двухпортовыми ОЗУ.

Управление массивом блоков осуществляется автоматом, состоящим из двух частей с собственными наборами состояний. Первая часть управляет работой MAC1, чтением S_k_RAM , чтением и записью X_i_RAM , чтением W_{ki_RAM} и записью X'_{ki_RAM} . Вторая часть управляет работой MAC2, чтением и записью e_m_RAM , чтением X'_{ki_RAM} , чтением и записью W_{ki_RAM} .

Заключение

С учетом основных характеристик рассмотренных структур можно заключить, что предпочтение следует оказывать структуре контроллера в виде массива независимых блоков.

Литература

- Hansen C.H., et al. Active control of noise and vibration. – 2nd ed. CRC Press, 2012.
- Kuo S.M., Morgan D.R. Active noise control systems: algorithms and DSP implementations. John Wiley & Sons, Inc., 1995.
- Snyder S.D. Active noise control primer. Springer, 2000.
- Elliott S. Signal processing for active control. Academic Press, 2000.
- Уидроу Б., Стирнз С. Адаптивная обработка сигналов: Пер. с англ. - М.: Радио и связь, 1989. - 440с.
- Morgan, D.R., and D.A. Quinlan. «Local silencing of room acoustic noise using broadband active noise control». Applications of Signal Processing to Audio and Acoustics, 1993. Final Program and Paper Summaries., 1993 IEEE Workshop on. IEEE, 1993.
- Roure A. «Self-adaptive broadband active sound control system.» Journal of Sound and Vibration 101.3 (1985): 429-441.
- <http://www.ti.com/lstds/ti/processors/dsp/overview.page> (27.08.2015)
- http://www.xilinx.com/support/documentation/data_sheets/ds180_7Series_Overview.pdf (27.08.2015)
- Hawkes, G.C. «DSP: Designing for Optimal Results. High-Performance DSP Using Virtex-4 FPGAs». (2005).
- Mayer-Baese U. Digital Signal Processing with field programmable Gate Arrays. Springer, 2007.

УДК 004.383.3; 004.272.44

КОНФИГУРИРУЕМЫЙ ВЫЧИСЛИТЕЛЬ НА БАЗЕ FPGA ДЛЯ ПОТОКОВОЙ ОБРАБОТКИ ВИДЕОСИГНАЛОВ

*Сизов М.М., инженер-программист Института автоматики и электрометрии СО РАН,
e-mail: sizov.m.m@gmail.com;*

*Зюбин В.Е., д.т.н., ведущий научный сотрудник Института автоматики и электрометрии СО РАН,
e-mail: zyubin@iae.nsk.su.*

CONFIGURABLE FPGA ARCHITECTURE FOR VIDEO STREAM PROCESSING

Sizov M.M., Zyubin V.E.

In this paper we present an analysis of current FPGA development approaches, from that analysis requirements for a video stream processing algorithms development system are deduced. We proposed an approach for implementing of such system based on hybrid FPGA + ARM SoC and a configurable pipelined FPGA architecture for video stream processing. System's implementation consists of predefined algorithm blocks, and a Tcl-based IDE extension which transforms high-level configuration file to a complete VHDL project and automates building and flashing process. The system was designed with the goal of simplifying a development of high-performance stream processing applications, but in the same time system reduces set of problems which can be easily solve. This drawback could be mitigated as the system is extensible and uses standard AXI4-Stream protocol.

Key words: FPGA, analysis requirements, video stream, VHDL, ARM SoC.

Ключевые слова: гибридные архитектуры, конвейерная обработка видеосигнала, ПЛИС, FPGA, VHDL.

Введение

Цифровые системы обработки видеосигналов – одно из важнейших направлений в информационных технологиях, связанных с автоматизацией технологических процессов, бесконтактными измерениями, контролем качества и др. Источник сигнала в таких системах – цифровая камера, формирующая поток видеок кадров в сотни мегабайт в секунду. Для обработки таких потоков данных используются подходы, предполагающие параллелизм и конвейеризацию [1]. Особый интерес у исследователей вызывают алгоритмы на основе конвейера, поскольку позволяют начать обработку информации, не дожидаясь конца кадра, «на лету». Для реализации таких алгоритмов используются программируемые вентильные матрицы (Field Programmable Gate Array, или FPGA) [2], которые в дополнение привлекательны своей компактностью, низким энергопотреблением и возможностью контроля периферийного оборудования.

При всех достоинствах разработка систем на базе FPGA трудоемка. При программировании используются специализированные языки, так называемые языки описания аппаратуры (VHDL, Verilog), программы пишутся в терминах триггеров, регистров и их соединений; а существующие ограничения на объем памяти, находящийся на FPGA, и сложность реализации пользовательского интерфейса, как правило, приводят к необходимости взаимодействия с процессором общего назначения.

Заметно снижает трудоемкость разработки использование гибридных архитектур, совмещающих на одном кристалле FPGA и ARM-процессор. В существующих на

Анализируются существующие решения по разработке вычислителей на базе FPGA для потоковой обработки видеосигналов с учетом специфики рассматриваемой области, формулируются требования к системе создания алгоритмов потоковой обработки видео, предлагается архитектура конфигурируемого конвейерного вычислителя, ориентированного на обработку видеосигнала, и описывается реализация предложенного подхода, ориентированного на аппаратную платформу Zynq.

данный момент гибридных архитектурах FPGA связаны с процессором через шину данных AMBA AXI [3]. Ведущие производители FPGA компании Xilinx и Altera предлагают семейства гибридных кристаллов Zynq [4] и Arria V [5], соответственно. Эти кристаллы используют сторонние производители. Так плата Zedboard (Digilent) на основе кристалла Zynq имеет стандартный для ПК набор периферии и FMC-коннектор (FPGA mezzanine card) с возможностью подключения высокоскоростной видеокамеры [6]. Программирование поддерживается специализированным IDE, включающим средства трансляции VHDL/Verilog, генерации файла-прошивки, загрузки, эмуляции. Платформа обеспечивает возможность проведения как исследовательских работ, так и создание практических приложений в области цифровых систем обработки видеосигналов.

Хотя производители облегчают реализацию взаимодействия с внешним ОЗУ, предоставляя программные драйверы и IP-ядра с реализацией механизма DMA [7], сложность программирования взаимодействия процессора и FPGA остается высокой. Это стимулирует исследователей искать новые подходы, и в первую очередь предметно-ориентированные, снижающие трудоемкость программирования FPGA.

В статье анализируются существующие решения для разработки на FPGA с учетом специфики рассматриваемой области, формулируются требования к системе

создания алгоритмов потоковой обработки видео, предлагается архитектура конфигурируемого конвейерного вычислителя, ориентированного на обработку видеосигнала, и описывается реализация предложенного подхода, ориентированного на аппаратную платформу Zynq.

Высокоуровневые подходы к разработке алгоритмов на FPGA

LabVIEW и Matlab. Общая тенденция развития инструментов для программирования FPGA – увеличение уровня абстракции понятий, которыми оперирует программист. В качестве примеров можно привести программные пакеты LabVIEW [8] и Matlab [9], которые в дополнение к средствам программирования процессоров общего назначения имеют расширения, предназначенные для программирования FPGA штатными средствами (языки G, Simulink, Matlab). Недостатки использования этих средств – отсутствие возможностей профилирования и анализа генерируемого VHDL-кода, необходимость приобретать кроме специализированных расширений еще и базовые пакеты, содержащие с точки зрения программирования FPGA заведомо избыточную функциональность, за которую приходится платить. Что касается LabVIEW, то это решение еще и ориентировано на оборудование National Instruments. Matlab позволяет генерировать код на языках аппаратуры из программ на языках Simulink (конечные автоматы и диаграммы потока данных) и Matlab (матрицы). Matlab позволяет использовать оборудование основных производителей FPGA (Altera, Xilinx), в принципе допускает работу с Zynq, но не поддерживает потоковый интерфейс передачи данных (AXI-Stream), что снижает общую пропускную способность создаваемых систем.

Языки управления потоком исполнения. Существующие решения Handel-C [10], Impulse-C [11], Vivado High-Level Synthesis [12] используют Си-подобный синтаксис для создания алгоритма и последующей трансляции в языки описания аппаратуры. К недостаткам подхода относят разницу в парадигмах: язык Си относится к языкам управления потоком исполнения, поэтому эффективное использование возможностей FPGA предполагает наличие у программиста знаний архитектуры FPGA и преобразований, выполняемых транслятором на этапе кодогенерации.

Dataflow подход. Dataflow-парадигма более привлекательна для программирования FPGA. Программа описывается сетью взаимодействующих функциональных модулей (Actor) через однонаправленные каналы связи, данные представлены токенами. По каналам связи передаются также управляющие токены, например, токен начала кадра изображения или токен начала строки. В классическом варианте модули ожидают данных на всех входных каналах связи, проводят вычисления и после этого передают результаты на выход модуля. Однако для увеличения производительности системы используется конвейерная обработка данных.

Примером использования Dataflow-парадигмы с конвейерной архитектурой является язык Caph [13], он позволяет описывать функциональные модули и описывать их связи,

затем транслирует внутреннее представление в HDL-код.

FPGA обладают тремя уровнями памяти (триггеры 10-100 Кб, SRAM-блоки 100-1000 Кб, внешняя DDR-память, к которой имеется доступ либо через пины общего назначения, либо через механизмы DMA), при написании программы на языке Caph нет явных механизмов указания типа используемой памяти, что приводит к случаям неэффективного распределения памяти и сужению класса решаемых задач при обработке видео, также не решается проблема взаимодействия с процессором общего назначения. Caph использует функциональную парадигму программирования, что также увеличивает требования к квалификации программиста.

В результате проведенного анализа были сформулированы требования к системе разработки алгоритмов потоковой обработки видеосигналов на FPGA:

1. Использование системы не должно требовать специализированных знаний об устройстве FPGA и языков описания аппаратуры.

