

**НАУЧНЫЙ ЖУРНАЛ**

**ТЕЛЕКОММУНИКАЦИИ  
И ИНФОРМАЦИОННЫЕ  
ТЕХНОЛОГИИ**

**№1-2016**

*(Дата издания: июнь 2016 г.)*

## **РЕДАКЦИОННАЯ КОЛЛЕГИЯ:**

Андреев Владимир Александрович  
д.т.н., профессор, Поволжский государственный университет телекоммуникаций и информатики,  
Самара, Россия

Бачевский Сергей Викторович  
д.т.н., профессор, Санкт-Петербургский государственный университет телекоммуникаций им. проф.  
Бонч-Бруевича, Санкт-Петербург, Россия

Зимин Игорь Викторович  
Кыргызский государственный технический университет имени И.Раззакова. Институт электроники и  
телекоммуникаций, Бишкек, Кыргызстан

Ланчиков Павел Николаевич  
НП Учебный центр Huawei (Москва), Шеньчжень, Китай

Маркосян Мгер Вардкесович  
к.т.н., доцент, Ереванский НИИ средств связи, Ереван, Армения

Прохода Александр Николаевич  
к.воен.н., доцент, Балтийский военно-морской институт им. Ф.Ф. Ушакова, Калининград, Россия

Рябко Борис Яковлевич  
д.т.н., профессор, Сибирский государственный университет телекоммуникаций и информатики, Ново-  
сибирск, Россия

Соколов Владимир Андреевич  
к.т.н., профессор, Московский технический университет связи и информатики, Москва, Россия

Титов Евгений Вадимович  
к.т.н., профессор, Московский технический университет связи и информатики, Москва, Россия

## **УЧРЕДИТЕЛЬ:**

**МОСКОВСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ СВЯЗИ  
И ИНФОРМАТИКИ (МТУСИ)**

## **РЕДАКЦИОННАЯ ПОДГОТОВКА:**

**Отдел информационных ресурсов и интеллектуальной собственности  
МТУСИ (ОИРИС МТУСИ)**

*В журнал включены материалы лучших докладов, представленных на научных конференциях 7-го Молодежного научного Форума МТУСИ.*

# СОДЕРЖАНИЕ

## «Цифровые технологии радиосвязи и телерадиовещания»

<i>Сухов Е.С., Смирнов Е.В.</i> <b>ПОВЫШЕНИЕ ПОМЕХОЗАЩИЩЕННОСТИ АПЕРТУРНЫХ АНТЕНН</b>	7
<i>Чупеев А.Р., Симонов П.И.,</i> <b>ПРИМЕНЕНИЕ ТЕХНОЛОГИИ SDR НА ОСНОВЕ PXI</b>	11
<i>Ворожищев И.В., Минаева О.Н.</i> <b>ИССЛЕДОВАНИЕ СПОСОБОВ ФОРМИРОВАНИЯ ПОСЛЕДОВАТЕЛЬНОСТЕЙ ЗАДОВА-ЧУ</b>	15
<i>Славянский А.С., Сперанский В.С.</i> <b>БЕСПРОВОДНЫЕ СЕНСОРНЫЕ УСТРОЙСТВА НА ОСНОВЕ ЭФФЕКТА ПЕРЕИЗЛУЧЕНИЯ</b>	20
<i>Толкачев М.М., Лобов Е.М.</i> <b>РАЗРАБОТКА АЛГОРИТМА СЛЕЖЕНИЯ ЗА НЕСУЩЕЙ ПРИ ИСПОЛЬЗОВАНИИ НИЗКОСКОРОСТНЫХ ТУРБОКОДОВ В ПОМЕХОЗАЩИЩЕННОЙ СИСТЕМЕ СВЯЗИ</b>	26
<i>Токарева Е.М., Комаров С.Н.</i> <b>КОНЦЕПЦИЯ РАДИОТЕХНИЧЕСКОГО ЭФИРНОГО ВЕЩАНИЯ «РАДИО МТУСИ»</b>	30
<i>Болотов А.О., Холюков Р.Г., Кубицкий А.А., Бажин А.В.</i> <b>ИССЛЕДОВАНИЕ ВЛИЯНИЯ ДИНАМИЧЕСКИХ ГОЛОВОК НА ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ УСИЛИТЕЛЯ НА ЛАМПАХ И УСИЛИТЕЛЯ НА ПТ В ПРОГРАММЕ MICRO-CAP</b>	34
<i>Тактакишвили В.Г., Комаров С.Н.</i> <b>РАЗРАБОТКА И ИССЛЕДОВАНИЕ ПЕРЕДАТЧИКА С КВАДРАТУРНОЙ АМПЛИТУДНОЙ МОДУЛЯЦИЕЙ</b>	38
<i>Колпаков В. М., Синюхин В.С., Сорокин А.С.</i> <b>ОЦЕНКА ВЛИЯНИЯ ВЫБОРА МОДЕЛИ СОТОВОГО РАДИОПОКРЫТИЯ НА ХАРАКТЕРИСТИКИ ФУНКЦИОНИРОВАНИЯ СЕТИ МОБИЛЬНОЙ СВЯЗИ</b>	42
<i>Афонин С.П., Кубанков Ю.А.</i> <b>РАЗВИТИЕ ЗАКОНОДАТЕЛЬСТВА О КОММЕРЧЕСКОЙ ТАЙНЕ</b>	47
<i>Григорьев И.Д., Орлов В.Г.</i> <b>АНАЛИЗ ТИПОВ ПРИЛОЖЕНИЙ В АВИАЦИОННОЙ СИСТЕМЕ ПЕРЕДАЧИ ДАННЫХ</b>	50

<i>Пушкина Е.В., Фролова М.М., Власюк И.В.</i> <b>ИССЛЕДОВАНИЕ И РАЗРАБОТКА МЕТОДА РЕСТАВРАЦИОННОЙ ОБРАБОТКИ СИГНАЛОВ ДЛЯ СИСТЕМ ЦИФРОВОГО ТЕЛЕВИДЕНИЯ</b>	<b>54</b>
<i>Топильский С.А., Власюк И.В.</i> <b>ИССЛЕДОВАНИЕ МЕТОДОВ КОДИРОВАНИЯ И ПРОТОКОЛОВ ПЕРЕДАЧИ СИГНАЛОВ ТЕЛЕВИЗИОННЫХ ИЗОБРАЖЕНИЙ В СИСТЕМАХ ИНТЕРНЕТ-ТЕЛЕВИДЕНИЯ ПО ТЕХНОЛОГИИ OTT-TV</b>	<b>58</b>
<i>Кухтенко В.Ф., Фриск В.В.</i> <b>ИСПОЛЬЗОВАНИЕ СИСТЕМЫ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ МАТЛАВ И SIMULINK В ЛАБОРАТОРНОЙ РАБОТЕ «ИССЛЕДОВАНИЕ ПАССИВНЫХ ЦЕПЕЙ ПРИ ГАРМОНИЧЕСКОМ ВОЗДЕЙСТВИИ НА ПОСТОЯННОЙ ЧАСТОТЕ»</b>	<b>63</b>
<i>Панкратов Ф.С., Жуков И.А., Ганин В.И.</i> <b>МЕТОД СПЕКТРАЛЬНОГО АНАЛИЗА АУДИО-СИГНАЛА FFT (БЫСТРОЕ ПРЕОБРАЗОВАНИЕ ФУРЬЕ)</b>	<b>66</b>
<i>Тихомирова Е.О., Барков А.С., Степанова А.Г.</i> <b>ФИЛЬТРЫ В МАТЛАВ</b>	<b>70</b>
<i>Каледина А.В., Чехов А.С., Каравашкина В.Н.</i> <b>ПРИМЕНЕНИЕ HIGH-K ПЛЁНОК В СОВРЕМЕННОЙ ЭЛЕКТРОНИКЕ</b>	<b>74</b>
<b>«NGN: сетевые технологии и системы телекоммуникаций»</b>	
<i>Дуплищева Я.В., Шаврин С.С.</i> <b>РАЗРАБОТКА СИСТЕМЫ ФОРМИРОВАНИЯ ГРУППОВОГО СЕАНСОВОГО КЛЮЧА ДЛЯ ГРУППЫ ОБЪЕКТОВ VDL-4</b>	<b>77</b>
<i>Демин И.П., Жуков В.Ю., Киселев А.С., Смирнов С.И., Джалалов И.К.</i> <b>МАКЕТ ФРАГМЕНТА СЕТИ ДОСТУПА GERON БАЗОВОЙ КАФЕДРЫ "РОСТЕЛЕКОМ" МТУСИ</b>	<b>81</b>
<i>Вилков А. Р., Касапов К.В., Маликова Е.Е.</i> <b>ПОСТАНОВКА ЛАБОРАТОРНЫХ РАБОТ НА БАЗЕ ОБОРУДОВАНИЯ SI3000 КОМПАНИИ ISKRATEL</b>	<b>84</b>
<i>Зеленский П.В., Сазонов А.С., Маликова Е.Е.</i> <b>РАЗРАБОТКА ЛАБОРАТОРНЫХ РАБОТ НА БАЗЕ СИМУЛЯТОРА HUAWEI ENSP</b>	<b>88</b>
<i>Каменков М.А., Синюхин В.С., Часовников Е.Д.</i> <b>ПОСТАНОВКА ЛАБОРАТОРНЫХ РАБОТ ПО ИССЛЕДОВАНИЮ МОБИЛЬНЫХ ТЕЛЕФОНОВ НА ЛАБОРАТОРНОМ СТЕНДЕ</b>	<b>92</b>

<i>Стрельников В.Г., Шалаумов Э.В., Яшина М.В.</i> <b>ПРОЕКТИРОВАНИЕ МАСШТАБИРУЕМОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ С ВОЗМОЖНОСТЬЮ ИНТЕГРАЦИИ ОБРАБОТКИ ДАННЫХ В МОБИЛЬНОЕ ПРИЛОЖЕНИЕ</b>	<b>96</b>
<i>Денисов И.А., Пименов С.А., Шелухин О.И.</i> <b>НЕКОТОРЫЕ ОСОБЕННОСТИ ИЗМЕРЕНИЯ, ПОЛУЧЕНИЯ И ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ ТРАФИКОВЫХ МАТРИЦ</b>	<b>99</b>
<i>Новиков А.М., Волков М.М.</i> <b>ОЦЕНКА РАССТОЯНИЯ МЕЖДУ ДВУМЯ УСТРОЙСТВАМИ, СОЕДИНЁННЫМИ ЧЕРЕЗ Wi-Fi</b>	<b>101</b>
<i>Толмачев Р.В., Воронова Л.И.</i> <b>РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ КОНТЕНТ-АНАЛИЗА ИНТЕРНЕТ-ПУБЛИКАЦИЙ</b>	<b>103</b>
<i>Курдюмов А.А., Загвоздкина А.В.</i> <b>СОЗДАНИЕ ПРИЛОЖЕНИЕ ДЛЯ WINDOWS, УПРОЩАЮЩЕГО ВЗАИМОДЕЙСТВИЕ ПОЛЬЗОВАТЕЛЯ С СИСТЕМОЙ ПРИ КОПИРОВАНИИ ФАЙЛОВ</b>	<b>107</b>
<i>Акимов П.М., Губанкова Е.В., Гузеев А.В., Сосновиков Г.К.</i> <b>ИССЛЕДОВАНИЕ ПРАКТИЧЕСКОГО ИСПОЛЬЗОВАНИЯ МЕТОДОВ ПАРСИНГА ДАННЫХ ДЛЯ РАЗРАБОТКИ ВЕБ-СЛУЖБ НА ПРИМЕРЕ ПАРСИНГА МАТЕМАТИЧЕСКИХ ВЫРАЖЕНИЙ</b>	<b>110</b>
<i>Мирошниченко А.В., Семенова Т.И.</i> <b>ПРОГРАММА ДЛЯ БИНАРНОГО ШИФРОВАНИЯ РОТОРНЫМ АЛГОРИТМОМ</b>	<b>114</b>
<i>Морозов И.С., Гуриков С.Р.</i> <b>РАЗРАБОТКА ЭЛЕКТРОННОГО МОДУЛЯ ПОСЕЩАЕМОСТИ ДЛЯ ДЕКАНАТА ВУЗА</b>	<b>118</b>
<i>Пугачев Д.А., Шелухин О.И.</i> <b>ИСПОЛЬЗОВАНИЕ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ АВТОМАТИЧЕСКОЙ ИДЕНТИФИКАЦИИ ЛИЧНОСТИ ПО ИЗОБРАЖЕНИЯМ ЛИЦ В ВИДЕОПОТОКЕ</b>	<b>122</b>
<i>Креймер А.В., Михайлова А.В., Новиков А.С., Беленькая М.Н.</i> <b>РАЗРАБОТКА МУЛЬТИМЕДИА-СЕРВЕРА ПОД УПРАВЛЕНИЕМ RTNLINUX</b>	<b>126</b>
<i>Прохоров Д.О., Трофлянина Н.В., Фомин С.А., Беленькая М.Н.</i> <b>АНАЛИЗ СЕТЕВОГО ТРАФИКА НА ОСНОВЕ КОЛИЧЕСТВЕННЫХ ХАРАКТЕРИСТИК ПОТОКА ТРАНСПОРТНОГО УРОВНЯ</b>	<b>130</b>

<i>Челышков С.А., Гузеев А.В.</i> <b>ИССЛЕДОВАНИЕ СРЕДСТВ МАРШРУТИЗАЦИИ ПРИ ИНТЕГРАЦИИ ИНФОРМАЦИОННЫХ СИСТЕМ</b>	<b>133</b>
<i>Трофлянин В.В., Беленькая М.Н.</i> <b>ИССЛЕДОВАНИЕ МЕТОДОВ И СРЕДСТВ СИСТЕМ ПЕРЕДАЧИ МУЛЬТИМЕДИЙНЫХ ДАННЫХ</b>	<b>136</b>
<i>Арбузов С.В., Стогова К.В., Волчков В.П.</i> <b>СИНТЕЗ ОПТИМАЛЬНОГО ФИЛЬТРОВОГО ДЕМОДУЛЯТОРА ДЛЯ ЦИФРОВОЙ СИСТЕМЫ СВЯЗИ С ПОСИМВОЛЬНОЙ ПЕРЕДАЧЕЙ</b>	<b>140</b>
<i>Безруков И.М., Волчков В.П.</i> <b>ИССЛЕДОВАНИЕ ПОМЕХОУСТОЙЧИВОСТИ ЦИФРОВОЙ СИСТЕМЫ СВЯЗИ С КАНАЛЬНЫМ ПРЕКОДЕРОМ И ФИНИТНОЙ ПОСИМВОЛЬНОЙ ПЕРЕДАЧЕЙ</b>	<b>145</b>
<i>Дуплищева Я.В., Калугин Ю.А., Сухоруков А.С., Терехов А.Н.</i> <b>АНАЛИЗ ВОЗМОЖНОСТИ ПРОГРАММНО-АППАРАТНОЙ РЕАЛИЗАЦИИ ЛАБОРАТОРНОЙ РАБОТЫ «ФОРМИРОВАНИЕ СИГНАЛОВ АМПЛИТУДНОЙ МОДУЛЯЦИИ»</b>	<b>150</b>
<i>Дюркович И.В., Силантьева А.С., Корзинкин В.В.</i> <b>ПОКОЛЕНИЯ ПРОГРАММ ИСКУССТВЕННО-СОЦИАЛЬНЫХ СИСТЕМ</b>	<b>153</b>
<i>Походун А.И., Павлюк В.В.</i> <b>ОПТИМАЛЬНАЯ ФИЛЬТРАЦИЯ ДИСКРЕТНОГО СИГНАЛА С УЧЕТОМ МЕЖСИМВОЛЬНОЙ ИНТЕРФЕРЕНЦИИ</b>	<b>156</b>