2. Должна быть предусмотрена возможность описания взаимодействия с процессором общего назначения.

3. Результирующий код должен иметь формат, допускающий визуальный анализ, конкретно VHDL.

Конфигурируемый вычислитель

Для решения проблем взаимодействия с процессором общего назначения предлагается использовать гибридные архитектуры.

Алгоритм обработки потокового видео предлагается описывать в виде конфигурируемого вычислителя – последовательно соединенных библиотечных блоков (алгоритмических примитивов), образующих тракт обработки данных (рис. 1).



Рис. 1: Схема тракта обработки данных

Соединение блоков однонаправленно и осуществляется по стандартному протоколу AXI-Stream [14], ориентированному на потоковую обработку сигналов. При этом однозначно определены ядро-передатчик данных (master) и ядро-приемник (slave).

Начальный источник данных для тракта – IP-ядро (от Intellectual Property) видеочасти, расположенный в FPGA, или процессор общего назначения (использующий DMA). По завершению IP-ядро DMA сохраняет результат обработки в ОЗУ процессора.

В качестве алгоритмических примитивов выступают: фильтр свертки (размытие, резкость), гамма-корректор, медианный фильтр, конечный автомат.

Конечный автомат задается наборами состояний и переменных, а также правилами перехода между состояниями. Состояние представляет собой действия, выполняемые при получении новой порции данных: арифметические и логические операции с переменными, проверка значений переменных, конца строки и конца кадра.

Переменные конечного автомата доступны со стороны процессора общего назначения по протоколу AXI-Lite и могут хранить как дополнительную информацию, так и флаги, например, флаг необходимости сохранения кадра на жестком диске.

Реализация конфигурируемого вычислителя

Предложенный подход был реализован для платы Zedboard с гибридным кристаллом Zynq 7000 [4] и CMOS камеры VITA-2000. Плата имеет 512 Мб ОЗУ, 10/100/1000 Ethernet, VGA, HDMI, USB-JTAG, USB-UART, Audio I/O, PMOD-интерфейс, слот SD-карты. Взаимодействие платы и видеокamеры организовано через интерфейс FMC (FPGA Mezzanine Card).

FPGA часть кристалла Zynq обеспечивает взаимодействие с периферией и позволяет реализовывать пользовательские алгоритмы, в том числе, ресурсоемкие вычисления. ARM-процессор выполняет функции процессора общего назначения. С платой поставляется среда разработки Vivado (Xilinx) и драйверы для двух операционных систем – Linux и HiKerネル. Среда разработки Vivado обеспечивает возможность создания пользовательских программ на языках VHDL, Verilog, Си, средства отладки и симуляции. Ядро системы Vivado написано на языке Tcl и допускает консольное управление проектом на этом языке.

Система создания алгоритмов обработки потокового видео включает программу «генерации проекта и загрузки в целевую плату» (Tcl), шаблон интерфейсной программы для процессора общего назначения (Си) и библиотеку алгоритмических примитивов (VHDL). Библиотека алгоритмических примитивов подключена к системе Vivado как пользовательский репозиторий. Алгоритмические примитивы реализуют протокол передачи видеопотока поверх стандартного интерфейса AXI (AXI4-Stream Video IP [15]), каждый примитив взаимно-однозначно связан с одноименной Tcl-функцией. Алгоритмические примитивы разрабатываются, моделируются и отлаживаются с помощью стандартных инструментов, входящих в поставку IDE Vivado.

Создаваемые алгоритмические примитивы должны поддерживать протокол AXI4-Stream Video IP [15] и заданный формат видеопотока (разрешение кадра, количество байтов на пиксель). Список доступных алгоритмических примитивов указывается в Tcl-скрипте.

Создание алгоритма обработки потокового видео ведется на отдельном ПК и состоит из следующих шагов:

1. В текстовом редакторе задается вычислительный тракт – последовательность алгоритмических примитивов в виде списка параметризованных Tcl-функций и описания конечного автомата на языке VHDL с использованием AXI-Stream шаблона.
2. Запускается программа «генерации проекта и загрузки в целевую плату», которая через консольное управление IDE Vivado:
 - создает проект и блок-схему (Block Design) для FPGA, задает используемые IP-ядра DMA и видеокamеры;
 - подключает IP-ядра алгоритмических примитивов в блок-схему проекта;
 - запускает компилятор с автоматической трассировкой и прошивает полученный bitstream-файл в FPGA;
 - создает код для HiKerネル по шаблону, после чего производит его компиляцию и загрузку в ОЗУ ARM-процессора.

Заключение

В работе был предложен подход и реализована система создания алгоритмов обработки потокового видео в виде конфигурируемого конвейерного вычислителя.

Конвейерный вычислитель описывается как последовательность соединенных алгоритмических примитивов. Созданное на языке Tcl программное расширение IDE Vivado автоматически преобразует описание вычислителя в исполняемый код для платформы Zedboard (гибридный кристалл Zynq компании Xilinx).

К ограничениям предлагаемого метода можно отнести ориентированность на архитектуру Zynq, существование задач,

использующих функции, не входящие в библиотеку алгоритмических примитивов, и автоматическую трассировку IDE Vivado, которая, может привести к рассинхронизации сигналов и невозможности размещения проекта на заданном кристалле. В этих случаях хотя и наблюдается сокращение времени выполнения проекта, возникает необходимость либо в расширении библиотеки, либо в ручной трассировке, и, как следствие, необходимость использовать высококвалифицированных программистов. В случае же, когда ограничения не проявляются, предлагаемый подход снижает и квалификационные требования к программисту.

Подход также может быть применим в случае гибридных кристаллов компании Altera, предоставляющей IDE Quartus с консольным Tcl-интерфейсом.

В будущем планируется расширить библиотеку алгоритмических примитивов, добавить в систему средства, упрощающие генерацию конечных автоматов, и ввести возможность графического описания вычислителя.

Литература

1. Аминев Д.А., Фокин А.Н. Методы селекции кадрового синхроимпульса для ввода несжатого видеопотока от однонаправленной одноразрядной цифровой линии и их реализация на ПЛИС // Цифровая обработка сигналов. 2014. №1. С. 52-55.
2. Панкратов В.Г., Карих А.А., Панфилов В.Н., Гуров А.Д. Вычисление функции неопределенности для пассивной локации на ПЛИС и графическом процессоре // Цифровая обработка сигналов. 2014. №1. С. 56-65.
3. AXI AMBA [Электронный ресурс] // Ссылка: <http://www.arm.com/products/system-ip/amba-specifications.php>
4. Xilinx Zynq [Электронный ресурс] // Ссылка: <http://www.xilinx.com/products/silicon-devices/soc/zynq-7000.html>
5. Altera Aria V [Электронный ресурс] // Ссылка: <http://www.altera.com/devices/fpga/aria-fpgas/aria-v/hard-processor-system/armv-sochps.html>
6. Crockett L. H. et al. The Zynq Book: Embedded Processing with the Arm Cortex-A9 on the Xilinx Zynq-7000 All Programmable Soc. – 2014.
7. Лысаков К.Ф., Шадрин М.Ю. Особенности применения аппаратных устройств на базе FPGA для задач потоковой обработки изображений // Вестник Новосибирского государственного университета. Серия: Информационные технологии. – 2009. – Т. 7. – №. 3. – С. 15-22.
8. FPGA [Электронный ресурс] // Ссылка: <http://www.ni.com/labview/fpga>.
9. Matlab FPGA design [Электронный ресурс] // Ссылка: <http://www.mathworks.com/solutions/fpga-design/>
10. Loo S.M. et al. Handel-C for rapid prototyping of VLSI coprocessors for real time systems // System Theory, 2002. Proceedings of the Thirty-Fourth Southeastern Symposium on. – IEEE, 2002. – С. 6-10.
11. Antola A. et al. A novel hardware/software codesign methodology based on dynamic reconfiguration with Impulse C and CoDeveloper // Programmable Logic, 2007. SPL'07. 2007 3rd Southern Conference on. – IEEE, 2007. – С. 221-224.
12. Vivado High Level Synthesis [Электронный ресурс] // Ссылка: <http://www.xilinx.com/products/design-tools/vivado/integration/esl-design.html>
13. Serot J., Berry F., Ahmed S. Implementing stream-processing applications on FPGAs: a DSL-based approach // Field Programmable Logic and Applications (FPL), 2011 International Conference on. – IEEE, 2011. – С. 130-137.
14. AMBA AXI4-Stream Protocol Specification v1.0 [Электронный ресурс] // Ссылка: <http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ih0051a/index.html>.
15. AXI4-Stream Video IP and System Design Guide [Электронный ресурс] // Ссылка: http://www.xilinx.com/sup-port/documentation/ip_documentation/axi_videoip/v1_0/ug934_axi_videoip.pdf.

МНОГОПРОЦЕССОРНЫЕ КЛАСТЕРЫ НА ОСНОВЕ СИГНАЛЬНЫХ ПРОЦЕССОРОВ СО СТАТИЧЕСКОЙ СУПЕРСКАЛЯРНОЙ АРХИТЕКТУРОЙ*

Букварев Е.А., научный сотрудник кафедры информационных радиосистем Нижегородского государственного технического университета им. П. Е. Алексеева, e-mail: bukwarev@ Rambler.ru;

Кузин А.А., доцент кафедры информационных радиосистем Нижегородского государственного технического университета им. П. Е. Алексеева, e-mail: kuzin_alex@nntu.nnov.ru;

Приблудова Е.Н., кандидат технических наук, доцент кафедры информационных радиосистем Нижегородского государственного технического университета им. П.Е. Алексеева, e-mail: pribludova@nntu.nnov.ru;

Рындык А.Г., доктор технических наук, профессор, заведующий кафедрой информационных радиосистем Нижегородского государственного технического университета им. П. Е. Алексеева, e-mail: a_ryndyk@nntu.nnov.ru.