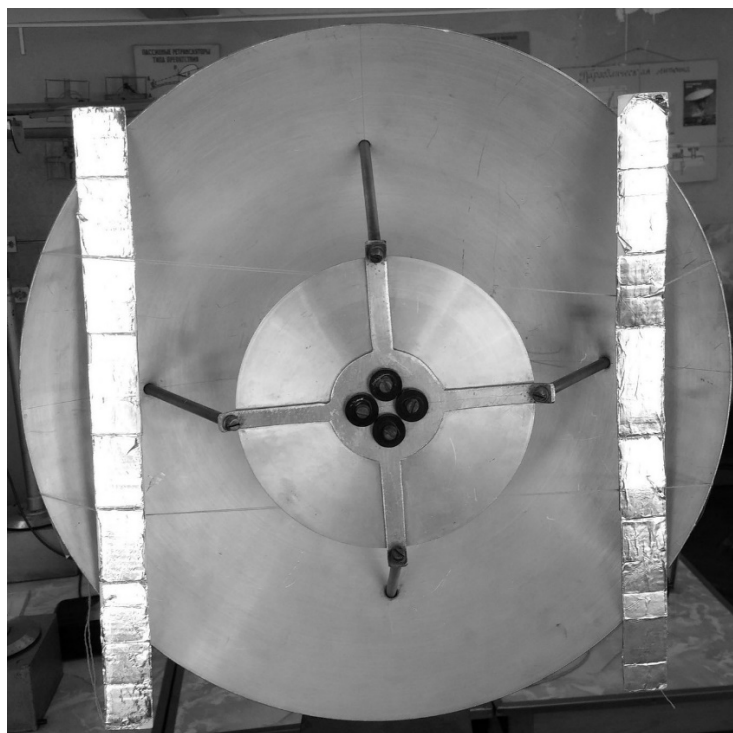
# ПОВЫШЕНИЕ ПОМЕХОЗАЩИЩЕННОСТИ АПЕРТУРНЫХ АНТЕНН

*Сухов Е.С.,  
МТУСИ, студент группы БЗС1201  
Смирнов Е.В.,  
МТУСИ, доцент*

В современном мире в связи со все большим внедрением информационных технологий и обслуживающих их радиотехнических устройств электромагнитная обстановка ухудшается с каждым годом. Для поддержания всех уже введенных в эксплуатацию систем и беспрепятственного внедрения новых требования к их электромагнитной совместимости растут и усложняются. Электромагнитная совместимость – способность технических средств функционировать с заданным качеством в заданной электромагнитной обстановке и не создавать недопустимых электромагнитных помех другим электротехническим средствам. Одним из компонентов радиотехнических систем, в значительной степени, определяющей их помехозащищенность, является антенна.

Помехозащищенность самих антенн характеризуются уровнем боковых лепестков диаграммы направленности, так как чаще всего именно с направления боковых лепестков приходит помеха, если антенна работает в режиме приема, или создается помеха, если антенна работает в режиме передачи. Одним из перспективных методов снижения уровня боковых лепестков (УБЛ) уже находящихся в эксплуатации антенн является внесение в раскрыв антенн неоднородностей, например, экранов или линз [1]. В указанной работе анализ характеристик таких антенн проводился в режиме передачи апертурным методом и для неоднородностей простой геометрической формы. В отличие от передающего режима, исследование электродинамических характеристик антенн в приемном режиме позволяет проводить более тонкие исследования, как направленных свойств антенн, так и их рассеивающих свойств. В [3-4] на основе метода ортогональных составляющих [2] было показано, что рассеянное антенной поле имеет сложную структуру, каждая компонента которой выполняет определенную функцию в формировании электродинамических характеристик антенны. Было показано, что только часть поля рассеяния антенны участвует в передаче информации и формировании диаграммы направленности антенны в режиме приема. Для определения всех этих компонент поля рассеяния приемной антенны нужно знать полное поле рассеяния антенны и поле антенны в режиме передачи. В настоящее время для определения этих полей широкое применение получили программные продукты, которые позволяют проводить электродинамическое моделирование реальных антенных систем сложной конфигурации разными методами, а не только апертурным, например, CST, HFSS, FEKO. Применение этих пакетов существенно расширяют возможности исследования влияния неоднородностей, расположенных в апертуре антенны, на формирование минимумов ДН антенны в направлении боковых лепестков.

Теоретические и экспериментальные исследования в этом направлении проводятся в рамках выполнения моей выпускной работы и включают в себя изучение возможности повышения помехозащищенности антенн Кассегрена, однозеркальных параболических антенн, а также двухзеркальных антенн типа АДЭ. Ниже, учитывая ограниченный объем публикации, представлены результаты исследования возможности повышения помехозащищенности только двухзеркальных антенн Кассегрена с помощью внесения в раскрыв антенны неоднородностей типа экран. Этот метод прост, эффективен, а главное, экономически целесообразен для использования на уже работающих устройствах для корректировки их параметров под изменившиеся требования. Внешний вид исследуемой антенны Кассегрена с экранами показан на рисунке 1.

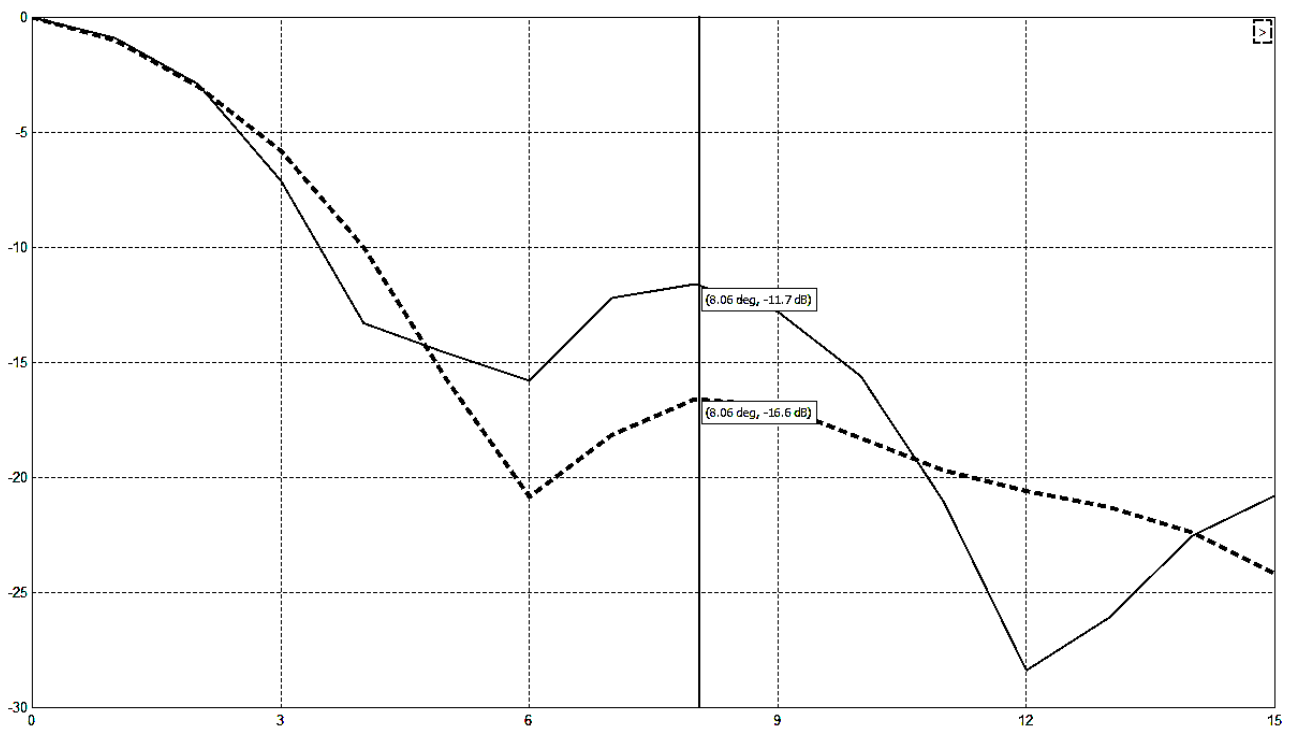


*Рис. 1. Исследуемая антенна Кассегрена с экранами*

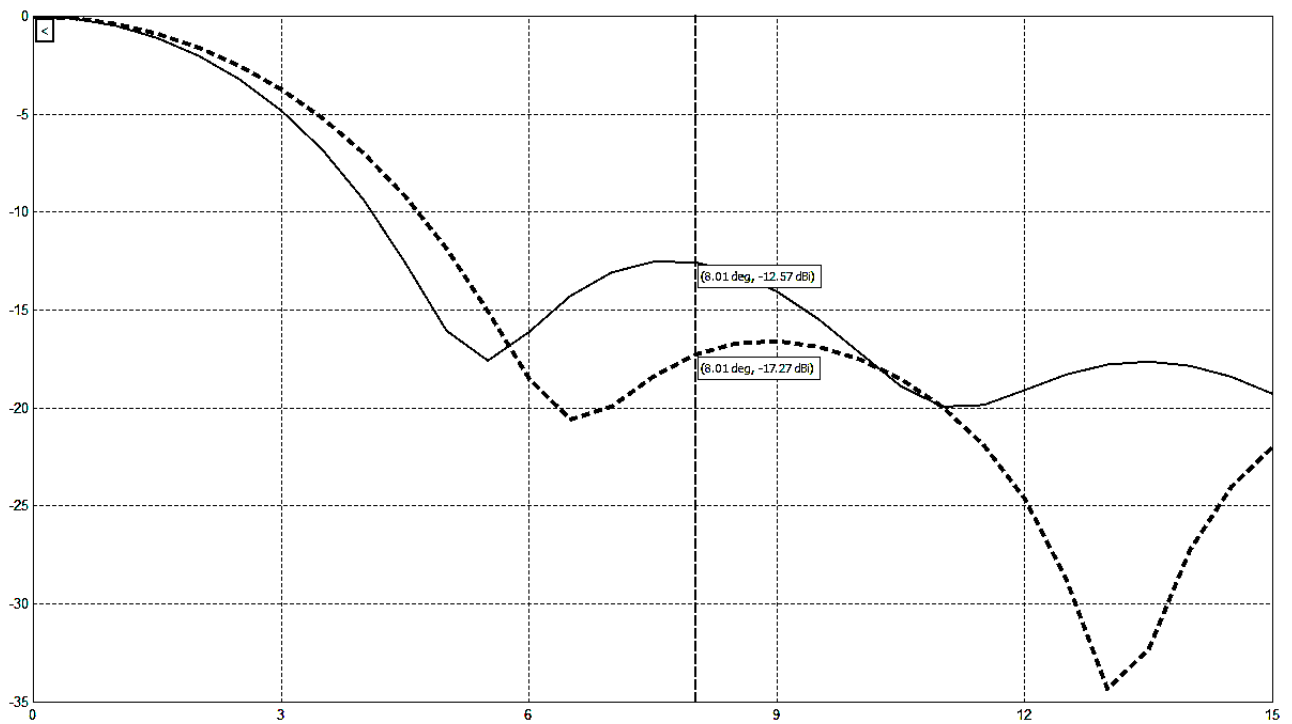
В настоящей работе в качестве инструмента теоретического исследования использовался программный пакет FEKO 7.0, оценочная лицензия которого была предоставлена автору дистрибьютором этого программного продукта в России ЗАО “НПП “Родник”. Расчет электродинамических характеристик антенны Кассегрена велся токовым методом, а сами токи определялись в пакете FEKO с помощью метода моментов. Ниже экспериментально и теоретически демонстрируется возможность подавления боковых лепестков апертурных антенн, на примере антенны Кассегрена. Подавление реализуется за счет внесения в раскрыв антенны прямоугольных экранов. Физически процесс подавления боковых лепестков можно объяснить тем, что диаграмма направленности апертурной антенны (в нашем случае Кассегрена) формируется как сумма диаграмм направленности элементарных площадей раскрыва антенны. И есть участки раскрыва антенны которые в большей степени чем остальные участвуют в формировании боковых лепестков. Их и предлагается затенить экраном, исключая их излучение из общего поля в раскрыве антенны, формирующего диаграмму направленности самой антенны. В нашем случае необходимо найти компромисс между подавлением боковых лепестков ДН, и снижением уровня мощности антенны по главному лепестку. Что и позволит получить наилучшее соотношение сигнал/шум. Для этого мы можем изменять несколько параметров: расположение экранов, ширину и длину. То есть, изменяя область затенения, мы изменяем ДН всей антенны.

На рисунке 2 показаны результаты экспериментального исследования антенны Кассегрена с экранами (пунктирная линия) и без экранов (сплошная линия), а на рисунке 3 результаты теоретического исследования антенны Кассегрена с экранами (пунктирная линия) и без экранов (сплошная линия). Сравнение этих рисунков между собой показывает, что теоретические и экспериментальные результаты между собой очень хорошо коррелируют и поэтому результаты и выводы, полученные с помощью пакета FEKO 7.0 для других типов антенн можно считать надежными и обоснованными. Также экспериментально было выявлено наилучшее положение экранов относительно кромки антенны – 6 см, 0,06λ В используемой антенне вспомогательное зеркало крепилось к главному с помощью 4 стоек, которые не позволили поставить экран в непосредственной близости от вспомогательного зеркала. Теоретические исследования показали, что если приблизить экраны к центру антенны, то это может дать еще большее подавление бокового лепестка. В данном случае первый боковой лепесток удалось подавить приблизительно на 6.4 Дб.





**Рис. 2.** Влияние экранов в апертуре антенны Кассегрена на уровень первого бокового лепестка ДН антенны (пунктирная линия – с экраном, эксперимент)



**Рис. 3.** Влияние экранов в апертуре антенны Кассегрена на уровень первого бокового лепестка ДН антенны (пунктирная линия – с экраном, теория)

## Литература

1. *Ерухимович Ю.А.* Подавление ближних боковых лепестков в апертурных антеннах, - Труды НИИР, 1980, 4, с. 80-87.
2. *Смирнов Е.В.* Применение метода ортогональных составляющих в задачах рассеяния приемных антенн. // INTERMATIC 2015 Материалы Международной научно-технической конференции “Фундаментальные проблемы радиоэлектронного приборостроения”, 1-5 декабря 2015г., г. Москва- М.:МИРЭА, 2015. Часть 5. С.274-278.
3. *Смирнов Е.В.* Об информационной составляющей поля рассеяния приемных антенн. //INTERMATIC 2014 Материалы Международной научно-технической конференции “Фундаментальные проблемы радиоэлектронного приборостроения”, 1-5 декабря 2014г., г. Москва- М.:МИРЭА, 2014. Часть 5. С.19-23
4. *Смирнов Е.В.* Исследование информационного канала взаимодействия произвольных приемных антенн. // Т-Сотт – Телекоммуникации и транспорт. 2015, том 9, № 7, С. 41-46.
5. *Смирнов Е.В.* Обобщенная антенная теорема // Т-Сотт: Телекоммуникации и транспорт. 2014. Т. 8. № 12. С. 83-87.
6. *Смирнов Е.В., Коленко Д.О., Дудрин Д.А.* Исследование характеристик группового облучателя гибридной зеркальной антенны // Т-Сотт: Телекоммуникации и транспорт. 2014. Т. 8. № 11. С. 88-94.

# ПРИМЕНЕНИЕ ТЕХНОЛОГИИ SDR НА ОСНОВЕ PXI

Чупеев А.Р.,  
МТУСИ, студент группы ЗРС1101  
Симонов П.И.,  
МТУСИ, к.т.н.

Радиосистема – это любое устройство, которое обеспечивает передачу или прием сигнала беспроводным способом в радиочастотной области электромагнитного спектра для обмена информацией. В современном мире радиосистемы присутствуют во множестве устройств, таких как телефоны, компьютеры, дверные замки автомобилей, телевизоры, спутниковые модемы и т.д.

Все существующие радиосистемы можно поделить на 2 группы: аппаратно-характеризуемые радиосистемы, в которых параметры приемника (или передатчика) определяются схемотехническими решениями, и программно-характеризуемые радиосистемы (*SDR, Software Defined Radio*), которые имеют другой принцип работы, что отражается в их названии — это программно-характеризуемая радиосистема, то есть параметры системы определяются программным обеспечением, хотя они и ограничены аппаратными ресурсами. Программные изменения касаются диапазона частот, типа модуляции, выходной мощности, но не ограничиваются этими параметрами.

Традиционно в радиосистемах функции физического уровня (фильтрации, модуляции/демодуляции, преобразование спектра, усиление сигналов) выполнялись аппаратными методами, и любые изменения в системе могли быть внесены только непосредственным физическим вмешательством в схему устройства. Это приводило к высокой стоимости изделий и ограниченности в поддержке нескольких стандартов сигналов. Программно-определяемые радиосистемы получаются эффективнее и дешевле. Такие устройства могут иметь несколько режимов работы в различных диапазонах частот и большой набор дополнительных функций за счет способности к обновлению программного обеспечения.

*SDR* объединяет аппаратные и программные технологии, где все или некоторые из работающих функций настраиваются (изменяются) по средствам программного обеспечения. Такие устройства обычно могут содержать программируемые логические интегральные схемы (*FPGA*), цифровые сигнальные процессоры (*DSP*), процессоры общего применения (*GPP*), программируемые системы на кристалле (*SoC*). Использование этих технологий позволяет изменять и расширять функциональные возможности радиосистем без аппаратного вмешательства в схему.

Принцип работы *SDR* основывается на оцифровке принятого радиосигнала и дальнейшей его обработке уже в цифровой форме. Среди вариантов построения *SDR* приемника можно выделить 2 группы, использующие возможности ПК, и автономные системы, в которых обработка и прием сигналов происходят в одном законченном устройстве.

Рассмотрим примерную архитектуру сотового телефона, выполненного на базе технологии *SDR*. Трансиверы *SDR* реализуют свои функции путем запуска соответствующих модулей и блоков программного обеспечения на аппаратной платформе общего назначения. Аналоговая часть аппаратного обеспечения для выполнения таких функций, как, например, подстройка частоты, фильтрация, модуляция и демодуляция, заменяется программным кодом, который реализует эти функции в «цифре». Такое перераспределение позволяет одному радиоустройству перепрограммировать свои смесители и фильтры для поддержки большого количества схем модуляции, что позволит работать ему во множестве различных диапазонов частот.

В обычном сотовом телефоне (рис. 1) в аналоговом режиме исходящие сигналы выходят из аналогового сигнального процессора и подаются на последовательность функциональных блоков такой же конструкции, какие применяются в большинстве радио- и телевизионных станций — супергетеродинной архитектуры. Из сигнального процессора несущая модулируется сигналом и переносится на промежуточную частоту, а затем повышается до более высокой частоты, после чего усиливается и подается на антенну.

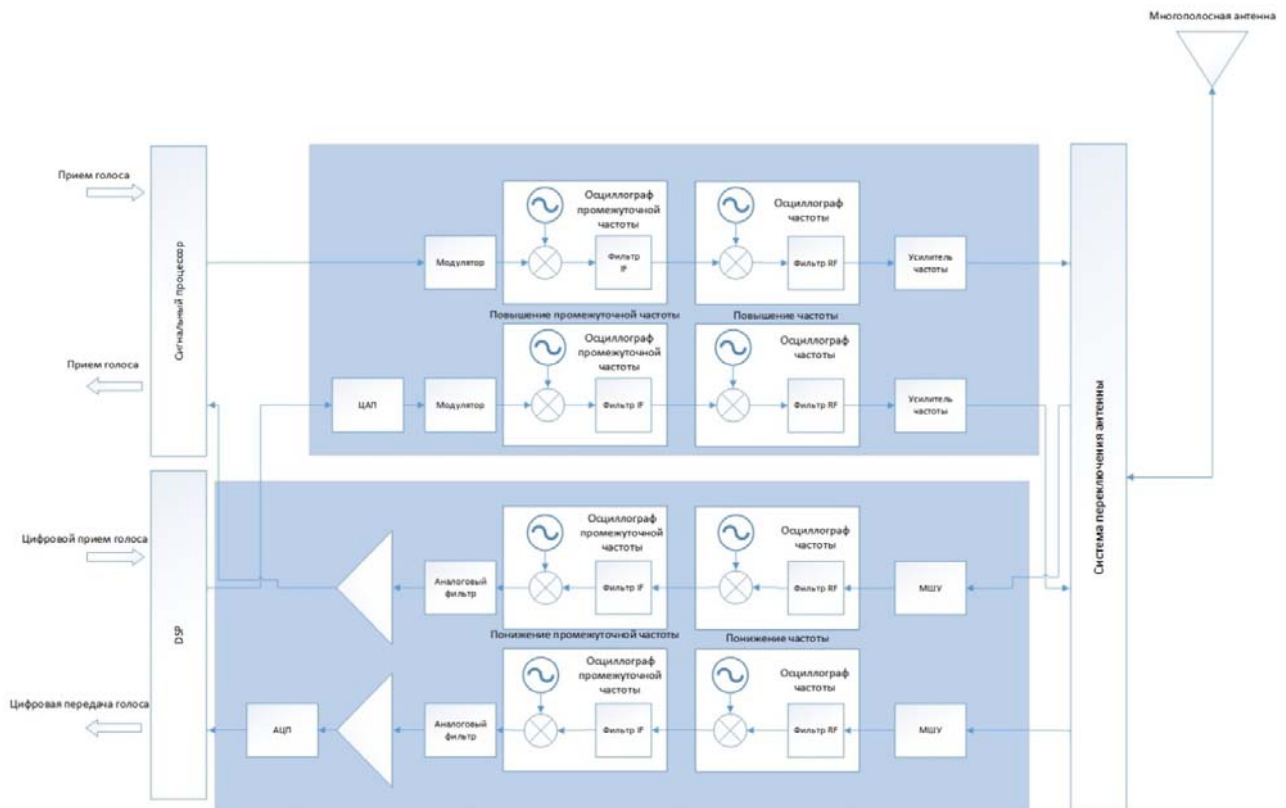


Рис. 1. Архитектура двухрежимного сотового телефона

В обратном направлении аналоговый приёмник понижает принятый аналоговый сигнал в двух каскадах, выбирает соответствующий канал и затем подает его на сигнальный процессор для демодуляции. Цифровой трансивер работает аналогичным образом — за исключением того, что вместо аналогового используется цифровой сигнальный процессор. При этом могут также использоваться операции декомпрессии, шифрования и фильтрации.

В сущности, сотовый телефон имеет конфигурируемые функции с множеством программных ядер, которые активируются и деактивируются при необходимости. Технология *SDR* предлагает более элегантное решение, использующее программируемые цифровые сигнальные процессоры, которые сначала загружают и затем запускают те функциональные блоки, которые необходимы для реализации поддержки конкретного стандарта.

Первым шагом на пути преобразования традиционного сотового телефона к системе *SDR* является преобразование как можно большего числа узлов из аналоговых в цифровые. То есть все аналоговые *baseband*-цепи должны быть заменены на цифровые. На стороне передатчика это означает преобразование в цифровой вид речи как можно ближе к микрофону, все операции обработки далее (сжатие, фильтрация, модуляция) выполняются цифровыми цепями. Аналогичные преобразования требуются и в приёмной части телефона. Здесь стоит задача преобразовать входной радиочастотный сигнал в цифровую форму как можно ближе к антенне для дальнейшей обработки в программируемых устройствах, и затем преобразовать его обратно в аналоговую форму как можно ближе к динамику телефона.

Следующим шагом является использование аналого-цифровых и цифро-аналоговых преобразователей для того, чтобы достичь программируемости на высоких частотах. *SDR*-трансивер требует меньше половины всего аппаратного обеспечения традиционного сотового телефона. Вместо дублирования цепей передачи и приёма программное радио *SDR* имеет всего одну цепочку, параметры которой настраиваются в соответствии с конкретными требованиями. Все функции определяются программно, поддерживается множество стандартов при использовании одного приемника и передатчика. Приемник имеет аналоговый оконечный каскад, при этом выход ПЧ преобразуется в цифровую форму и затем подвергается обработке в ЦСП. Выбор канала осуществляется переключением фильтров, эту операцию реализует ЦСП.

Прямой путь реализации SDR — использовать архитектуру прямого преобразования, к которой аналоговые цепи напрямую понижают радиочастотный сигнал, пропуская этап промежуточной частоты (рис. 2). Затем этот сигнал оцифровывается при помощи аналого-цифрового преобразователя, после чего требуемый канал может быть выбран путем подключения соответствующего фильтра. Такая функциональность может быть реализована в цифровом сигнальном процессоре. Такая архитектура позволяет поддерживать более широкий диапазон частот и полос пропускания за счет прямого преобразования и отсутствия каскадов промежуточной частоты.

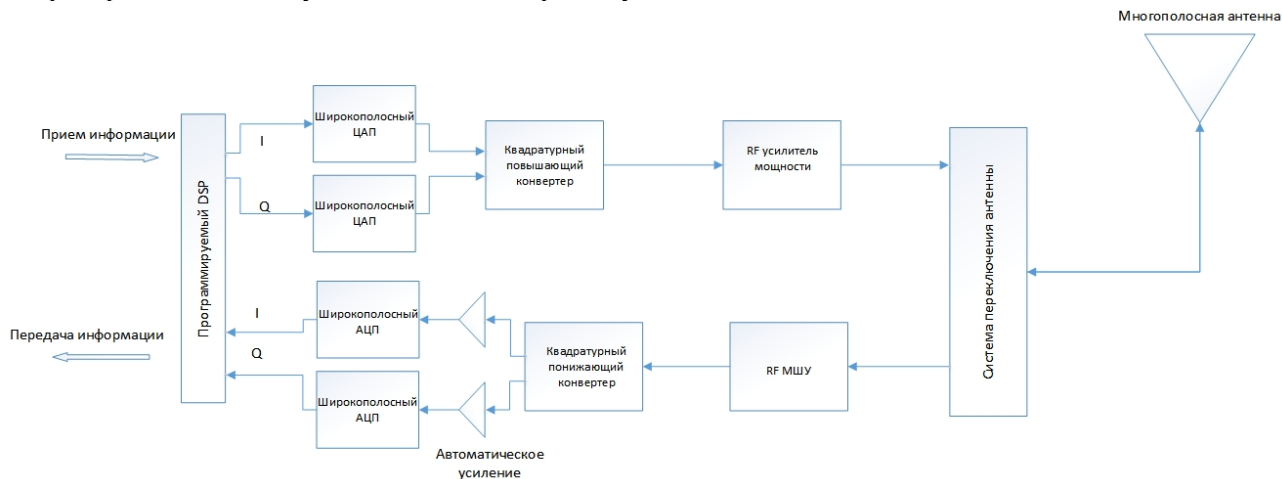


Рис. 2. Архитектура сотового телефона с прямым преобразованием

Можно выделить четыре компонента, параметры которых являются критически важными для систем SDR и фактически определяют их характеристики: это АЦП, ЦСП, фильтры и радиочастотные усилители. АЦП представляет собой наиболее критичный компонент в системе SDR, поскольку его быстродействие влияет на то, насколько близко антенна должна быть к самому АЦП. Другим важным компонентом устройств SDR является ЦСП. Несмотря на постоянный рост производительности, для некоторых систем может оказаться более выгодным использование параллельной обработки цифровых данных несколькими ЦСП. Однако такой подход приводит к увеличению размеров, веса, рассеиваемой мощности и стоимости, поэтому прибегать к нему стоит лишь в случае крайней необходимости. Фильтры критически важны в системах SDR, поэтому необходимо тщательно подходить к их выбору. Неплохим решением является использование отдельных микросхем цифровых фильтров. Эти устройства могут осуществлять функции фильтрации сигнала при низкой сложности и небольшой стоимости. В отличие от варианта реализации функций фильтрации на ЦСП, этот вариант не является программируемым, но, тем не менее, поскольку частотные диапазоны для стандартов сотовой телефонии являются фиксированными, их использование в конструкциях сотовых телефонов SDR вполне оправдано.

Отличительной особенностью систем SDR является возможность создания и описания фильтров, используя различные языки, например, язык *LABView*, который является наиболее используемым средством написания инструментария для систем *PXI/PXIe*. Данный язык позволяет использовать как готовые *DSP*, *FIR*, *IIR* фильтры, так и написать их самостоятельно. Имеется возможность описания фильтров целиком на ПК, так и с обработкой фильтра в системах ПЛИС.

На базе платформы *PXI* для программируемого радио можно создать системы самого разного назначения: модуляторы/демодуляторы сигналов, системы мониторинга спектра, системы радиолокации/радионавигации, системы РЭБ, комплексные устройства имитации базовых станций связи. Разработка приложений для систем программируемого радио на базе *PXI* ведется в среде *LabVIEW* с использованием специального набора библиотек для программирования ПЛИС (*LabVIEW FPGA* и *FPGA RF Communications library*).

*PXI* системы, использующие технологию SDR, можно собрать самостоятельно, либо воспользоваться готовыми решениями от компаний-поставщиков *PXI*, например, компании *National Instruments*.

В сравнении с традиционными средствами, *SDR* дает следующие очевидные преимущества:

1. Уменьшенные масса и габариты изделия.
2. Уменьшение потребляемой мощности.
3. Упрощение конструкции.
4. Уменьшение стоимости.
5. Масштабируемость решения.