MULTIPROCESSOR CLUSTERS BASED ON SIGNAL PROCESSORS WITH STATIC SUPER-SCALAR ARCHITECTURE

Bukvarev E.A., Kuzin A.A., Pribludova E.N., Ryndyk A.G.

In this paper two structures of multiprocessor clusters on basis of signal processors with static super-scalar architecture are presented. Some design features are considered, configuration and appearance of the multiprocessor cluster are shown and characteristics of cluster are provided.

The developed multiprocessor clusters are intended for use into integrated module in stationary and mobile systems of high-performance digital signal processing.

Key words: digital processing, signal processor, multiprocessor clusters, integrated module.

Ключевые слова: цифровая обработка, сигнальный процессор, многопроцессорные кластеры, интегрированный модуль.

Введение

В общем случае, вычислительный кластер – это набор вычислительных узлов, объединенных некоторой коммуникационной сетью. Каждый вычислительный узел имеет свою оперативную память и работает под управлением своей операционной системы. Наиболее распространенным является использование однородных кластеров, где все узлы абсолютно одинаковы по своей архитектуре и производительности. Особенностью кластера является применение компонентов серийного выпуска.

Для задач цифровой обработки сигналов представляется перспективным использование в качестве узлов кластера специализированных высокопроизводительных процессоров (ВПЦОС). Как правило, DSP-процессоры имеют встроенную память, а коммуникации между процессорами внутри кластера обеспечиваются встроенными средствами процессоров.

Постановка задачи

Разработка мультипроцессорного кластера на основе DSP-процессоров является частной задачей проектирования интегрированного модуля ЦОС с возможностью

Представлены две структуры многопроцессорных кластеров на основе сигнальных процессоров со статической суперскалярной архитектурой отечественной разработки. Рассмотрены некоторые конструктивные особенности, показаны компоновка и внешний вид многопроцессорного кластера, приведены характеристики кластера.

масштабирования производительности.

Выбранный вариант компоновки интегрированного модуля (ИМ) в виде базовой (несущей платы) с устанавливаемыми на ней submodule (мезонинами) допускает установку от одного до пяти мультипроцессорных кластеров, что обеспечивает изменение производительности в широких пределах.

Submodule (СМ) должен представлять собой функционально и конструктивно законченный четырехпроцессорный кластер. Кроме того, СМ должен иметь специальное распределение сигналов высокоскоростных LINK-портов по разъемам с целью упрощения трассировки базовой платы.

Варианты структуры многопроцессорного кластера

Структуры мультипроцессорного кластера базируются на двух свойствах [1, 2] применяемого процессора: мультипроцессорном интерфейсе и скоростных LVDS портах обмена данными (в дальнейшем – LINK-портах). Четырехпроцессорный кластер спроектирован в соответствии с рекомендациями [2] и представляет собой мультипроцессорную систему с разделяемой общей шиной.

Первый вариант кластера показан на рис. 1 и состоит из:
– четырех процессоров ВПЦОС0 – ВПЦОС3;
– блока синхронной динамической памяти SDRAM с организацией 32М x 64;

* Работа выполнена при поддержке Министерства образования и науки РФ в рамках договора № 02.G25.31.0061 от «12» февраля 2013 года (в соответствии с Постановлением Правительства Российской Федерации от 9 апреля 2010 г. № 218).

- загрузочного постоянного запоминающего устройства FLASH объемом 16Мбайт;
- тактового генератора с частотой 100МГц;
- тактового буфера ТБ;
- термодатчиков;
- источника вторичного питания ИВП.

Процессоры, динамическая и постоянная память подключены к системной шине кластера с разрядностью данных 64 бита. Адресные пространства всех процессоров и внешней памяти образуют единое 32-разрядное адресное пространство [1, 2].

Тактовый сигнал частотой 100 МГц (системная частота SCLK), формируемый генератором, усиливается ТБ и распределяется по ВПЦОС и SDRAM.

Термодатчики процессоров образуют подсистему термомониторинга, доступную по интерфейсу SPI через внешний разъем 3.

Отладочный интерфейс JTAG (JTAG-цепочка) охватывает последовательно все процессоры кластера и через буферную схему подключается к внешнему разъему 3.

Источники ИВП формируют напряжение питания ядра (1,0 В) и внешних контактов (2,5 В) ВПЦОС.

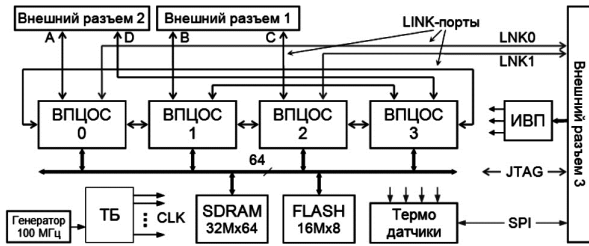


Рис. 1. Первый вариант структуры многопроцессорного кластера

Через системную шину организован мультипроцессорный интерфейс (МПИ), который имеет встроенный арбитраж и позволяет объединять на общей шине до четырех процессоров без дополнительных логических схем. Кроме того, через МПИ любой процессор имеет доступ к ресурсам других процессоров, включая внутреннюю память и системные регистры [2]. Специальный широковещательный (Broadcast) режим МПИ позволяет выполнять запись с системной шины во внутреннюю память одновременно во все ВПЦОС кластера.

Распределение адресного пространства мультипроцессорной памяти кластера детально приведено в [1, 2], в пространстве внешней памяти используется только зона адресов 0x40000000 – 0x44000000, соответствующая строке MSSD0, к которому подключена общая динамическая память.

Полнодуплексные LINK-порты обеспечивают межпроцессорные соединения типа точка-точка. Соединения между ВПЦОС

и соединения с внешними разъемами показаны в табл. 1.

Таблица 1. Соединения LINK-портов

Полнодуплексные соединения		Разрядность (бит)
ВПЦОС0 LINK0	Разъем 2 ("А")	4
ВПЦОС0 LINK1	ВПЦОС3 LINK2	4
ВПЦОС0 LINK2	Разъем 3 (LNK0)	1
ВПЦОС0 LINK3	ВПЦОС1 LINK0	4
ВПЦОС1 LINK1	Разъем 1 ("В")	4
ВПЦОС1 LINK2	ВПЦОС3 LINK1	4
ВПЦОС1 LINK3	ВПЦОС2 LINK0	4
ВПЦОС2 LINK1	Разъем 3 (LNK1)	1
ВПЦОС2 LINK2	Разъем 1 ("С")	4
ВПЦОС2 LINK3	ВПЦОС3 LINK0	4
ВПЦОС3 LINK3	Разъем 2 ("D")	4

Часть LINK-портов используется для соединения процессоров внутри кластера, что позволяет снизить интенсивность информационного обмена между процессорами по системной шине. По одному LINK-порту с каждого ВПЦОС идут выводы на внешние разъемы 1 и 2 (обозначены А, В, С и D). Данные порты позволяют организовать на базовой плате обмен с соседними кластерами. Режим работы перечисленных LINK-портов – четырехбитовый с тактовой частотой 400 МГц.

На внешний разъем 3 выведены два LINK-порта от ВПЦОС0 и ВПЦОС2 (обозначены LNK0 и LNK1), работающие в однобитовом режиме с тактовой частотой 100 МГц. Указанные порты подключаются к интерфейсной ПЛИС базовой платы, обеспечивая загрузку программ и обмен данными с целевым устройством.

Примененные в кластере ВПЦОС имеют двунаправленный четырехразрядный порт пользовательского ввода/вывода, обозначенный в [1] как FLAG3-0. В схеме кластера эти сигналы соединяются с входами запросов прерывания IRQ3-1, как показано в табл. 2.

Установка одного из сигналов FLAG3-0 приводит к прерыванию ВПЦОС, вход IRQ которого подключен к этому сигналу, что позволяет на аппаратном уровне синхронизировать выполнение программ на различных процессорах кластера. Так, например, аппаратная синхронизация используется встроенным тестом (BIST), выполняемым одновременно всеми процессорами кластера.

Используемые процессоры имеют четыре контакта (т.н. конфигурационный порт), которые определяют режим работы и поведение процессора после снятия сигнала сброса. Сигналы и их значения, установленные по умолчанию, перечислены в табл. 3. Как видно из табл. 3, после снятия сигнала сброса процессор ВПЦОС0 начинает загрузку программы из Flash-памяти, остальные – ожидают загрузку по LINK-порту.

Таблица 2. Связь сигналов IRQ и FLAG

Вход IRQ	Связан с выходом	Вход IRQ	Связан с выходом
ВПЦОС0 IRQ0	–	ВПЦОС2 IRQ0	–
ВПЦОС0 IRQ1	ВПЦОС3 FLAG1	ВПЦОС2 IRQ1	ВПЦОС0 FLAG2
ВПЦОС0 IRQ2	ВПЦОС1 FLAG1	ВПЦОС2 IRQ2	ВПЦОС1 FLAG2
ВПЦОС0 IRQ3	ВПЦОС2 FLAG1	ВПЦОС2 IRQ3	ВПЦОС3 FLAG3
ВПЦОС1 IRQ0	–	ВПЦОС3 IRQ0	–
ВПЦОС1 IRQ1	ВПЦОС0 FLAG1	ВПЦОС3 IRQ1	ВПЦОС0 FLAG3
ВПЦОС1 IRQ2	ВПЦОС3 FLAG2	ВПЦОС3 IRQ2	ВПЦОС1 FLAG3
ВПЦОС1 IRQ3	ВПЦОС2 FLAG2	ВПЦОС3 IRQ3	ВПЦОС2 FLAG3

Таблица 3. Конфигурация кластера по умолчанию

Сигнал	ВПЦОС				Инициализация
	0	1	2	3	
BMS	0	1	1	1	0 – загрузка из Flash, 1 – по LINK
BM	0	0	0	0	0 – прерывания IRQ запрещены (1 – разрешены)
TMR0E	1	1	1	1	1 – данные LINK-порта 4 бита (0 – LINK 1 бит)
BUSLOCK	1	1	1	1	1 – запись в SYSCON разрешена всегда (0 – однократная запись)

Далее программа, загруженная в ВПЦОС0, копируется по LINK-порту в ВПЦОС1, затем в ВПЦОС2 и ВПЦОС3. Запуск программы на ВПЦОС3 является окончанием инициализации кластера.

Возможна начальная инициализация ВПЦОС из Flash-памяти, для чего сигнал BMS устанавливается низким уровнем для всех процессоров субмодуля. Операции начальной загрузки полностью соответствуют приведенным в [3].

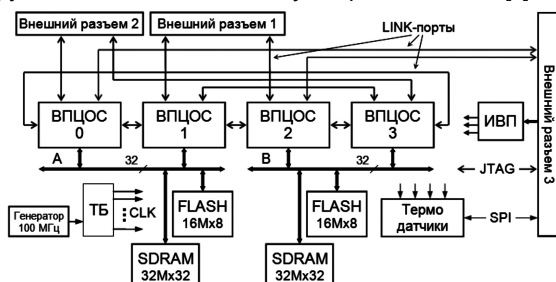


Рис. 2. Второй вариант структуры многопроцессорного кластера

При расчетах на начальных этапах проектирования было выявлено незначительное (от 2 до 3пФ) превышение номинальной емкостной нагрузки некоторых цепей системной шины: адреса A23-A1 и данных D7-D0. Это не только несколько увеличивает динамическую потребляемую мощность, но и изменяет временные характеристики сигналов, что может быть источником потенциальных сбоев в работе кластера, особенно на границах рабочего диапазона температур.

Исходя из вышесказанного, был разработан другой вариант субмодуля, структура которого показана на рис. 2. Общая системная шина разделена на две – А и В, попарно соединяющие ВПЦОС, в результате чего получилось два двухпроцессорных кластера. Разрядность данных уменьшена с 64 до 32, чтобы не увеличивать количество микросхем SDRAM и сохранить геометрические размеры субмодуля. К шине В подключена дополнительная Flash-память начальной загрузки. Анализ временных соотношений для сигналов системной шины в разработанных субмодулях, выполненный по методике [4], под-

Характеристики кластера (вариант 1)

Количество высокопроизводительных процессоров ЦОС	4
Тактовая частота ядра CCLK, МГц	400
Тактовая частота системной шины SCLK, МГц	100
Разрядность системной шины, бит	64
Пиковая пропускная способность системной шины, Мбайт	800
Объем динамической памяти (организация), Мбайт	256 (32M×64)
Тактовая частота Link- портов LCLK, МГц	400
Разрядность Link- портов, бит	4
Пиковая пропускная способность каждого Link- порта, МБайт/с	400
Внешнее питание	5В/1,9А; 3,3В/1А
Диапазон рабочих температур, °С	- 40 ... +65
Размеры, мм	80×150×7,7

твердил работоспособность представленных вариантов.

Конструкция и характеристики кластера

Вычислительные кластеры (субмодули) выполнены на 16-слойной печатной плате размерами 80×150мм с двусторонней установкой элементов. Расположение основных компонентов на плате варианта 1 показано на рис. 3, слева – верхняя сторона, справа – нижняя.

Для подключения субмодуля к базовой плате применены малогабаритные разъемы ST4/SS4 фирмы Samtec с высотой стыковки 4,5 мм и шагом между контактами 0,4 мм. Характеристики разъемов позволяют организовать межмодульную связь по LINK-портам с частотой до 400 МГц.

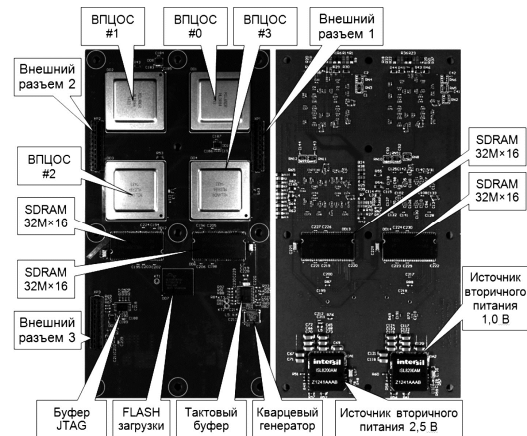


Рис. 3. Расположение компонентов на плате субмодуля

Заключение

Разработанные субмодули успешно прошли исследовательские испытания в составе 20-процессорного интегрированного модуля (5 кластеров), подтвердив все основные технические характеристики.

Четырехпроцессорные вычислительные кластеры, являющиеся основным функциональным узлом интегрированного модуля, могут быть использованы автономно – конструкция субмодулей допускает их непосредственное встраивание в аппаратуру потребителя.

Литература

1. Сигнальный процессор со статической суперскалярной архитектурой 1967ВЦ2Ф, К1967ВЦ2Ф, К1967ВЦ2ФК. Спецификация. URL: http://milandr.ru/uploads/Products/product_294/spec_1967VC2.pdf.
2. Мякочин Ю.О. 32-разрядный суперскалярный DSP-процессор с плавающей точкой // Компоненты и технологии. 2013. №7.
3. ADSP-TS20x TigerSHARC® Processor Boot Loader Kernels Operation (EE-200). Revision 1.0, March 2004. Ana-log Devices, Inc.
4. Кузин А.А., Плужников А.Д., Приблудова Е.Н., Сидоров С.Б. Анализ временных соотношений для сигналов в проектируемых цифровых модулях и оценка работоспособности. «Цифровая обработка сигналов» №2 2014г. – с. 70-77.

УДК 621.396.96: 681.32

ОТЛАДКА АЛГОРИТМОВ ОБРАБОТКИ РАДИОЛОКАЦИОННЫХ СИГНАЛОВ В СИСТЕМЕ GNU RADIO

Липатников В.С., инженер научно-технического отдела АО «Ордена Трудового Красного Знамени Всероссийский научно-исследовательский институт радиоаппаратуры» (АО «ВНИИРА»), г. Санкт-Петербург, e-mail: meteo@vniira.ru;

Чинёнов Д.А., инженер-программист научно-технического отдела АО «ВНИИРА» г. Санкт-Петербург, e-mail: meteo@vniira.ru/

RADAR SIGNAL PROCESSING ALGORITHMS DEBUG IN GNU RADIO SYSTEM

Lipatnikov V.S., Chinenov D.A.

The radar signal processing device model is described. The free and open-source SDR-system GNU Radio used as the modeling system. The short review of GNU Radio modeling main opportunities was given. Every fragment of radar signal processing device simple model is described in detail.

Key words: radar signal, phase shift keying, modeling, GNU Radio.

Ключевые слова: радиолокационный сигнал, фазовая манипуляция, моделирование, GNU Radio.

Введение

В настоящее время при проектировании устройств цифровой обработки радиолокационных сигналов (ЦОС) большое внимание уделяется компьютерному моделированию. При отладке алгоритмов работы устройств ЦОС моделируют не только отлаживаемые устройства, но и обрабатываемые сигналы.

При работе с конкретной элементной базой отработку алгоритмов производят, как правило, в интегрированных средах разработки, возможно с привлечением дополнительных инструментов. Например, проектирование на базе программируемых логических интегральных схем (ПЛИС) фирма Altera производят с помощью интегрированной среды Quartus, а разработку программ для цифровых процессоров обработки сигналов (ЦПОС) фирма Texas Instruments осуществляют в среде Code Composer Studio.

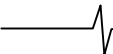
Для работы с этими средами в качестве удобного инструмента разработки прототипов устройств и алгоритмов обработки, а также визуализации массивов данных, активно используется широко распространенный пакет MATLAB/Simulink фирма MathWorks. Его тесная интеграция со средами разработки позволяет отлаживать не только прототипы устройств создаваемые с его помощью, но и реализованные на базе элементов, предоставляемых используемой средой проектирования. Так, например, взаимодействуют друг с другом среда Quartus и пакет Simulink. Также, из разработанного с помощью MATLAB/Simulink прототипа устройств можно получить автоматически сгенерированный код для встраивания его в проектируемое устройство. Однако, для достижения оптимального результата, алгоритмы, отработанные в MATLAB/Simulink, вручную переводят на язык описания аппаратуры (для ПЛИС) или

Описана модель устройства обработки радиолокационного сигнала. В качестве системы моделирования использована свободно распространяемая SDR-система с открытыми исходными кодами GNU Radio. Проведен краткий обзор ее основных возможностей по моделированию и подробно описан каждый фрагмент упрощенной модели устройства обработки радиолокационного сигнала.

язык программирования (для ЦПОС). Тем не менее, первоначальная разработка и отладка алгоритмов ЦОС в пакете MATLAB – довольно распространенное явление, на ограничение использования которого может повлиять только его высокая цена. Модули интеграции со средами разработки могут распространяться по отдельным дополнительным лицензиям, что может повысить суммарную стоимость владения средствами разработки.

Поиск среди свободно распространяемого программного обеспечения показал, что при разработке устройств обработки сигналов могут использоваться не только коммерческие продукты. Например, в [1] упомянуты свободно распространяемые программные продукты, используемые при разработке и отладке прототипов устройств на основе ПЛИС. Использование пакета Icarus Verilog позволит смоделировать работу разрабатываемого на языке описания аппаратных схем Verilog или SystemVerilog устройства, а генерация входных и визуализация выходных данных может быть осуществлена с помощью языка Python и его библиотек numpy и matplotlib.

При разработке модулей обработки сигналов радиолокационных устройств удобно компоновать их из блоков, обеспечивающих базовые функции ЦОС, таких как, например, накопление отсчетов, фильтрация или выполнение преобразования Фурье. Для моделирования устройств, состоящих из таких базовых блоков, с использованием свободно распространяемого программного обеспечения предложен набор инструментов GNU Radio [2]. Попробуем смоделировать с его помощью основные устройства обработки сигналов, которые могут



применяться в радиолокационных станциях.