### **Литература**

1. Что такое программно-определяемое радио (SDR)? [Электронный ресурс]: интернет-статья / Элтех – Интернет – сайт – 2014. - . – URL: [http://www.eltech.spb.ru/article/что\\_такое\\_programmno\\_opredelyaemoe\\_radio\\_sdr](http://www.eltech.spb.ru/article/что_такое_programmno_opredelyaemoe_radio_sdr)
2. SDR-приемник с нуля [Электронный ресурс]: интернет-статья / Habrahabr – Интернет-сайт – 2013. - . – URL: <https://habrahabr.ru/sandbox/60229/>
3. Технология Software Defined Radio. Теория, принципы и примеры аппаратных платформ [Электронный ресурс]: интернет-статья / Беспроводные технологии – Интернет-сайт – 2007. - . – URL: <http://www.wireless-e.ru/articles/technologies/200>

# ИССЛЕДОВАНИЕ СПОСОБОВ ФОРМИРОВАНИЯ ПОСЛЕДОВАТЕЛЬНОСТЕЙ ЗАДОВА-ЧУ

Ворожищев И.В.,  
МТУСИ, студент группы БРМ1202  
Минаева О.Н.,  
МТУСИ, старший преподаватель

Последовательности Задова-Чу, также известные как ЛЧМ-подобные последовательности (Generalized Chirp-Like), относятся к последовательностям с постоянной амплитудой и нулевой автокорреляцией (Constant Amplitude Zero Autocorrelation).

Последовательности Задова-Чу рассчитываются по формулам:

$$y_U(n) = e^{-\frac{2\pi i}{P} \cdot U \frac{n(n+1)}{2}} \text{ для нечётных } P$$

$$y_U(n) = e^{-\frac{\pi i}{P} \cdot U n^2} \text{ для чётных } P, \text{ где}$$

$P$  – дина последовательности

$U$  – индекс последовательности

$n$  – элемент последовательности

Такие последовательности набирают популярность в современных стандартах подвижной связи по причине некоторых свойств, выгодно отличающих их от псевдошумовых последовательностей, широко используемых для синхронизации в системах подвижной связи:

Постоянная амплитуда последовательности и её ДПФ.

Ограничивает пик-фактор сигнала и его влияние на сигналы других пользователей, а также снижает требования к тракту передачи;

«Идеальная» циклическая автокорреляция.

Корреляция последовательности с её циклически сдвинутой копией представляет собой дельта-функцию, что позволяет получить из одной последовательности несколько ортогональных;

Абсолютное значение циклической взаимной корреляционной функции между любыми последовательностями Задова-Чу постоянно и обратно квадратному корню длины последовательности, если модуль разности корней последовательности — относительно простое число по отношению к её длине;

Дискретное преобразование Фурье последовательности Задова-Чу является последовательностью Задова-Чу.

Это свойство позволяет создать последовательность в частотной области без применения ДПФ. [1, с. 145-146]

На рисунках 1-3 приведены графики, иллюстрирующие перечисленные выше свойства при длине последовательности 139:

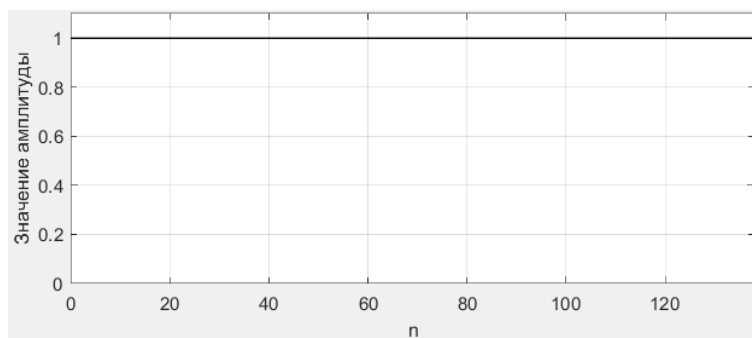
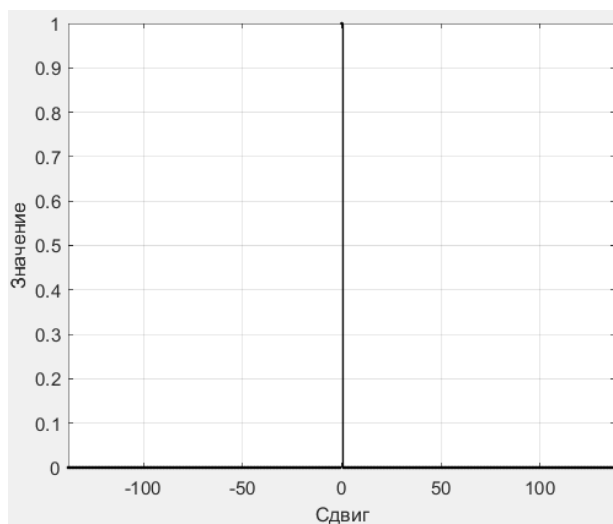
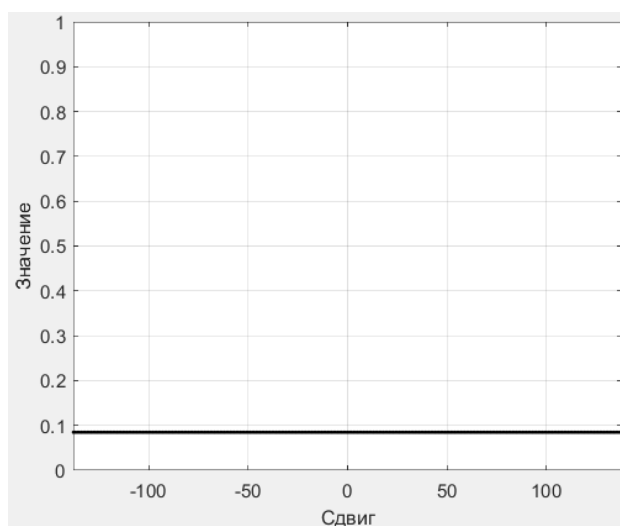


Рис. 1. Амплитуда последовательности Задова-Чу



*Рис. 2. Циклическая АКФ последовательности с индексом 25*



*Рис. 3. Циклическая ВКФ последовательностей с индексами 25 и 29*

Последовательности Задва-Чу широко применяются в стандарте 3GPP LTE. Они используются в первичном сигнале синхронизации (Primary Synchronization Signal), зондирующем и демодуляционном опорных сигналах канала связи «вверх» (Uplink Sounding Reference Signal и Uplink Demodulation Reference Signal) и преамбуле произвольного доступа (Random Access Channel Preamble). [1, с. 145]

Первичный сигнал синхронизации отвечает за синхронизацию мобильной станции с базовой со стороны базовой станции. В нём используются три последовательности с индексами 29, 34 и 25, так как такие последовательности обладают хорошими периодическими АКФ и ВКФ, а также низкой чувствительностью к сдвигу частоты. [1, с. 147-150]

Зондирующий и демодуляционный опорные сигналы канала связи «вверх» используются для оценки качества канала связи. Длина используемых последовательностей зависит от количества используемых поднесущих. [1, с. 360]

Преамбула произвольного доступа отвечает за синхронизацию мобильной станции с базовой со стороны мобильной станции. В зависимости от формата преамбулы используются последовательности длиной 139 или 839, индексы которых также зависят от формата преамбулы. [1, с. 434-435]

Наиболее простой метод. Элементы последовательности заранее рассчитываются и записываются в память устройства при производстве устройства или на этапе его инициализации: во время работы устройства нужный элемент не рассчитывается, а считывается из памяти. Таким образом,



операция вычисления заменяется простой операцией поиска, что увеличивает скорость формирования последовательности в процессе работы устройства.

Элементы таблицы рассчитываются по каноническому уравнению:

$$y_U(n) = e^{-\frac{2\pi i}{P} y_U(n)}, \text{ где}$$

$$Y_U(n) = U \frac{n(n+1)}{2} \text{ для нечётных } P$$

$$Y_U(n) = U \frac{n^2}{2} \text{ для чётных } P$$

Данный метод был предложен Срджаном Будишиным в 2010 году в [2] для последовательностей с длиной, которая является простым числом. Предложено альтернативное выражение для последовательностей Задова-Чу, согласно которому

$$y_U(n) = e^{-\frac{2\pi i}{P} x_U(n)}, \text{ где}$$

$$X_U(n) = U \frac{n(n+P)}{2} \text{ mod } P$$

Это выражение устанавливает связь последовательностей Задова-Чу с теорией квадратичных вычетов.

Основное отличие последовательностей, полученных с помощью предложенного выражения, от канонической формулы — ограниченность количества наборов коэффициентов  $X_U(n)$ . В то время как набор  $Y_U(n)$ , полученный с помощью канонической формулы, уникален для каждого индекса  $U$ , последовательности, полученные с помощью предложенного выражения, имеют всего два уникальных набора.

Согласно [2], любая последовательность с индексом

$$U = K^2 \text{ mod } P$$

может быть получена децимацией с шагом  $K$  последовательности с индексом  $U = 1$ :

$$X_U(n) = X_1(Kn) \text{ mod } P$$

Децимация не требует повторного вычисления и является по своей сути просто перестановкой элементов уже рассчитанной последовательности.

Индекс  $U$ , удовлетворяющий условию, является квадратичным вычетом по модулю  $P$ . Так как не все индексы соответствуют этому условию,  $K$  ограничивается диапазоном значений  $K = 1, 2, \dots, (P-1)/2$ .

Если индекс не удовлетворяет условию, то он является квадратичным невычетом. Последовательность с таким индексом может быть получена децимацией с шагом  $K$  последовательности с индексом, равным  $V$ , если существует такое  $V$ , что

$$U = VK^2 \text{ mod } P$$

В таком случае:

$$X_U(n) = X_V(Kn) \text{ mod } P$$

В качестве  $V$  можно использовать любой невычет по модулю  $P$ , а вместо последовательности с индексом 1 можно использовать любой индекс, являющийся вычетом по модулю  $P$ .

Получение последовательностей децимацией действительно только для предложенного выражения, так как последовательности, полученные с его помощью, являются перестановками набора элементов двух последовательностей. Очевидно, что последовательности, полученные по канонической формуле, таким свойством не обладают.

Таблица, достаточная для получения последовательностей Задова-Чу длиной 139 методом децимации, имеет следующий вид (Таблица 1)

Сравнение было выполнено на симуляторе ПЦС ADSP-BF506F, встроенном в среду разработки VisualDSP++. На рисунке 5 изображены зависимости времени выполнения программ от количества последовательностей длины 139, формируемых обоими методами. Непрерывно рассчитывались последовательности равной длины двадцати пяти разных индексов.

Таблица, необходимая для реализации метода децимации

$U/n$	0	1	2	3	4	5	6	...	137	138
Вычет/невывет	-	В	Н	Н	В	В	В	...	В	Н
$K$	-	1	1	40	2	12	59	...	50	25
Исходная последовательность $y_1(n)$	$W^0$	$W^{70}$	$W^2$	$W^{74}$	$W^8$	$W^{82}$	$W^{18}$	...	$W^2$	$W^{70}$
Исходная последовательность $y_2(n)$	$W^0$	$W^1$	$W^4$	$W^9$	$W^{16}$	$W^{25}$	$W^{36}$	...	$W^4$	$W^1$

$$W = e^{-\frac{2\pi i}{P}}$$

Был разработан алгоритм, реализующий метод децимации:

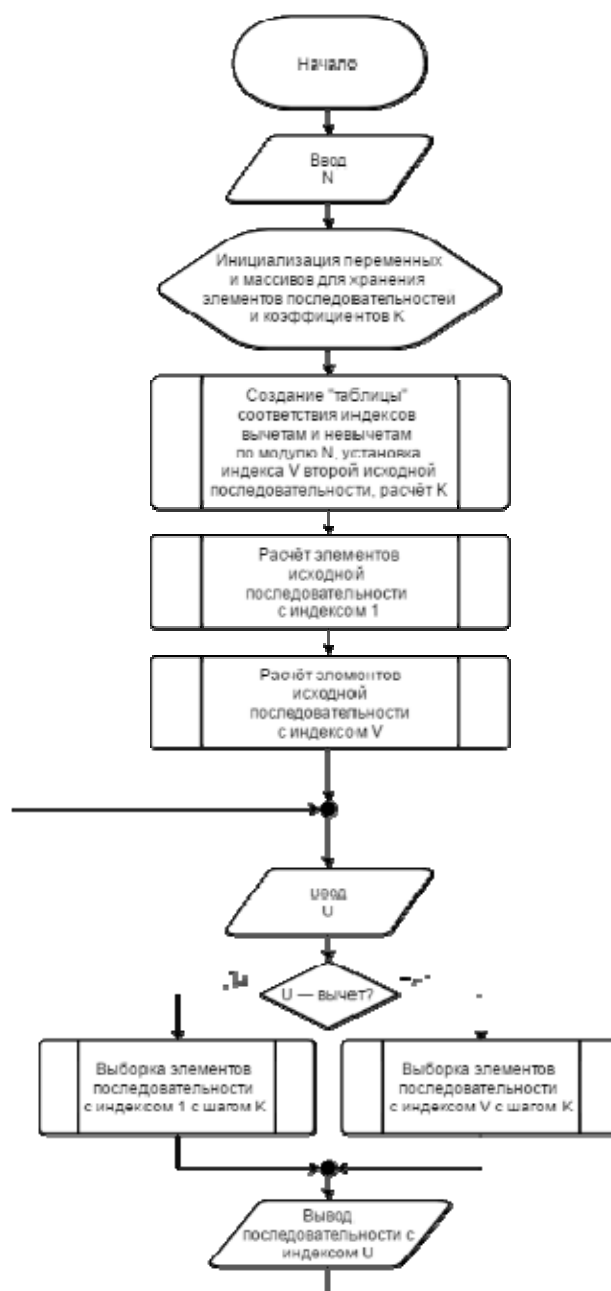


Рис. 4. Обобщённая блок-схема алгоритма

Согласно результатам, методу децимации требуется около 2,55 мс на инициализацию (составление таблицы соответствия и расчёт двух исходных последовательностей), после чего на расчёт каждой последовательности уходит 20-25 мкс. На расчёт каждой последовательности длиной 139 по канонической формуле уходит около 260 мкс.

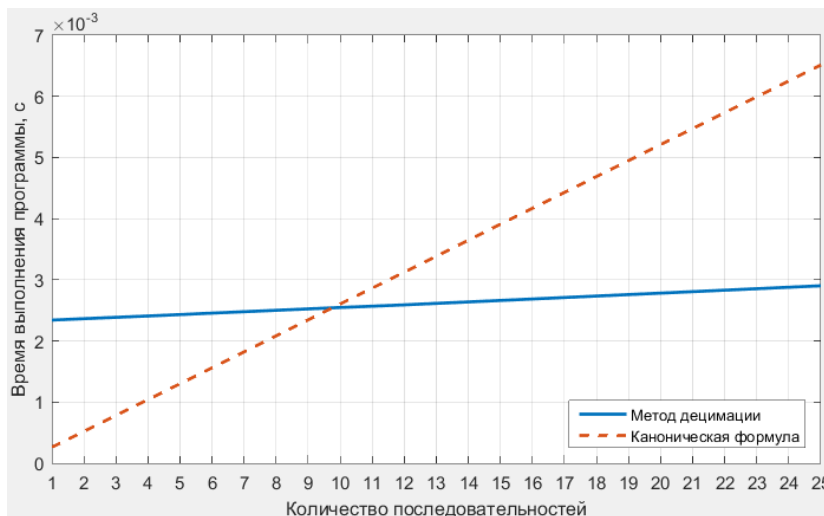


Рис. 5. График зависимости времени выполнения программы от количества последовательностей

Уже после десятой последовательности метод децимации показывает преимущество в скорости, а с учётом времени генерирования одной последовательности этот метод становится весьма полезным в случаях, когда необходимо в реальном времени формировать множество последовательностей разных индексов одинаковой длины.