Обзор особенностей GNU Radio

GNU Radio представляет собой набор программных инструментов (Software Development Tool), включающих в себя библиотеку компонентов, реализованных на языке программирования C++ с предоставлением интерфейса прикладного программирования (API) на языках C++ и Python, и ряд вспомогательных утилит. Компоненты библиотеки представляют собой множество классов, в которых реализована функциональность различных блоков обработки сигналов. Среди блоков встречаются источники сигналов, приемники сигналов и, собственно, блоки обработки. Блоки обработки могут быть синхронными (количество отсчетов входных сигналов равно количеству отсчетов в выходных сигналах), децимирующими (с понижением количества выходных отсчетов) и интерполирующими (с повышением количества выходных отсчетов).

Из блоков-источников сигналов часто используются встроенный программный генератор сигнала синусоидальной, пилообразной или прямоугольной формы, генератор шума, источник постоянного сигнала, файл с набором данных в двоичном формате, звуковой файл в формате WAV, аудиоплата ПК или подключенный совместимый аппаратный ВЧ-модуль, для которого в библиотеке имеется соответствующий модуль сопряжения. Блоками-приемниками могут быть также ВЧ-модуль, аудиоплата, звуковой или произвольный двоичный файл. Блоки-приемники, позволяющие визуализировать принимаемый сигнал во временной или частотной области, подобно тому, как он отображается на экране осциллографа или анализатора спектра, очень удобны для моделирования и отладки работы устройств ЦОС.

Среди блоков обработки могут встречаться как простые, например, задержка сигнала на заданное количество отсчетов или выполнение элементарной арифметической операции над каждым входным отсчетом, так и более сложные, например, фильтры или блоки, выполняющие преобразование Фурье. Каждый блок обработки принимает один или несколько входных потоков данных, обрабатывает их и генерирует один или несколько выходных потоков данных (количество которых может отличаться от количества входных потоков) [2]. Некоторые блоки обработки принимают на вход или передают на выход цифровые отсчеты только в виде выборок фиксированной длины (векторов), как, например, в блоке, выполняющем преобразование Фурье. В цепочке соединенных друг с другом блоков тип выходных данных одного блока, должен совпадать с типом входных данных следующего. При возникновении необходимости ввода данных одного типа в блок со входами, принимающими данные другого типа, можно использовать блоки-преобразователи типов.

При разработке приложений с использованием GNU Radio на языках C++ или Python используется следующая концепция. Создается объект класса блока верхнего уровня `top_block` или класса, унаследованного от него. Создаются объекты блоков-компонентов проектиру-

емого устройства, блоков генерации и визуализации и соединяются друг с другом с помощью метода `connect()` объекта класса `top_block`. Схему таких соединений блоков принято называть потоковым графом (flowgraph). После создания всех соединений производится запуск схемы на выполнение с помощью метода `start()`. Если присутствуют блоки, использующие графический интерфейс пользователя (блоки визуализации), то требуется обеспечить необходимую программную обвязку для сопряжения с используемой соответствующими блоками визуализации графической библиотекой.

Проектирование может сопровождаться множественным использованием повторяющихся фрагментов схем. Для повышения удобства использования таких фрагментов GNU Radio позволяет создавать пользовательские блоки, состоящие из компонентов, входящих в комплект поставки библиотеки. Такие пользовательские блоки называются иерархическими. Возможно создание пользовательских блоков, не использующих другие блоки, а выполняющих определенные пользователем функции. Эта особенность является очень полезной во время разработки и реализации подпрограмм ЦОС на языках C/C++, поскольку разработанная подпрограмма, протестированная в виде пользовательского блока с помощью GNU Radio, может быть перенесена с минимальными изменениями в рабочий проект для процессора общего назначения или специализированного процессора ЦОС, среда разработки для которого поддерживает проектирование на языках C/C++. Для разработки пользовательских блоков предоставлена утилита `gr_modtool`, упрощающая их сопряжение с остальной частью библиотеки.

Поскольку разработка моделей на языках программирования C++ и Python с использованием библиотеки GNU Radio требует определенной подготовки и более глубоких знаний API библиотеки, может быть использовано средство визуального проектирования GNU Radio Companion (GRC). С его помощью схема соединений блоков (потоковый граф) может быть представлена в удобной для восприятия форме.

Программа GNU Radio Companion позволяет создавать как схемы верхнего уровня, так и иерархические пользовательские блоки. Разработанные в GRC потоковые графы конвертируются в сценарии на языке Python, сохраняемые в текущем каталоге, а для иерархических блоков – в каталоге с настройками GRC, откуда они считываются при последующем запуске программы и могут использоваться наряду со стандартными блоками GNU Radio.

При разработке модели части тракта обработки радиолокационного сигнала, описанной ниже, некоторые ее фрагменты реализованы в виде пользовательских блоков на языке C++, другие – в виде иерархических блоков, выполненных в GRC. Для отладки фрагментов разработано несколько схем верхнего уровня, в некоторых контрольных точках которых подключены соответствующие блоки визуализации.

Рассмотрим в следующем разделе модель тракта обработки радиолокационного сигнала.

Моделирование устройств обработки сигналов

Пример тракта цифровой обработки сигналов когерентной доплеровской радиолокационной станции приведен в [3]. Рассмотрим для реализации с помощью GNU Radio похожую общую модель, примерная функциональная схема которой приведена на рис. 1.



Рис. 1. Общая функциональная схема тракта цифровой обработки РЛС

Типичный тракт цифровой обработки начинается с аналого-цифрового преобразователя (АЦП), на который сигнал на промежуточной частоте (ПЧ) поступает от приемника. После АЦП цифровые отсчеты сигнала поступают на цифровой фазовый детектор (ЦФД), который может быть выполнен в виде отдельной микросхемы или реализован на основе ПЛИС. Отсчеты синфазной (*I*) и квадратурной (*Q*) составляющих на пониженной частоте дискретизации с выхода ЦФД поступают на вход фильтра сжатия. Фильтр сжатия может быть реализован программно в цифровом процессоре обработки сигналов (ЦПОС), а при выборе ПЛИС с достаточным количеством ресурсов целесообразнее реализовать его аппаратно. Далее отсчеты сжатого сигнала накапливаются в когерентном накопителе, после чего производится обработка накопленных отсчетов сигналов, отраженных от искомым целей для решения задачи их обнаружения и вычисления доплеровской скорости. Эти процедуры выделены на схеме, приведенной на рис. 1, в отдельные функциональные блоки обнаружителя и вычислителя скорости. Проведем моделирование каждого блока и проверку его работы, используя набор инструментов GNU Radio.

Модель цифрового фазового детектора

Потоковый граф моделируемой схемы цифрового фазового детектора (ЦФД) тракта обработки радиолокационного сигнала построенный в программе GNU Radio Companion приведен на рис. 2.

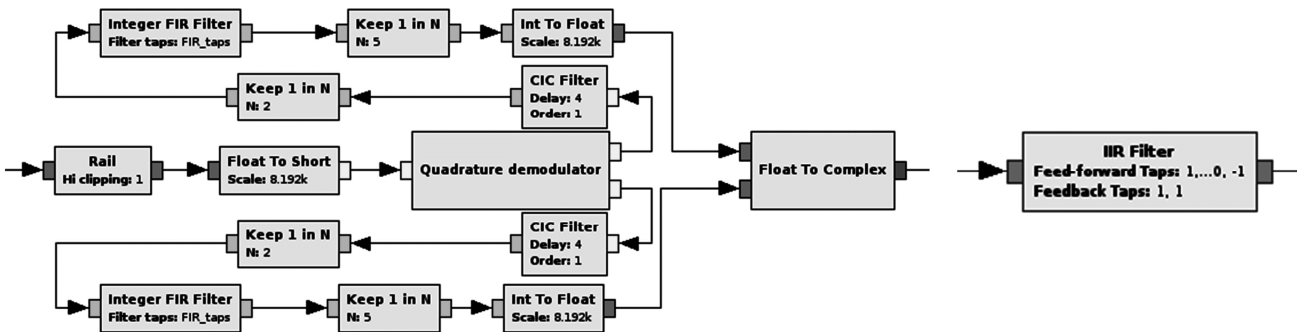


Рис. 2. Поточный граф модели ЦФД в программе GNU Radio Companion

Рис.3. Блок «IIR Filter» в GRC

После точки входа в схему блок «Rail» ограничивает входной сигнал, отсекая значения, не входящие в указанный пользователем диапазон, и присваивая им максимальное или минимальное значение соответственно при выходе за верхнюю или нижнюю границу (имитация перегрузки АЦП). Блок «Float to Short» конвертирует входной поток действительных значений в поток целых 16-разрядных значений с коэффициентом масштабирования 8192 (2^{13}). Таким образом, блоки «Rail» с параметрами среза [-1;+1] и «Float to Short» с коэффициентом масштабирования 8192 имитируют работу 14-битного АЦП.

Далее сигнал идет в блок квадратурного демодулятора «Quadrature demodulator», где раскладывается на *I* и *Q* составляющие. Пользовательский блок «Quadrature demodulator» реализован на языке C++. Поскольку промежуточная частота в данном случае равна трем четвертям частоты дискретизации, его реализация может быть сильно упрощена за счет отсутствия умножителей и генератора колебаний, как описано в [4].

После разложения сигнала на квадратуры, каждая из них подвергается обработке в каскадном интегрально-гребенчатом фильтре (Cascaded integral-comb filter – CIC-filter). Такие фильтры являются фильтрами с бесконечной импульсной характеристикой, передаточная функция которых имеет вид [5]:

$$H_k(z) = \left[\frac{1 - z^{-D}}{1 - z^{-1}} \right]^K, \tag{1}$$

где *D* – количество тактов задержки в гребенчатой части, а *K* – количество каскадов фильтра.

GNU Radio предоставляет для использования стандартный блок «IIR Filter» (рис. 3) и для реализации CIC-фильтра порядка *K* = 1 с задержкой *D* = 4 можно указать ряды значений (1, 0, 0, 0, -1) и (1, 1) в качестве параметров «Feed-forward Taps» и «Feedback Taps» соответственно.

Однако, этот блок работает только с действительными данными, поэтому блок «CIC Filter» реализован в виде пользовательского блока на языке C++ и вставлен в схему ЦФД. Поточный граф с реализацией схемы CIC-фильтра стандартными блоками GNU Radio в программе GNU Radio Companion, приведенный на рисунке 4, подробно описан в [2].

Далее блок «Keep 1 in N» с параметром «N» равным «2» отбрасывает одно значение из двух, осуществляя таким образом децимацию сигнала. После первой деци-

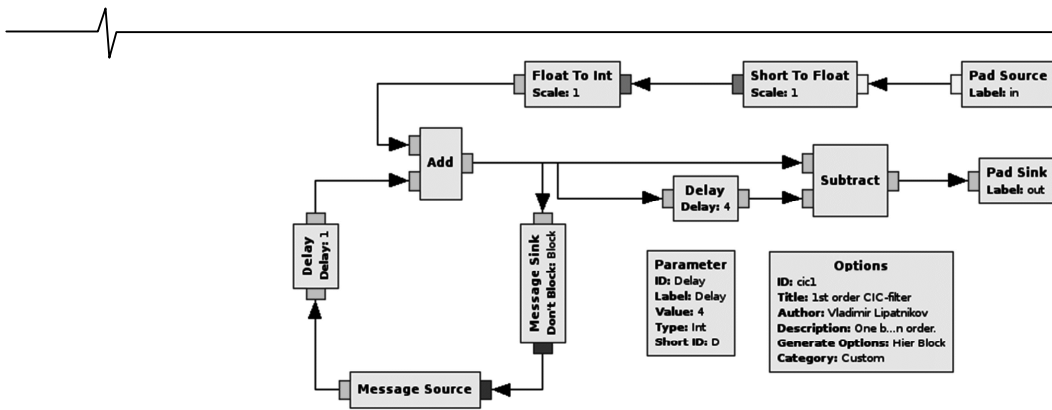


Рис. 4. Поточковый граф пользовательской реализации CIC-фильтра

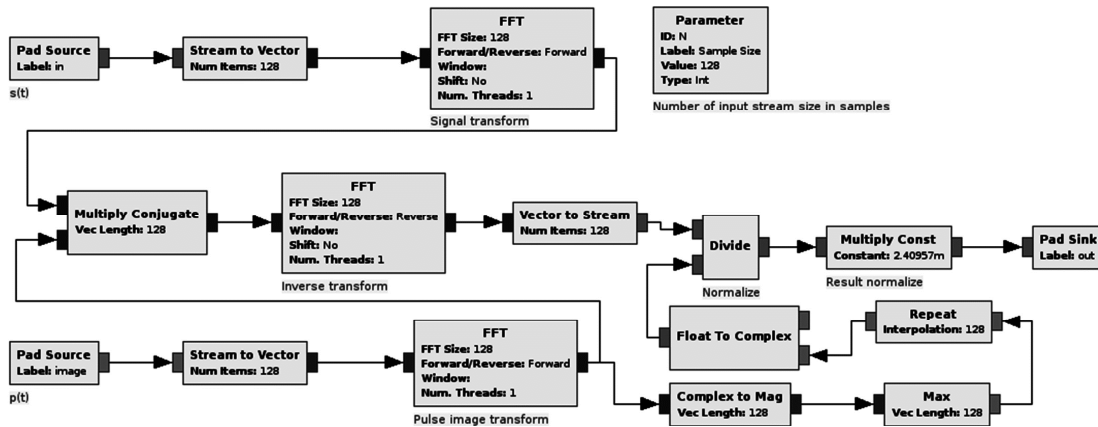


Рис. 5. Поточковый граф фильтра сжатия

мации сигнал подается на фильтр нижних частот с конечной импульсной характеристикой (КИХ), реализованный в виде пользовательского блока «Integer FIR Filter» и включенного после него блока «Keep 1 in N» с параметром «5». В данном случае снова использован пользовательский блок вместо стандартного блока «Decimating FIR Filter», поскольку стандартный блок использует арифметику с плавающей запятой, а нам необходимо проверить алгоритм работы с целочисленной арифметикой.

Далее, согласно схеме, синфазная и квадратурная составляющие из целочисленных значений конвертируются в действительные и объединяются в блоке «Float to Complex» в качестве действительной и мнимой частей комплексного сигнала, следующего на выход из данного пользовательского блока для дальнейшей обработки в устройствах, позволяющих использовать арифметику с плавающей запятой.

Модель фильтра сжатия

Предположим, что фильтр сжатия будет реализован на микропроцессорном устройстве, имеющем арифметику с плавающей запятой, поэтому для его моделирования поначалу могут быть использованы блоки GNU Radio, входящие в комплект его поставки.

Рассмотрим поточковый граф простого фильтра сжатия, который может быть реализован через блоки быстрого преобразования Фурье (БПФ), приведенный на рис. 5.

У фильтра имеются два входа: первый – для отсчетов принятого сигнала $p = \{p_0, p_1, \dots, p_{N-1}\}$, второй – для выборки зондирующего импульса $s = \{s_0, s_1, \dots, s_{N-1}\}$.

Выходной сигнал фильтра должен быть равен

$$s_{сж.} = \frac{4\pi^2}{N} \frac{F^{-1}\{P^* \cdot S\}}{\max\{|P|\}}, \quad (2)$$

где $P = \{p\}$, $S = \{s\}$, $F\{\}$ – оператор дискретного преобразования Фурье (ДПФ), $F^{-1}\{\}$ – оператор обратного дискретного преобразования Фурье (ОДПФ), $*$ – комплексное сопряжение, N – количество отсчетов принятого сигнала.

На схеме, приведенной на рис. 5, выборка цифровых отсчетов зондирующего импульса имеет действительный тип данных, а выборка отсчетов принятого сигнала после ЦФД – комплексный. Параметр N (блок «Parameter») пользовательского блока, реализующего фильтр сжатия, определяет количество отсчетов в выборке сигнала. Поскольку блоки «FFT» GNU Radio работают не с единичными отсчетами, а с векторами данных, сначала входные сигналы накапливаются в буферах и преобразовываются в векторы по N отсчетов в блоках «Stream to Vector». Далее в блоке «Multiply Conjugate» вектор результатов ДПФ принятого сигнала умножается на комплексно-сопряженный вектор Фурье-образа цифровых отсчетов зондирующего сигнала, и выполняется обратное преобразование Фурье в следующем блоке «FFT». Далее блоком «Vector to Stream» выполняется преобразование вектора выходных данных блока «FFT» в поток отсчетов. Выходной сигнал блока нормируется с использованием следующих блоков GNU Radio:

1. «Complex to Mag» – вычисление амплитуды комплексного результата;
2. «Max» – вычисление значения максимальной амплитуды комплексного результата (одно выходное значение при входном векторе длительностью N отсчетов);

3. «Repeat» – повтор выходного значения блока «Max» столько раз, сколько отсчетов во входном векторе блока «Max»;

4. «Float to Complex» – преобразование действительного типа данных в комплексный, так как последующее деление в блоке «Divide» осуществляется над комплексными отсчетами.

Для уменьшения уровня боковых лепестков выходного сигнала фильтра сжатия, он может быть построен с использованием метода обратных пульсаций [6]. Выходной сигнал такого фильтра определяется как

$$s_{\text{сжм}} = \frac{4\pi^2}{N} \cdot \frac{F^{-1}\{|Q|^2 \cdot S / P\}}{\max\{|Q|^2 / |P|\}}, \quad (3)$$

где $Q = \{q\}$, а $q = \{q_0, q_1, \dots, q_{N-1}\}$ – элементарный импульс кодовой последовательности, использованной для генерации зондирующего сигнала. Поточковый граф модели такого фильтра, построенный из стандартных блоков GNU Radio, приведен на рис. 6. В отличие от поточкового графа фильтра реализованного в виде пользовательского иерархического блока, приведенного на рис. 5, поточковый граф данного фильтра имеет еще

один параметр l – количество отсчетов в одном элементе кодовой последовательности зондирующего сигнала (второй блок «Parameter»). Этот параметр использован для формирования вектора заданных отсчетов Q^2 в блоке-источнике «Vector Source», принимающего в качестве параметра выражение на языке Python с подстановкой переменных Q и q , присутствующих на поточковом графе в блоках «Variable», тоже принимающих выражения на языке Python в качестве своих параметров. Вектор выходных отсчетов блока «Vector Source» делится на вектор Фурье-образа зондирующего импульса. Полученный результат перемножается с Фурье-образом принятого сигнала. Нормировка выходных результатов фильтра выполняется так же, как и в поточковом графе простого фильтра сжатия.

Для проверки и сравнения моделей оптимального и подоптимального фильтров сжатия подготовлен поточковый граф, приведенный на рис. 7-9. Для удобства визуального восприятия, разделения функциональных групп поточкового графа и исключения его загромождения соединительными линиями использованы блоки «Virtual Sink» и «Virtual Source».