### Литература

1. Sesia S., Toufik I., Baker M. LTE – The UMTS Long Term Evolution: From Theory to Practice. — Torquay, UK: John Wiley & Sons, 2009.
2. Budisin S. Decimation Generator of Zadoff-Chu Sequences // Sequences and Their Applications – SETA 2010, 6th International Conference, Paris, France, September 13-17, 2010. — с. 30–40.

# БЕСПРОВОДНЫЕ СЕНСОРНЫЕ УСТРОЙСТВА НА ОСНОВЕ ЭФФЕКТА ПЕРЕИЗЛУЧЕНИЯ

Славянский А.С.,  
МТУСИ, студент группы МИТ1502  
Сперанский В.С.,  
МТУСИ, к.т.н., профессор

Большую часть используемых сенсорных устройств занимают устройства имеющие «классическую» структуру, т.е. имеют тракт приема и тракт передачи. Значительно меньшую часть сенсорных устройств представляют устройства, принцип работы которых основан на эффекте «переизлучения».

Физический смысл эффекта «переизлучения» аналогичен «рассеиванию» в радиолокации и заключается в том, что при изменении электропроводности, диэлектрической или магнитной проницаемости объекта, он становится источником вторичного излучения.[3] Изменение электропроводности цепи показано на рис. 1а.

При подаче на антенну логической 1, нагрузочное сопротивление  $R_n$  принимает очень малое значение ( $R_n \rightarrow 0$  [короткое замыкание]). В промежутки времени когда подается логический 0,  $R_n$  принимает большое значение ( $R_n \rightarrow \infty$  [холостой ход]). Нагрузочным сопротивлением управляет выходной транзистор микросхемы (ключевой режим), при этом антенна выполняет роль модулятора. При использовании прямоугольных импульсов в качестве управляющего сигнала, наблюдается АМ колебание с прямоугольной огибающей. На рис. 1б представлена временная диаграмма тока протекающего по антенне.

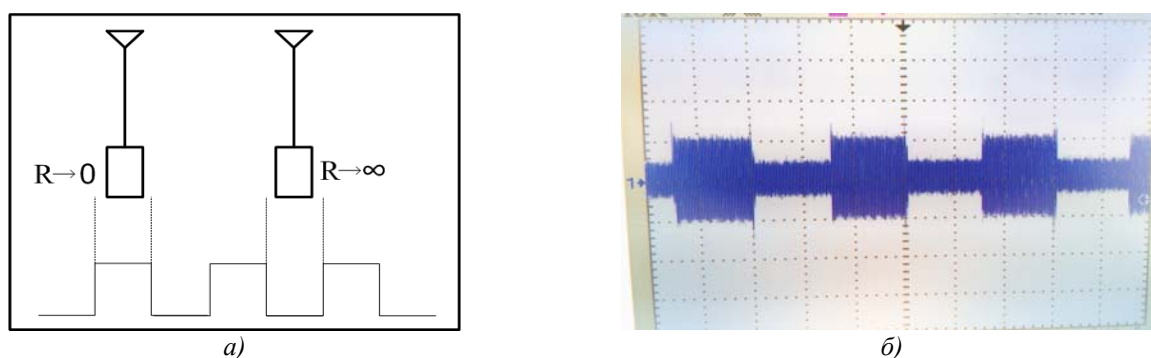


Рис. 1. а – физический смысл эффекта переизлучения;  
б – временная диаграмма тока протекающего по антенне

На рис. 2 приведены принципиальные схемы «полуактивного» и «пассивного» устройств.[1]

В схеме «пассивного» датчика, в качестве источника питания используются физические источники тока (ФИТ). ФИТ – устройства преобразующие воздействия окружающей среды (температура, давление и т.д.) в электрический ток

Помимо отличия по способу питания схем, есть различие в режиме работы. «Полуактивные» датчики позволяют производить съем телеметрической информации непрерывно во времени, путем изменения времязадающей емкости, что в свою очередь приводит к изменению частоты выходных импульсов (время-импульсная модуляция) или время задающего сопротивления, что приводит к изменению скважности выходных импульсов (широтно-импульсная модуляция). В свою очередь «пассивные» устройства позволяют регистрировать лишь превышение измеряемой величиной (температура, давление и т.д.) некоторого уровня.

Для съема того или иного параметра телеметрической информации необходимо применять соответствующие «измерительные» элементы. В роли таких элементов выступают радиокомпоненты изменяющие свои параметры в зависимости от воздействия окружающей среды, пример таких элементов приведены в таблице 1. «Измерительные» элементы имеют

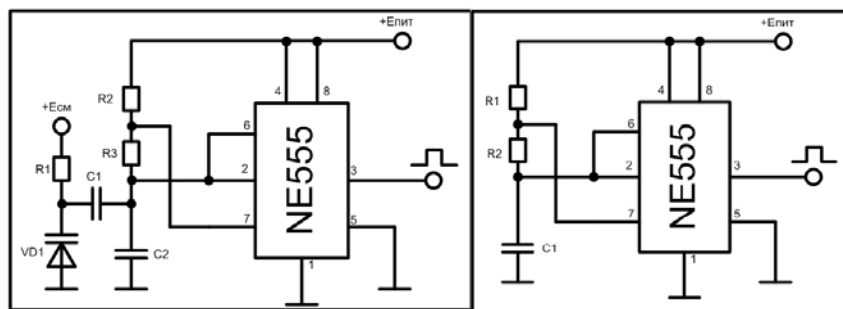


Рис. 2. Принципиальные схемы «полуактивного» и «пассивного» устройств

Таблица 1

**«Измерительные» элементы применяемые в «полуактивных» и «пассивных» устройствах**

Измеряемая величина	Название элемента	ФИТ
Температура	Модуль Пельте	+
	Терморезистор	-
	Термоконденсатор (Вари-конд)	-
Освещенность	Фоторезистор	-
	Фотоконденсатор	-
Давление	Полупроводниковый тензومتر	-
	Бароэдс	+
Вибрации	Модуль MEMS	+
	Пьезоэлемент	-

Как и в любой радиоаппаратуре немаловажным условием при эксплуатации устройств, принци, является соблюдение электромагнитной совместимости.

На рис. 3а,б представлены возможные организации каналов при работе на низких и высоких частотах. Причиной таких конфигураций канала является то, что реализовать миниатюрную низкочастотную антенну с узкой диаграммой направленности (ДН) весьма затруднительно, в отличии от высокочастотной.

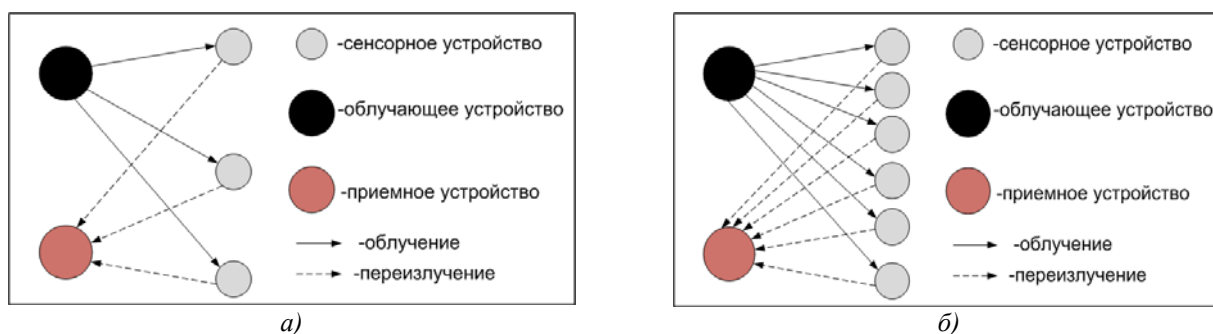
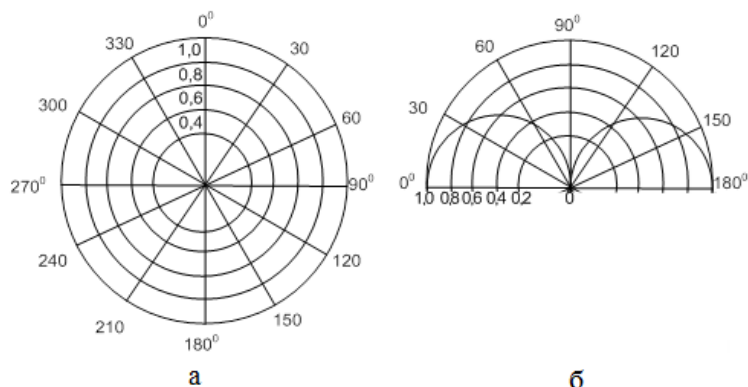


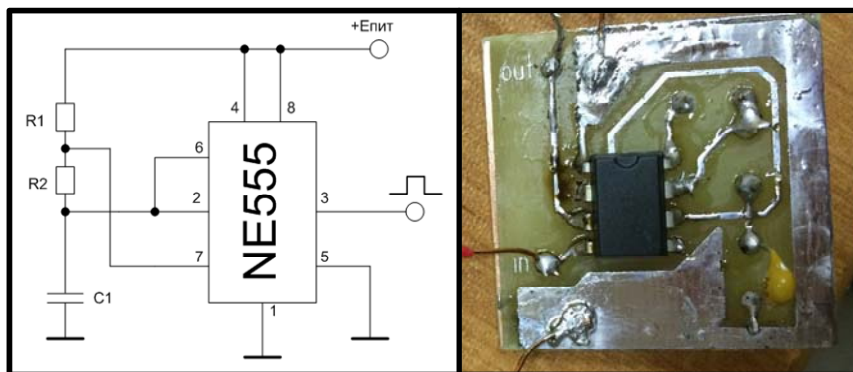
Рис. 3. Организация канала при работе на: а – НЧ; б – ВЧ

В рамках выполнения работы были проведены эксперименты по наблюдению эффекта «переизлучения». Для эксперимента использовались  $\lambda/4$  несимметричные вибраторы рассчитанные на частоту 199 МГц ( $\lambda=150.65$  см;  $L= \lambda/4\sim 37.7$  см), диаграммы направленности таких антенн представлены на рис. 4.



**Рис. 4.** ДН  $\lambda/4$  вибратора: а – в горизонтальной плоскости; б – в вертикальной плоскости

На рисунке 5 изображены принципиальная схема и макет сенсорного устройства используемого при проведении эксперимента. В качестве источника питания использовался лабораторный блок питания.



**Рис. 5.** Принципиальная схема и макет сенсорного устройства

Эксперимент состоял из 3х частей:

I. Наблюдение эффекта «переизлучения» [рис. 6, 7а, б];

Оборудование обозначенное на рис. 6, 8а,10.

(1)- Генератор ВЧ синусоидальных колебаний [Agilent 8648C] (облучающая антенна), F=199 МГц.

(2)- Генератор прямоугольных импульсов (облучаемая антенна), F=30 кГц.

(3)- Измерительное оборудование [Tektronix TDS 2022 / Agilent technologies N9340B] (осциллограф / анализатор спектра).

(4)- Индуктивный пробник, при помощи которого производились измерения тока протекающего по антенне, и наблюдение амплитудной модуляции.

Расстояние между антеннами  $\sim 10$  М.

II. Моделирование системы передачи телеметрической информации [рис. 8а,б, 9а,б];

III. Преднамеренное нарушение организации канала при работе на НЧ [рис. 10, 11а,б,в];

В таблице 2 представлены значения спектральных составляющих в контрольных точках.

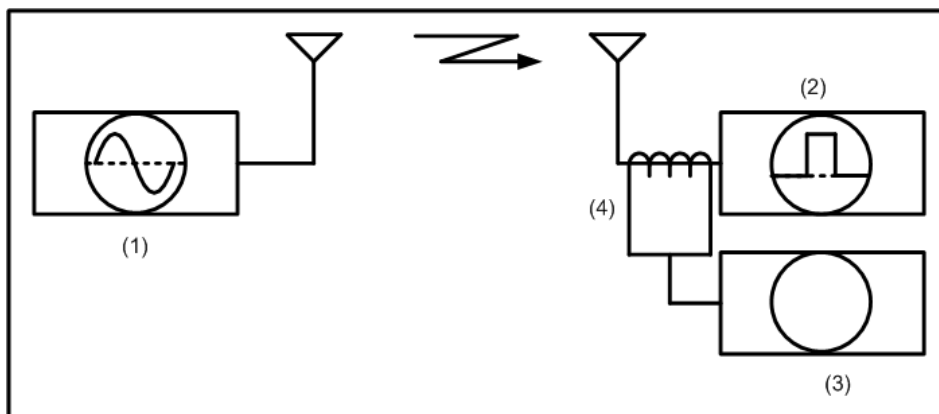


Рис. 6. Схема эксперимента по наблюдению эффекта «переизлучения»

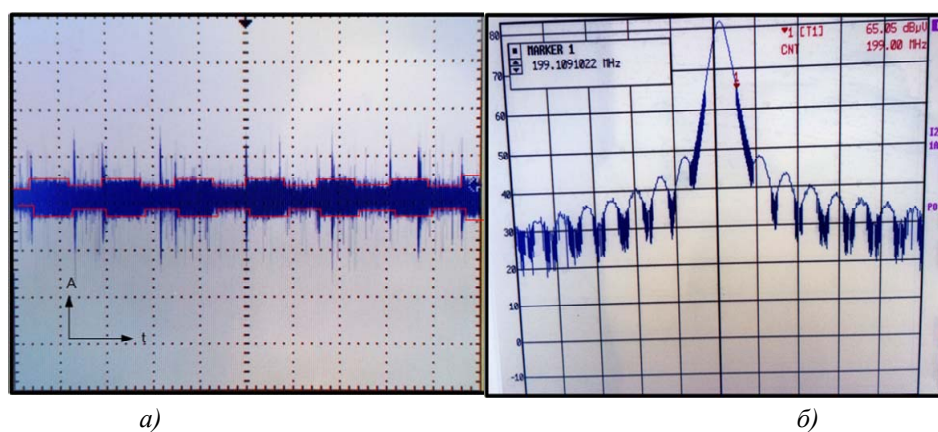


Рис. 7. а – временная диаграмма тока протекающего по антенне; б – спектральная диаграмма