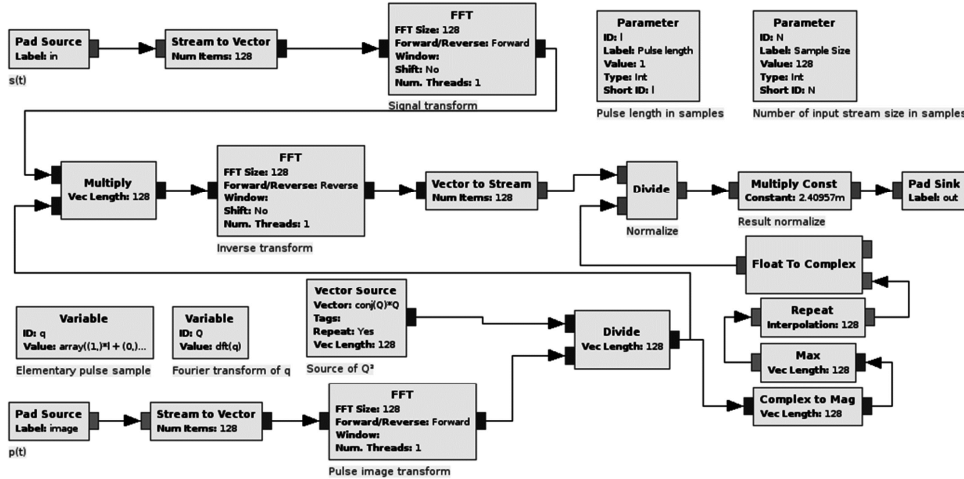


Рис. 6. Поточковый граф подоптимального фильтра сжатия

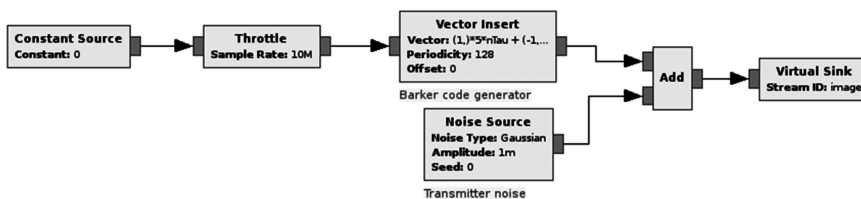


Рис. 7. Поточковый граф проверки фильтров сжатия: генерация импульсов

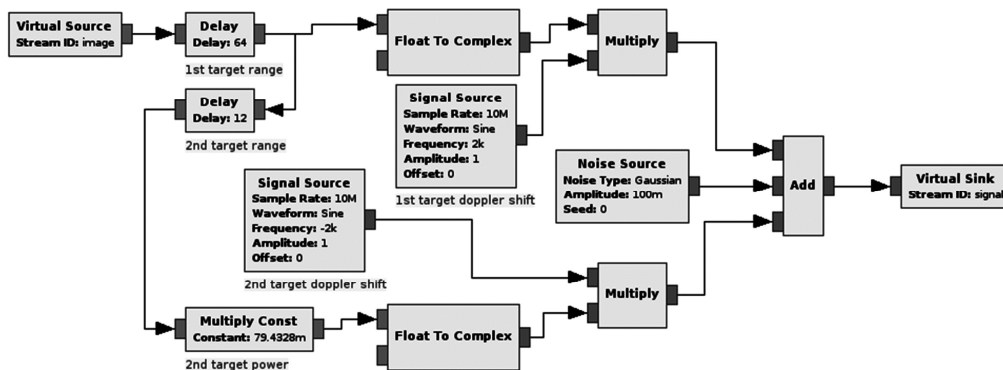


Рис. 8. Поточковый граф проверки фильтров сжатия: имитация двух целей

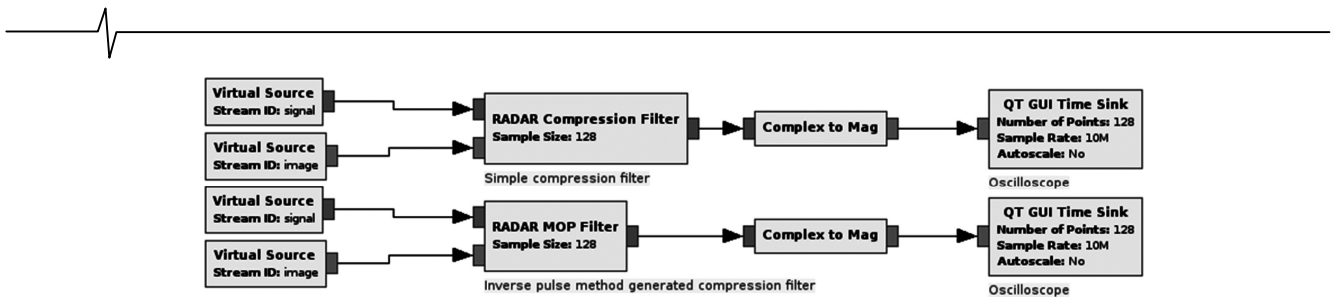


Рис. 9. Поточковый граф проверки фильтров сжатия: схема включения фильтров

Последовательность блоков на рис. 7 формирует 13-символьную кодовую последовательность Баркера. Блок «Throttle» нужен для ограничения времени работы процессора в случае, если его производительности более чем достаточно для работы на заданной частоте дискретизации. Блок «Vector Insert» вставляет в поток отсчетов, равных нулю, отсчеты, соответствующие кодовой последовательности. В блоке «Noise Source» имитируются шумы передатчика, которые складываются с исходной кодовой последовательностью в блоке «Add». С помощью блока «Virtual Sink» с идентификатором «image» осуществляется соединение со следующим фрагментом без загромождения схемы. Модель сигнала выполнена в квадратурах на нулевой частоте.

На рис. 8 показан поточковый граф генератора имитационного эхо-сигнала от двух целей с разными скоростями (доплеровскими частотами) и разной ЭПР (мощностью сигнала).

Блок «Virtual Source» с идентификатором «image» продолжает схему, приведенную на рис. 7, окончившаяся блоком «Virtual Sink» с таким же идентификатором. Два блока «Delay» задерживают сигнал на разное количество отсчетов: второй блок задерживает сигнал на такое количество отсчетов, чтобы второй сигнал после сжатия наложился на боковой лепесток первого сжатого сигнала. Амплитуда второго сигнала уменьшена на 22 дБ с помощью блока «Multiply Const». Формирование доплеровских сдвигов осуществляется в блоках «Multiply» в комплексном виде подмешиванием синусоидальных сигналов, сгенерированных блоками «Signal Source». Сигналы складываются в блоке «Add» с добавлением белого шума заданной мощности, сгенерированного блоком «Noise Source». Получившийся сигнал может быть передан в другую часть схемы с помощью блока «Virtual Sink» с идентификатором «signal».

На фрагменте поточкового графа, приведенном на рис. 9, показано подключение сигналов от блоков «Virtual Sink» с идентификаторами «signal» и «image» к пользовательским блокам «RADAR Compression Filter» и «RADAR MOP Filter» с разными реализациями фильтров сжатия, описанными выше.

В блоках «Complex to Mag» вычисляются амплитуды выходных комплексных сигналов фильтров и, затем, визуализируются с помощью блоков «GUI Time Sink». Окно с визуализацией результатов работы фильтров сжатия блоками «GUI Time Sink» после запуска поточкового графа на выполнение показано на рис. 10. Верхняя осциллограмма отражает результат оптимального фильтра сжатия, нижняя – подоптимального.

Как видно из рисунка, оптимальная фильтрация приводит к высокому уровню боковых лепестков, в которых

теряется сигнал от второй имитируемой цели, а в сжатом подоптимальным фильтром сигнале можно четко разделить две составляющие.

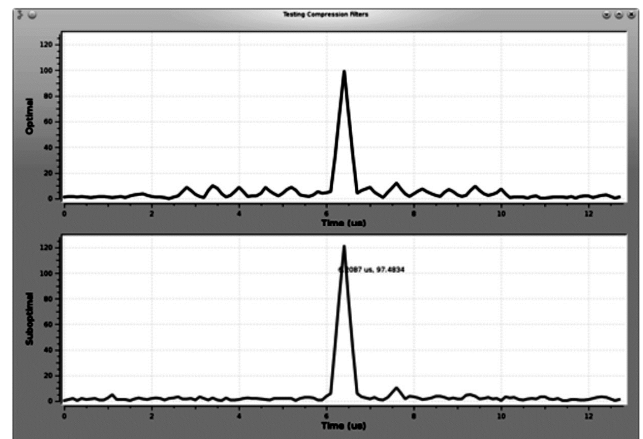


Рис. 10. Осциллограмма результатов сжатия

Таким образом, собрав несложную схему проверки в программе GNU Radio Companion, мы провели наглядное сравнение двух алгоритмов фильтрации.

Модель когерентного накопителя и обнаружителя

Пример поточковых графов блоков, моделирующих когерентный накопитель, пороговый обнаружитель и вычислитель доплеровского сдвига для восьми интервалов по дальности приведен на рис. 11 – 13. Как видно из рис. 11, в GRC схема выглядит довольно громоздкой даже для восьми отсчетов по дальности. Поскольку при когерентном накоплении сигнала вычисления производятся в каждой дискрете по дальности, которых может быть довольно много, использование программы GNU Radio Companion для моделирования дальнейших устройств обработки нецелесообразно. Для этих целей лучше реализовать на языке C++ пользовательский блок когерентного накопителя с вычислением преобразования Фурье, мощности сигнала, пороговой обработкой и оценкой доплеровской скорости в каждой дискрете по дальности в виде иерархического блока GNU Radio, который потом может быть включен в общий поточковый граф модели радиолокатора. Фрагмент создания блоков «Stream to Streams» и «Stream Mux» для иерархического пользовательского блока «Doppler Accumulator» и соединений между ними на языке Python приведен на рис. 14.

Пример вида экрана индикатора скорость-дальность (номер доплеровского канала – время), полученного с помощью блока «GUI Time Sink», приведенного на рис. 13, показан на рис. 15. На данном рисунке цели обнаружены на разных дальностях в 12-м и 4-м доплеровских каналах из 16.