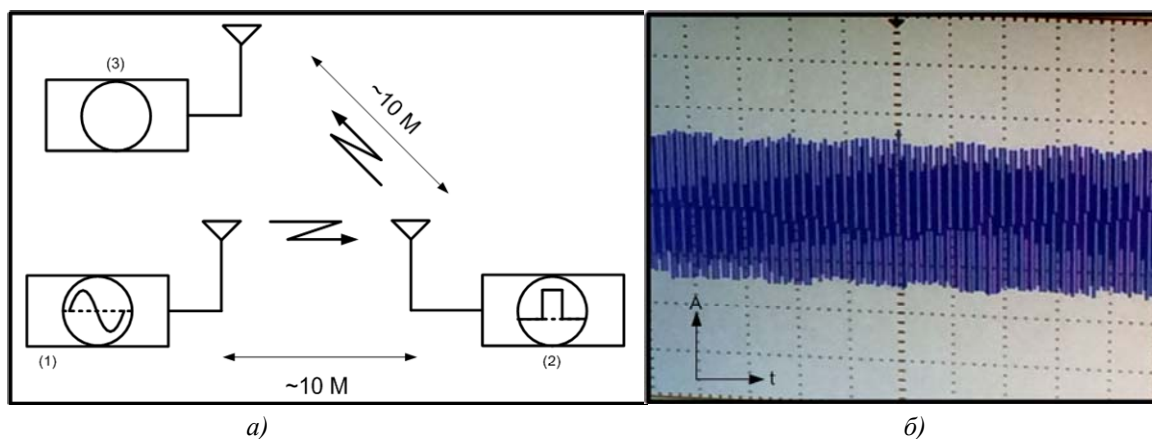
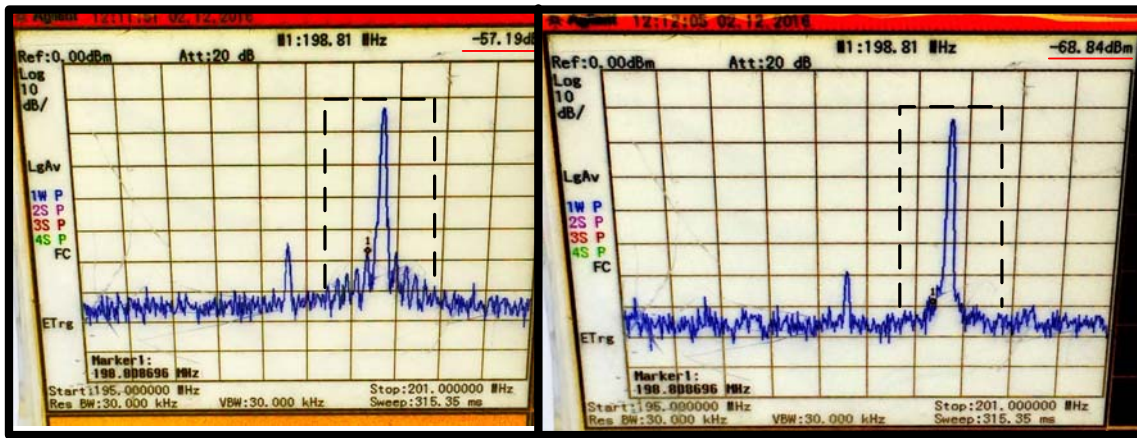


Рис. 8. а – схема физического моделирования системы передачи телеметрической информации; б – временная диаграмма полученная на измерительном устройстве ((3) – осциллографе)

На рисунках 11 а, б, в видно, что при размещении 2х сенсорных устройств в достаточной близости друг от друга, происходят постоянные изменения (так называемое «дыхание спектра») в полосе частот занимаемой переизлученным сигналом, что при использовании одного датчика не наблюдается. Поскольку оба датчика работают на одинаковой частоте, то по определению ЭМС они являются источниками взаимных помех, по этой причине получение временной диаграммы затруднительно.



а)

б)

Рис. 9. Спектральная диаграмма полученная на измерительном устройстве ((3) – анализаторе спектра):  
 а – генератор (2) включен, наблюдается спектр характерный для прямоугольных импульсов;  
 б – генератор (2)- выключен

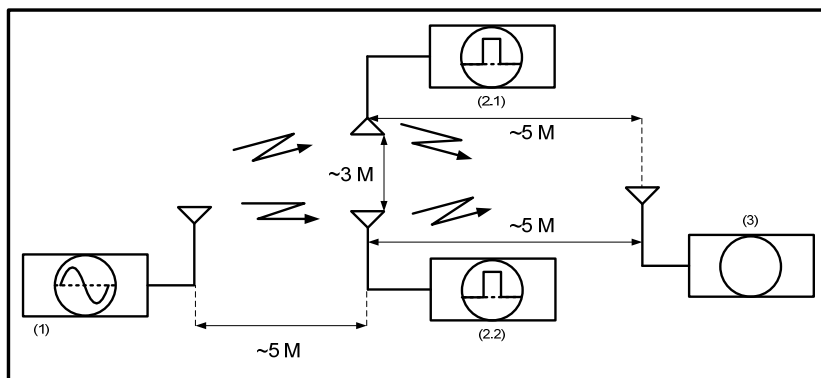
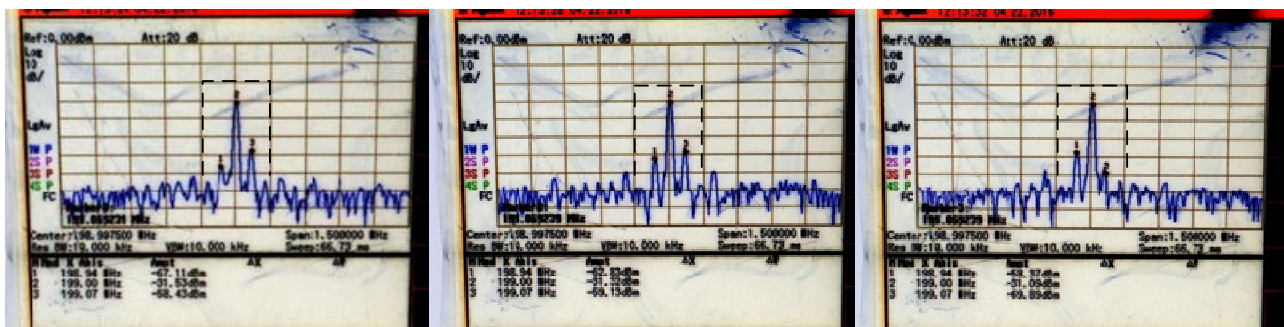


Рис. 10. Схема эксперимента «Преднамеренное нарушение организации канала при работе на НЧ»



а)

б)

в)

Рис. 11. Спектральная диаграмма полученная на измерительном устройстве ((3) – анализаторе спектра)),  
 при размещении 2х сенсорных устройств на расстоянии 3х метров друг от друга

В ходе работы был экспериментально изучен эффект «переизлучения», рассмотрены «полуактивные» и «пассивные» сенсорные устройства, так же рассмотрены возможные организации каналов и проведено экспериментальное наблюдение преднамеренного нарушения организации канала при работе на низких частотах.



### Значения спектральных составляющих в контрольных точках (маркерах)

Маркер	Частота, МГц	Значение см. рис. 11а	Значение см. рис. 11б	Значение см. рис. 11в
1	198.94	-67.11 dBm	-62.83 dBm	-58.37 dBm
2	199.00	-31.53 dBm	-31.32 dBm	-31.09 dBm
3	199.07	-58.43 dBm	-59.13 dBm	-69.89 dBm

Следующими шагами данной работы являются: повышение рабочей частоты, а, следовательно, увеличение числа сенсорных устройств, что позволит организовать сенсорную сеть для съема телеметрической информации, и экспериментальное изучение возможности внедрения сверхширокополосных сигналов.[2]

### Литература

1. Daniel M. Dobkin, The RF in RFID Passive UHF RFID in Practice.: Elsevier, 2008. 505 с.
2. Косичкина Т.П., Сидорова Т.В., Сперанский В.С. Сверхширокополосные системы телекоммуникаций. М.: Инсвязьиздат, 2008. 304 с.
3. Казаринов Ю.М. Радиотехнические системы: учебник для вузов по спец. «Радиотехника». М.: Выс. шк., 1990. С 198.
4. Сперанский В.С., Клиңцов О.И. Передача радио по волокну // Т-Comm: Телекоммуникации и транспорт. 2015. Т. 9. № 11. С. 38-40.
5. Косичкина Т.П., Сперанский В.С., Спирип А.П., Фролов А.А. Когнитивные сверхширокополосные радиосистемы как метод повышение эффективности использования радиочастотного спектра // Т-Comm: Телекоммуникации и транспорт. 2015. Т. 9. № 12. С. 37-43.
6. Сперанский В.С., Спирип А.П., Фролов А.А., Косичкина Т.П. Перспективы развития сверхширокополосных систем связи в направлении когнитивного радио // Системы синхронизации, формирования и обработки сигналов. 2015. Т. 6. № 1. С. 9-11.
7. Сперанский В.С., Клиңцов О.И. Оптические цифровые элементы обработки сигналов // Т-Comm: Телекоммуникации и транспорт. 2014. Т. 8. № 9. С. 80-82.
8. Сперанский В.С., Фролов А.А. Сложные дискретные частотные сверхширокополосные сигналы // REDS: Телекоммуникационные устройства и системы. 2014. Т. 4. № 1. С. 78-82.
9. Рахматуллин А.Ф., Сперанский В.С. Сравнительный анализ кодовых последовательностей для СШП сигналов // Т-Comm: Телекоммуникации и транспорт. 2012. Т. 6. № 9. С. 118-119.

# РАЗРАБОТКА АЛГОРИТМА СЛЕЖЕНИЯ ЗА НЕСУЩЕЙ ПРИ ИСПОЛЬЗОВАНИИ НИЗКОСКОРОСТНЫХ ТУРБОКОДОВ В ПОМЕХОЗАЩИЩЕННОЙ СИСТЕМЕ СВЯЗИ

*Толкачев М.М.,  
МТУСИ, студент группы БРА1201  
Лобов Е.М.,  
МТУСИ, к.т.н.*

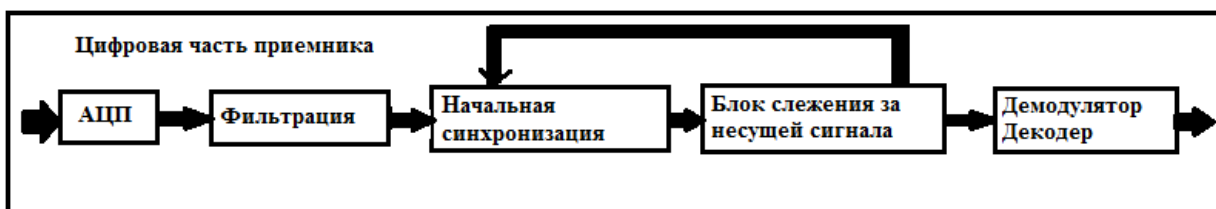
При проектировании системы приема-передачи для спутниковой радиосвязи необходимо осуществить слежение за несущей частотой сигнала при когерентном приеме с модуляцией ФМ-2 в определенных условиях. Спутниковые радиолинии характеризуются низким отношением сигнал шум (ОСШ). Для борьбы с ошибками, возникающими в данной системе, применяются помехоустойчивые сверточные коды. В данной системе используется турбо-код, описанный Международным Комитетом по Космическим Системам Передачи Данных (CCSDS). Условия работы (параметры) системы приемника указаны в таблице 1.

Таблица 1

**Параметры системы**

Тип кодера	Турбо-код (параллельный сверточный)
Относительная кодовая скорость	1/6
Число бит в пакке, количество	не более 1788
Среднеквадратичное отклонение (СКО) фазы опорного сигнала, град	4
Отношение энергии 1 бит кодируемой информации к энергетической спектральной мощности шума, дБ	$0 < E_b N_0 < 1$
Отношение энергии 1 бит информации к энергетической спектральной мощности шума, дБ	$-7,7 < E_s N_0 < -6,7$
Начальная разница между частотой сигнала и опорного сигнала, Гц	0,25
Линейное изменение несущей частоты принимаемого сигнала, Гц/с	0,25

Рассмотрим структурную схему системы, изображенную на рис. 1.



*Рис. 1. Структурная схема цифровой части приемника*

Данная схема содержит: аналогово-цифровой преобразователь, фильтр цифрового сигнала, устройство начальной синхронизаций (перестройка цифрового перестраиваемого гетеродина), устройство фазовой автоподстройки частоты (ФАПЧ), демодулятор и декодер сигнала. Модернизация будет производиться в блоке слежения за несущей частотой сигнала.

Поясним принцип работы устройства. После блоков преобразования в цифровой сигнал и его обработки (фильтрации), в результате первоначальной синхронизации, опорный генератор приемника будет настроен на несущую частоту сигнала с точностью 0.25 Гц. Линейное изменение несущей частоты сигнала (приведенное в таблице 1) в радиолинии возникает в результате движения спутника по эллиптической орбите, при этом расстояние между приемником и передатчиком будет меняться, возникает эффект Доплера. С эффектом Доплера будет бороться ФАПЧ в блоке слежения за несущей частотой сигнала. Скорректированный сигнал далее поступает на демодулятор и декодер.

Для применения когерентного приема необходимо реализовать систему синхронизации (слежение за несущей частотой сигнала). Для обеспечения синхронизации, как указано в структурной схеме на рисунке 1, используется 2 блока: блок вхождения в синхронизацию и блок поддержания состояния когерентного приема. Блоком поддержания состояния когерентного приема является система ФАПЧ. При линейно меняющейся несущей частоте сигнала в 0,25 Гц/с допустимую ошибку по фазе в 3-5 градуса обеспечивает система 2 порядка. Расчет и построение системы более высокого порядка позволит свести ошибку к 0, но это требует более сложных вычислений.

Произведем расчет коэффициентов петлевого фильтра. Для начала воспользуемся расчетными формулами, приведенными в [3], и переведем значение ОСШ (1 - 2).

$$EbN0 = 10^{\frac{EbN0_{dB}}{10}} = 10^{\frac{0.5}{10}} = 1.122 \quad (1)$$

$$EsN0 = EbN0 * r = 1.122 * \frac{1}{6} = 0.187 \quad (2)$$

Где  $r$  - относительная кодовая скорость,  $EbN0_{dB}$  - ОСШ для 1 бит кодируемой информации [дБ],  $EbN0$  - ОСШ на 1 бит кодируемой информации,  $EsN0$  - ОСШ на 1 бит символа. Зная число бит в пакете определим символьную скорость (3).

$$R_s = \frac{N}{r} = \frac{1788}{\frac{1}{6}} = 10728 \text{ бод} \quad (3)$$

Из полученных данных произведем расчет основных параметров ФАПЧ - шумовую полосу фильтра (4), собственную частоту (5) и резонансную частоту (6) колебаний контура ФАПЧ, которая в дальнейшем потребуется для определения коэффициентов цифрового фильтра петли – пропорционального и интегрирующего.

$$\Delta F = EsN0 * R_s * \delta\phi^2 = 0.187 * 10728 * 0.061 = 7.486 \text{ Гц} \quad (4)$$

$$F_n = \frac{2 * \Delta F}{\varepsilon + \frac{1}{4\varepsilon}} = \frac{2 * 7.486}{0.707 + \frac{1}{4 * 0.707}} = 14.116 \text{ Гц} \quad (5)$$

$$F = F_n * \sqrt{1 - \varepsilon^2} = 9.982 \text{ Гц} \quad (6)$$

В формулах использовался коэффициент затухания -  $\varepsilon = 0.707$  и исходный параметр СКО фазы  $\delta\phi = 0.061$  рад (или 3.5 градуса). Произведем вычисление промежуточных коэффициентов (7 - 8), формулы которых приводятся в [3].