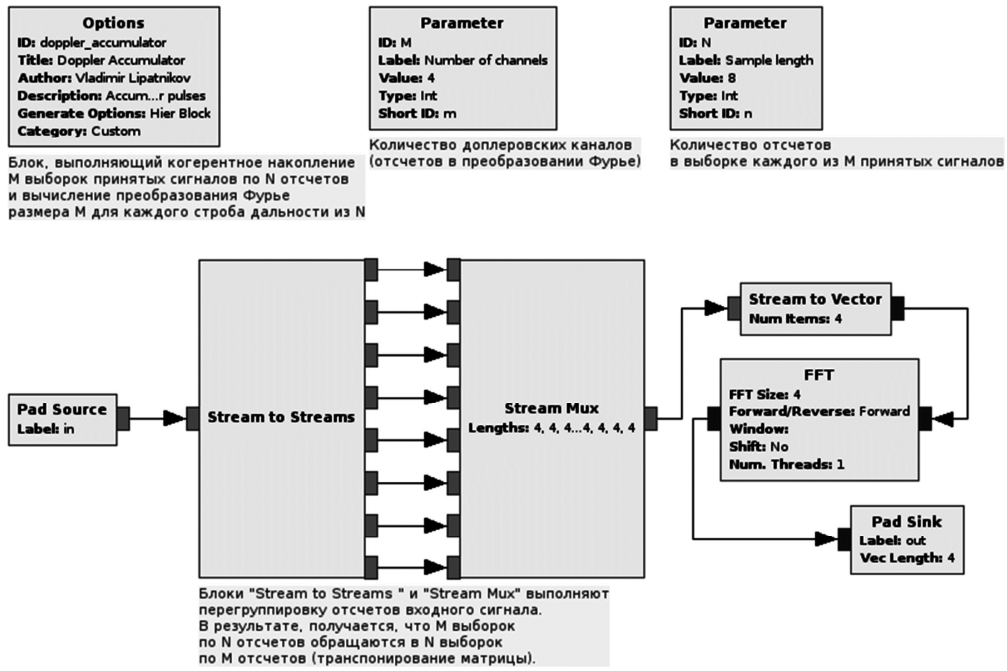


Рис. 11. Поточковый граф когерентного накопителя

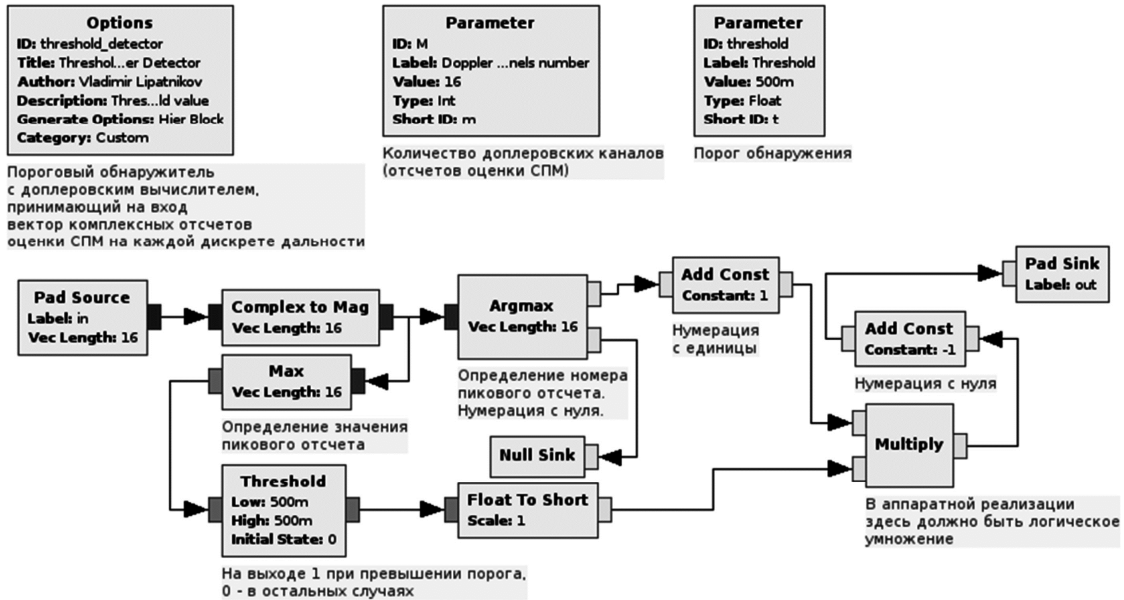


Рис. 12. Поточковый граф вычислителя доплеровской скорости с пороговым обнаружителем

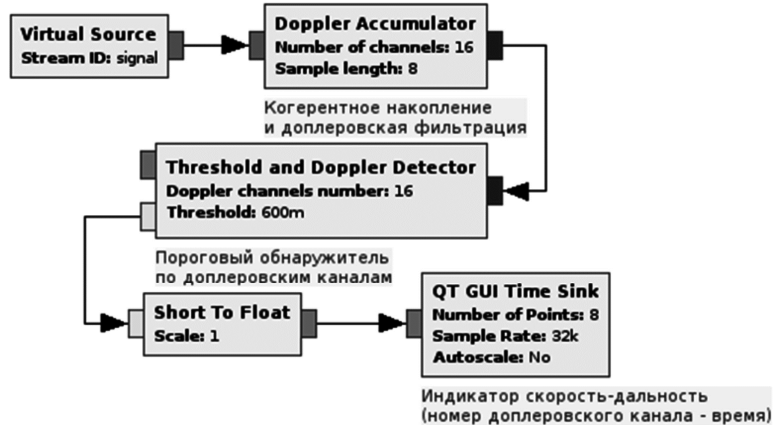


Рис. 13. Поточковый граф подключения индикатора скорость-дальность

```

#...Blocks
self.blocks_stream_to_streams_0 =
    blocks.stream_to_streams(gr.sizeof_gr_complex*1, N)
self.blocks_stream_mux_0 =
    blocks.stream_mux(gr.sizeof_gr_complex*1, ((M,)*N))
#...Connections
for i in range(self.N):
    self.connect((self.blocks_stream_to_streams_0, i),
                (self.blocks_stream_mux_0, i))

```

Рис. 14. Фрагмент листинга реализации блока «Doppler Accumulator» на языке Python

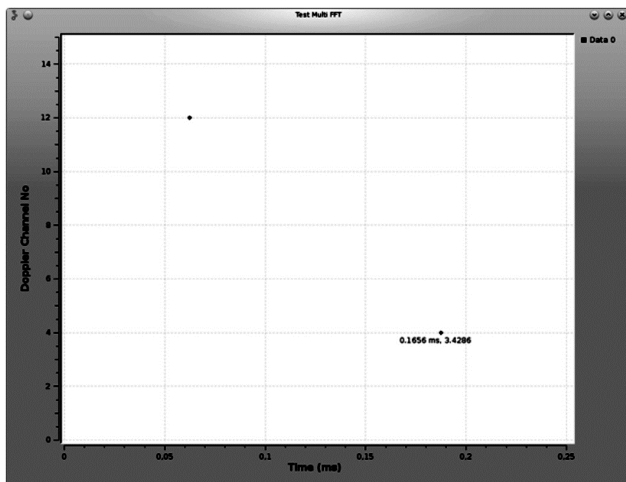


Рис. 15. Вид индикатора скорость-дальность (номер канала – время)

Заключение

Рассмотренная модель тракта обработки радиолокационного сигнала является сильно упрощенной, но вполне достаточной для демонстрации некоторых возможностей отладки алгоритмов ЦОС в системе GNU Radio. Постепенный переход от фрагментов модели, выполненных в виде иерархических блоков GRC, к пользовательским блокам, реализованным на языке C++, позволит приблизить модель к созданию реального устройства. Но при разработке первоначальных моделей в GRC лучше не увлекаться чрезмерным загромождением потоковых графов, поскольку это приводит не только к ухудшению восприятия, но и к замедлению работы самой программы визуального проектирования.

Это связано с тем, что она разработана на языке Python и, по всей видимости, текущая версия (на момент написания статьи – это 3.7.8) использует графические возможности компьютера далеко не самым наилучшим образом. Тем не менее, режим выполнения самих моделей близок к режиму реального времени при использовании разумных частот дискретизации (в нашем случае 10 МГц) и простых моделей отраженных сигналов. Использование более сложных моделей сигналов, отраженных, например, от распределенных объектов, потребует перехода от глобальной модели, выполняющейся бесконечно, пока пользователь не прервет ее выполнение, к модели, выполняющейся определенное число раз с выводом данных не в блок визуализации GNU Radio, а на график с помощью библиотеки matplotlib. В любом случае, система GNU Radio позволит выполнить только отладку алгоритмов: проверить их работу и/или выявить функциональные ошибки. Профилирование и оптимизация подпрограмм обработки должны выполняться с помощью средств разработки для целевой платформы, на базе которого проектируется устройство.

Литература

1. Максфилд К. Проектирование на ПЛИС. Курс молодого бойца. – М.: Издательский дом «Додэка-XXI», 2007. – 408 с.
2. Липатников В.С. Моделирование устройств цифровой обработки сигналов с помощью программного инструмента GNU Radio // 17-я Международная конференция «Цифровая обработка сигналов и ее применение – DSPA-2015». – Т. 2. – М., 2015. – С. 641–645.
3. Соловьев А.Г. Тракт цифровой обработки сигналов когерентной импульсно-доплеровской РЛС // Цифровая обработка сигналов. – 2000. – №2. – С. 2-5.
4. Справочник по радиолокации / Под ред. М.И. Скопника. – М.: Техносфера, 2015. – Т. 2. – 680 с.
5. Справочник по радиолокации / Под ред. М.И. Скопника. – М.: Техносфера, 2015. – Т. 1. – 672 с.
6. Сеницын Е.А., Чиненов Д.А. Особенности обработки сложных фазоманипулированных радиолокационных сигналов методом обратных пульсаций // Успехи современной радиоэлектроники. – 2015. – №3. – С. 119–121.