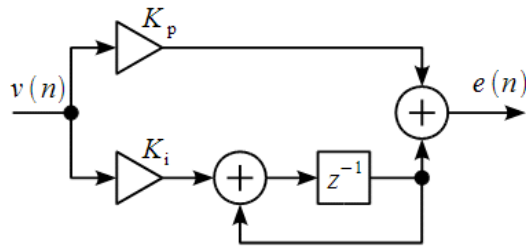
$$c0 = e^{-2\varepsilon * F_n * T_s} * \cos(F_n * T_s * \sqrt{1 - \varepsilon^2}) = \\ = e^{-2 * 0.707 * 14.116 * 10728} * \cos(14.116 * 10728 * \sqrt{1 - (0.707)^2}) = 0.99814 \quad (7)$$

$$c1 = e^{-\varepsilon * F_n * T_s} * \cos(F_n * T_s * \sqrt{1 - \varepsilon^2}) = \\ = e^{-0.707 * 14.116 * 10728} * \cos(14.116 * 10728 * \sqrt{1 - (0.707)^2}) = -1.99814 \quad (8)$$

В результате (9-10) получим коэффициенты цифрового фильтра 2 порядка, обеспечивающего минимальную ошибку по фазу в 4 град. Цифровой фильтр петли ФАПЧ изображен на рис. 3.

$$K_p = 1 - c_0 = 1 - 0.99814 = 1.859 \cdot 10^{-3} \quad (9)$$

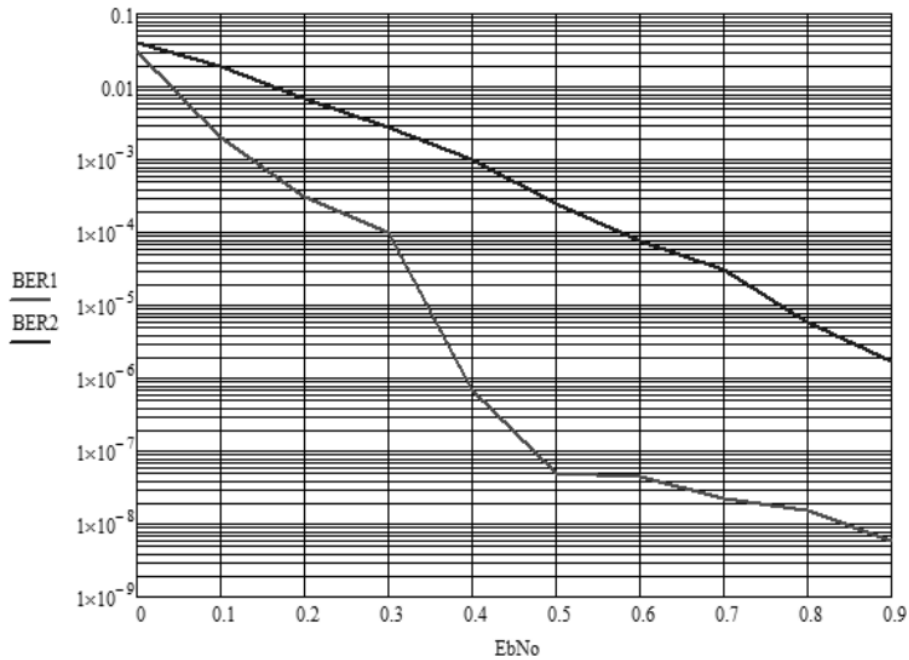
$$K_i = 2 + c_0 - g_1 = 2 - 0.001859 - 1.99814 = 1.72975 \cdot 10^{-6} \quad (10)$$



**Рис. 2.** Структурная схема цифрового фильтра петли ФАПЧ

Дисперсия фазовой ошибки (11) на выходе ФАПЧ 2 порядка будет составлять:

$$\Delta\Theta^2 = \frac{\Delta F * T_s}{EbN_0} = \frac{7.486 * 9.321 * 10^{-5}}{0.5} = 0,00035 \text{ рад}^2 \quad (11)$$



**Рис. 3.** Кривые помехоустойчивости работы системы когерентного приема

Характеристика помехоустойчивости системы с использованием ФАПЧ и декодера стандарта CCSDS (темная кривая) изображена на рис. 4. Для сравнения так же приведена характеристика помехоустойчивости при использовании (светлая кривая) одного турбо-кода стандарта CCSDS. Следовательно, система ФАПЧ вносит потери в систему, приблизительно 0,5 дБ. С целью уменьшения указанных потерь, предполагается использовать алгоритм декодирования Витерби для коррекции кодовых символов и определения фазовой ошибки. Одним из свойств сверточных кодов является то, что проходя по решетке диаграмм состояний декодера, возвращение к правильному пути происходит за 3-5 кодовых состояний, в результате чего, система будет реже вносить коррекцию. Однако это позволит снизить потери, вносимые системой ФАПЧ.

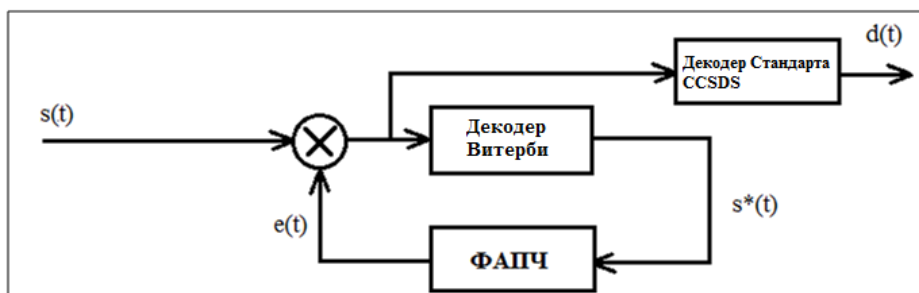


Рис. 4. Структурная схема работы алгоритма

При разработке алгоритма были проведены исследования работы турбокода стандарта CCSDS, определена его помехоустойчивость. Рассчитана система ФАПЧ исходя из параметров системы, построена характеристика помехоустойчивости с использованием системы синхронизации и турбо-кода стандарта CCSDS. Заметим, что система фазовой автоподстройки частоты вносит потери в систему, для того чтобы улучшить характеристику помехоустойчивости был предложен алгоритм восстановления несущей частоты с использованием декодера Витерби. В данный момент ведется исследование характеристики помехоустойчивости работы алгоритма, предполагается выигрыш в системе на 0,2-0,3 дБ.

### Литература

1. Скляр, Б. Цифровая связь. Теоретические основы и практическое применение./ Б. Скляр. – М.:Вильямс, 2007. – 1104с.
2. Морелос-Сарагоса, Р. Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение./ Морелос-Сарагоса – М.:Техносфера, 2005. – 320с.
3. Li, W. J Meiners. Introduction to phase-locked loop system modeling./ W. Li, J. Meiners. – Texas: Texas Instruments incorporated, 2005. режим доступа -<http://www.ti.com/lit/an/slyt169/slyt169.pdf>.
4. Nezami, M. RF Architectures and Digital Signal Processing Aspects of Digital Wireless Transceivers./ M. Nezami. – Noble pub, 2008. – 425с.
5. Лобов Е.М., Косилов И.С., Кандауров Н.А., Елсуков Б.А. Метод определения помехоустойчивости сложных сигнально кодовых конструкций на основе семейства широкополосных ортогональных сигналов и недвоичного ldpc-кода в условиях ионосферного канала // Т-Comm: Телекоммуникации и транспорт. 2014. Т. 8. № 8. С. 55-59.
6. Лобов Е.М., Косилов И.С., Кандауров Н.А., Елсуков Б.А. Методика оценки параметров частотной дисперсии ионосферного канала с помощью широкополосного фазоманипулированного сигнала // Т-Comm: Телекоммуникации и транспорт. 2014. Т. 8. № 9. С. 49-53.
7. Кандауров Н.А., Косилов И.С., Лобов Е.М. Метод определения с помощью имитационного моделирования помехоустойчивости сложных широкополосных сигнально-кодовых конструкций в условиях ионосферного канала // REDS: Телекоммуникационные устройства и системы. 2014. Т. 4. № 2. С. 108-112.

# КОНЦЕПЦИЯ РАДИОТЕХНИЧЕСКОГО ЭФИРНОГО ВЕЩАНИЯ «РАДИО МТУСИ»

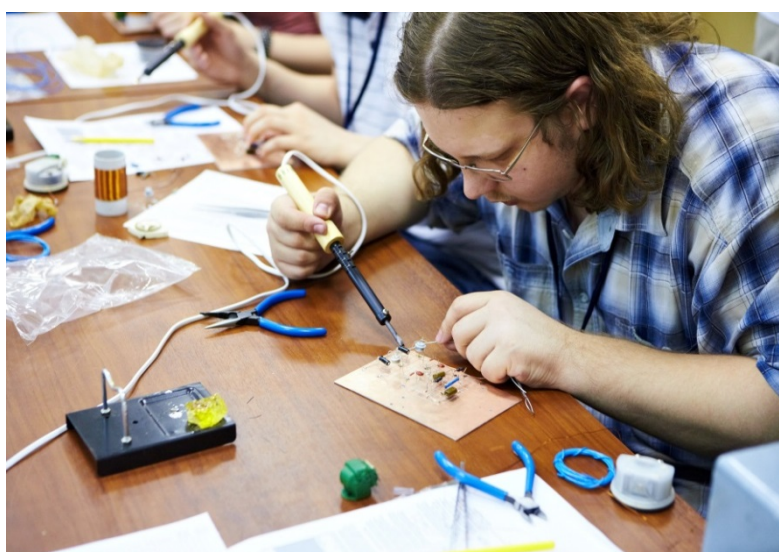
*Токарева Е.М.,  
МТУСИ, студентка группы БРМ1301  
Комаров С.Н.,  
МТУСИ, старший преподаватель*

В наше время молодым людям открыты все пути, занимайся, чем душе угодно, но зачастую в представленном многообразии "глаза разбегаются" и молодёжь все труднее чем-то увлечь. Создание студенческой радиостанции ориентировано, в первую очередь, на будущих инженеров. Да, возможно, они сами ещё об этом не догадываются, но кто, как не увлеченный сверстник, может по-настоящему заинтересовать? Поделиться личным опытом, знаниями и собственными увлечениями. Продвижение в молодёжной среде увлечения радиотехникой позволит детям, подросткам и молодым людям больше узнать о радиотехнике, потому что чего не знаешь, тем и не интересуешься, а заинтересовав, сможет с ранних лет определиться с выбором профессии и, зная свою цель, развиваться в этом направлении.

Когда молодой человек осмысленно подходит к выбору своей профессии, он получается специалистом на порядок более высокой квалификации чем те, кто, придя учиться, все еще не знает, зачем он здесь. Студенты, которые с младших курсов принимают участие в практических занятиях по своей будущей профессии, с гораздо большим интересом и пониманием изучают специальные дисциплины на старших курсах. Участники радиокружков, как правило, оканчивают ВУЗ либо с отличием, либо становятся профессионалами высокого класса. А ведь именно увлеченные своей профессией специалисты - престиж любого ВУЗ-а!

Во время Дней открытых дверей не только рассказываем школьникам о ВУЗ-е, но и показываем реальные практические приборы, изготовленные студентами в радиокружке или на дипломной практике. Начиная с первого курса на занятиях по Введению в профессию, приглашаем студентов заниматься в радиокружке, в секции Азбуки Морзе, в коллективе студенческого любительского и вещательного радио.

Радиокружок существует на кафедре РОС в лаборатории РПДУ. Члены радиокружка учатся паять, разбираться с радиосхемами, с элементной базой и собирают для себя различные радиотехнические конструкции. Общая программа в радиокружке есть только для начинающих. Те же, кто определился с областью своих увлечений, изготавливают в кружке то, что им интересно.



Секция изучения азбуки Морзе существует на кафедре РОС в лаборатории РПДУ. Члены секции учатся принимать на слух знаки Морзянки, передавать на телеграфном ключе и параллельно изучают правила работы в эфире. А еще, мы болтаем, смеемся и рассказываем анекдоты морзянкой. Кто первый без ошибок принял, тот раньше и смеется! Передаем морзянкой поэмы А. С. Пушкина, стихи, скороговорки!

Коллективная студенческая любительская радиостанция RU3C позволяет осуществлять радиосвязь со всем миром на коротких волнах, практиковать конструирование и установку антенных систем, изучать распространения радиоволн, а так же это языковая практика при работе телефоном – диалоги в эфире с носителями любого языка!



*Радиостанция RU3C и ее руководитель Дингес Сергей Иванович*

Вещательная студенческая радиостанция «Радио МГУСИ» транслирует в прямом эфире все, о чем было рассказано выше:

Пропагандирует интересы и увлечения техническими профессиями, рассказывает о МГУСИ и его жизни, студенты сами говорят о своей учебе и практической деятельности в секциях и кружках и о своих достижениях, студенты показывают в эфире работу приборов сделанных своими руками.

Частоты вещания:

- средние волны: 1584(1485) кГц, волна 189(202) м.
- короткие волны: 25900 кГц, волна 11,58 м.

Мощность радиопередатчиков: 400 Вт в режиме несущей.

Зона покрытия при приеме на внешние антенны:

- средние волны: Москва и Московская обл.
- короткие волны: первым скачком – Англия, Италия, Голландия, третьим скачком – Флорида, Алабама, Джорджия, Индиана, Тенесси, Илинойс, Висконсин.

В деятельности студенческого радио ценность имеет не столько работа на аудиторию, как сам процесс создания программ и вещания. Причем, второе значительно важнее. Хотя и приятно получать письма от слушателей, например, из Алабамы, которые к письму прикладывают звуковой файл с записью твоего голоса с их американского радиоприемника.

Аудитория у радиостанции, вещающей на средних и коротких волнах исключительно заинтересованная. Случайных людей в ней нет. Это либо радиолюбители, рыскающие по эфиру в поисках новых интересных радиостанций, либо знакомые, которые специально настраивают радиоприемники на частоту «Радио МГУСИ» не ради того, чтобы узнать что-то новое, а исключительно за компанию с тем, кто там будет вещать.

Также в аудиторию студенческой радиостанции обязательно попадут члены различных радиокружков, руководители которых тоже хотят силами кружковцев построить подобные радиостанции «Индивидуального радиовещания». И таких людей мы приглашаем выступать в нашем эфире.



Пропаганда занятий через личный пример и общение: личный пример дают радиокружок, секция Морзянки, связная радиостанция. А предметное общение можно реализовать в прямом эфире. Мало того, в эфире «Радио МТУСИ» можно создать центр радиотехнического общения!

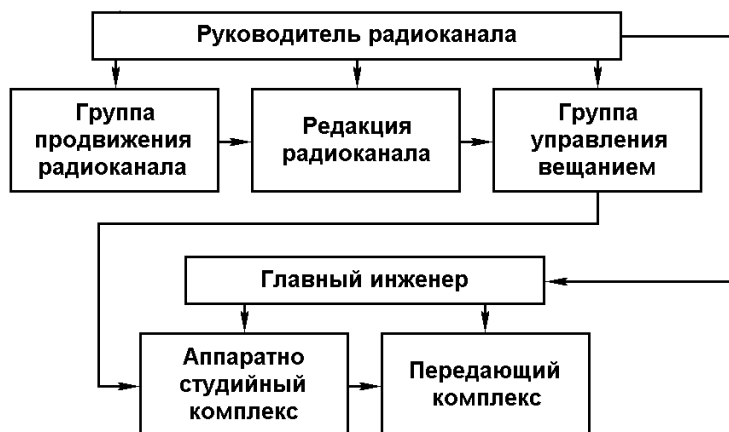
А дальше молва об интересной радиостанции, которую можно не только послушать, но и в эфире которой выступить, сама соберет целевую, заинтересованную аудиторию.

«Радио МТУСИ» - не одинокая радиостанция в студенческом эфире. В рамках федерального проекта «Индивидуальное радиовещание» создаются радиостанции при радиокружках по всей стране. Уже вещает «Радио Бонч» в Санкт-Петербурге, «Кенотрон» в Ленинградской области, «Странник» в деревне на границе Вологодской и Архангельской областей, готовится к выходу в эфир «Интерес» – в радиокружке Дома творчества «На Миуссах»... Строятся и другие радиостанции.

Поскольку «Индивидуальное радиовещание» - проект федеральный, то у него есть свой сайт: [www.cqf.su](http://www.cqf.su).

Инициатор проекта «Индивидуальное радиовещание» ООО «Радиовещательные технологии», директор которого помимо работы преподает на кафедре РОС МТУСИ Комаров Сергей Николаевич.

Структура радиоканала «Радио МТУСИ», утвержденная ректором в 2015 г.







*Руководитель радиоканала Комаров Сергей Николаевич  
и главный инженер Разин Олег Александрович в прямом эфире «Радио МТУСИ»*

Следующие этапы – подача заявки на лицензию, участие в конкурсе на радиочастоту, получение Разрешения на использование частот, ввод объекта связи в эксплуатацию и начало регулярного вещания. Параллельно – становление редакции, поиск студентов на роли музыкальных редакторов, звукорежиссеров, ведущих новостных и радиотехнических программ, составление сетки вещания, сочинение и запись позывных и «одежды эфира», освоение вещательного комплекса.

Если кратко, одним абзацем, охарактеризовать концепцию вещания нашей радиостанции, то она совпадает с целями всего проекта «Индивидуальное радиовещание»:

«Проект «Индивидуальное радиовещание» служит активизации технического творчества, развитию радиоконструирования, привлечению молодежи, в учебные заведения на специальности в областях радиотехники, радиосвязи, радиовещания, освещению исторических событий и этапов развития отечественной и мировой радиотехники, повышению уровня подготовки специалистов отрасли связи, а также повышению престижа инженерных профессий».

## **Литература**

1. Индивидуальное радиовещание. [www.cqf.su](http://www.cqf.su).
2. Комаров С.Н., Морозова А.П. Исследование усилителя мощности радиочастотных колебаний с многофазным возбуждением // Т-Сотм: Телекоммуникации и транспорт. 2014. Т. 8. № 10. С. 33-38.
3. Комаров С.Н., Тertyшниковa А.В. Разработка и исследование многофазных синтезаторов радиочастот с мощным выходом // Т-Сотм: Телекоммуникации и транспорт. 2013. Т. 7. № 9. С. 97-99.

# ИССЛЕДОВАНИЕ ВЛИЯНИЯ ДИНАМИЧЕСКИХ ГОЛОВОК НА ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ УСИЛИТЕЛЯ НА ЛАМПАХ И УСИЛИТЕЛЯ НА ПТ В ПРОГРАММЕ MICRO-CAP

*Болотов А.О., Холюков Р.Г.  
МТУСИ, студенты группы БРА 1301  
Кубицкий А.А.,  
МТУСИ, к.т.н., доцент  
Бажин А.В.,  
МТУСИ, инженер*

В силу объективной необходимости усилитель мощности звуковой частоты обычно испытывают и настраивают на эквивалент нагрузки в виде активного сопротивления 8 Ом или 4 Ома [4]. Однако реально усилитель мощности звуковой частоты работает на различные динамические головки, имеющие значительные электромеханические резонансы, создающие комплексные сопротивления нагрузки, фазовые сдвиги в выходной цепи усилителя мощности звуковой частоты и деформацию эквивалентной амплитудно-частотной характеристики [6]. В литературе нет всесторонних сведений о технических характеристиках усилителей на лампах и усилителей на полевых транзисторах при работе на эквивалент динамической головки.

Широко распространено субъективное мнение, что ламповый усилитель звучит лучше, чем транзисторный. В связи с этим проводится исследование характеристик усилителей мощности на электронных лампах и полевых транзисторах с целью сравнения их характеристик.

Для того, чтобы провести исследование усилителя мощности на электронных лампах были подобраны математические модели электронных ламп для программы Microcap. [7] Характеристики математических моделей ламп были протестированы. Результаты показали, что характеристики математических моделей ламп совпадают с характеристиками ламп из справочника. Принципиальная схема исследуемого УМЗЧ приведена на рисунке №1.

Первый каскад УМЗЧ собран по дифференциальной схеме на триодах 6Н23П (он усиливает напряжение и одновременно выполняет функцию фазоинвертора). В цепи катода используется источник тока, который обеспечивает высокую симметричность каскада. Оконечный каскад собран по двухтактной трансформаторной схеме в режиме близком к АВ на пентодах 6П18П. Трансформатор смоделирован как идеализированный без учета нелинейностей сердечника. По характеристикам лампы было определено необходимое сопротивление нагрузки для получения заданной мощности.

Исходя из этого выбран необходимый коэффициент трансформации.

Исследования проводились с нагрузкой в виде резистора 8 Ом, эквивалента одиночной динамической головки и в виде эквивалента двух динамических головок, комплексное сопротивление которых в звуковом диапазоне частот близко к 8 Ом. Характеристики УМЗЧ были получены при амплитуде напряжения на входе равной 1В и частоте 1 КГц. В ходе работы были получены: АЧХ, КПД, Рн, график гармонических составляющих и коэффициент гармоник [5].

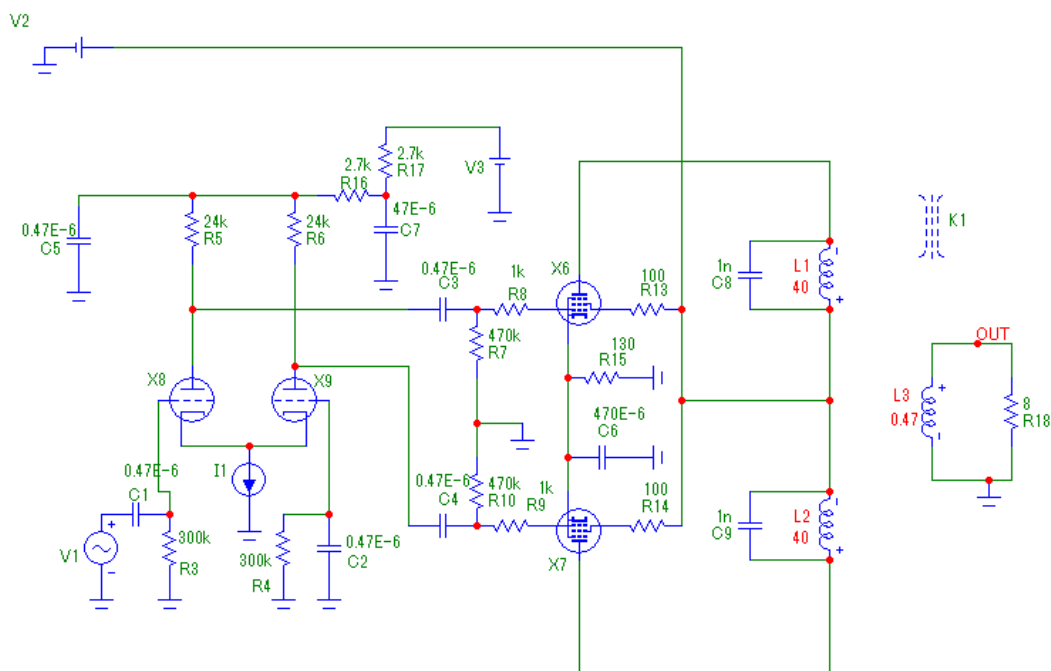


Рис. 1. Принципиальная схема УМЗЧ на электронных лампах

Результаты исследования приведены в таблице 1.

Таблица 1

	УМЗЧ с нагрузкой в виде резистора	УМЗЧ с нагрузкой в виде экв. Дин. Гол.	УМЗЧ с экв. 2-х Дин. Гол.
Р	8.7 Ватт	6.9 Ватт	9.3 Ватт
КПД	42.2%	28.7%	41.4%
Кг	2%	18.3%	12.3%

Полученные результаты говорят, что технические показатели УМЗЧ с сопротивлением на 8 Ом, отличаются от показателей УМЗЧ с нагрузкой в виде эквивалента одной или двух динамических головок. Это обусловлено в первую очередь высоким выходным сопротивлением УМЗЧ. Уменьшение выходного сопротивления можно добиться введением общей отрицательной обратной связи. Однако в случае лампового усилителя с трансформаторным выходным каскадом плохая фазовая характеристика трансформатора не позволяет ввести обратную связь достаточной глубины.

Далее проводились те же исследования с усилителем мощности звуковой частоты на полевых транзисторах. Проходная ВАХ которых близка к параболической, что позволяет ожидать меньших нелинейных искажений в случае использования симметричного двухтактного выходного каскада.

Первый каскад УМЗЧ на полевом транзисторе (MPF9200) выполнен как схема с общим истоком, с местной обратной связью.

Второй каскад, который является предвыходным, также выполнен по схеме с общим истоком на полевом транзисторе (MFE9200).

Оконечный каскад собран по схеме двухтактного истокового повторителя в режиме близкому к АВ на транзисторах (2SK308, 2SJ127). Применяется общая отрицательная обратная связь, последовательная по напряжению, охватывающая весь усилитель. Для повышения КПД используется вольтодобавка с выхода усилителя в истоковую цепь предвыходного каскада [5].

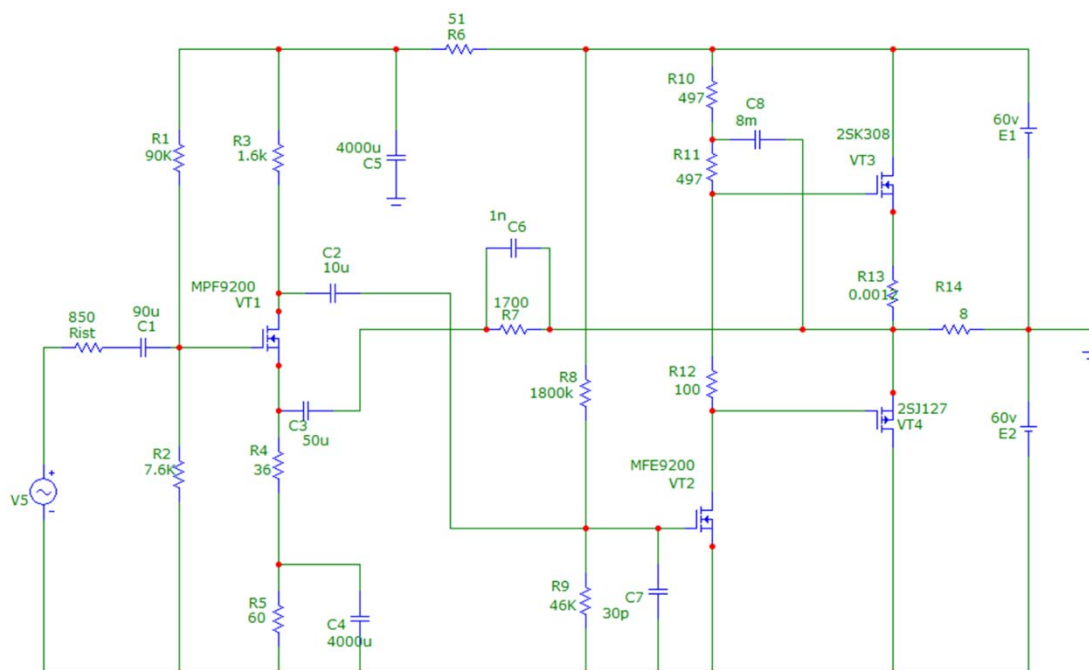


Рис. 2. Принципиальная схема УМЗЧ на полевых транзисторах

Также, как и для УМЗЧ на электронных лампах исследования проводились с нагрузкой в виде резистора 8 Ом, эквивалента одиночной динамической головки и в виде эквивалента двух динамических головок, комплексное сопротивление которых в звуковом диапазоне частот близко к 8 Ом. Все характеристики были получены при амплитуде напряжения на входе равной 1В и частоте 1 КГц. В ходе работы были получены: АЧХ, зависимости тока, напряжения и мгновенной мощности, рассеиваемой на стоке, КПД, средняя мощность нагрузки, спектр сигнала и коэффициент гармоник.

Также была снята АЧХ усилителя с большей глубиной обратной связи, за счет введения дополнительной петли обратной связи и измерен его коэффициент гармоник.

В качестве результатов исследования можно привести таблицу сравнения характеристик УМЗЧ на полевых транзисторах с разными нагрузками.

У усилителя с дополнительной петлей обратной связи АЧХ является равномерной в звуковом диапазоне частот, не зависимо от вида нагрузки. Коэффициент гармоник равен 0.01%.

Таблица 2

	УМЗЧ с нагрузкой в виде резистора	УМЗЧ с нагрузкой в виде экв. Дин. Гол.	УМЗЧ с экв. 2-х Дин. Гол.
Р	115.3 Ватт	55.4 Ватт	60.3 Ватт
КПД	51.6%	32.8%	43.3%
Кг	0.03%	0.04%	0.4%
Р	114.5 Ватт	169.9 Ватт	99.7 Ватт

Результаты исследований показывают, что при нагрузке в виде динамических головок могут иметь место значительные перегрузки окончных транзисторов, по току и пиковой мощности. Поскольку транзисторы в отличие от электронных ламп быстрее выходят из строя при перегрузках, то для предотвращения выхода их из строя необходимо принимать специальные меры по их защите. Увеличение глубины обратной связи делает усилитель менее чувствительным к сопротивлению нагрузки и уменьшает коэффициент искажений.

### **Вывод по проделанной работе:**

Трудности введения глубокой отрицательной обратной связи в усилителе мощности звуковой частоты на электронных лампах не позволяют устранить деформацию АЧХ, в основном вызванную высоким выходным сопротивлением, и всех основных характеристик при работе на разные динамические головки. Согласно таблице №1, работа УМЗЧ на комплексную нагрузку приводит к изменениям практически всех основных технических показателей УМЗЧ. Поэтому необходимо вводить глубокую отрицательную обратную связь для того, чтобы ламповый усилитель работал одинаково на разные нагрузки, либо необходимо принимать другие схемотехнические решения, снижающие выходное сопротивление без введения общей отрицательной обратной связи, для того чтобы сделать УМЗЧ универсальным. А в УМЗЧ на полевых транзисторах можно вводить глубокую ООС, что позволяет значительно снизить влияния комплексного сопротивления нагрузки в виде различных динамических головок на технические параметры УМЗЧ. Однако глубины отрицательной обратной связи, обеспечивающей  $K_T = 0.03\%$ , оказалось недостаточно для устранения изменения технических параметров УМЗЧ. Как при одиночной динамической головке, так и при двух динамических головках имеет место значительное возрастание пиковых (мгновенных) мощностей рассеиваемых на стоке транзисторов. Показано, что причиной этого также являются фазовые сдвиги из-за комплексной нагрузки. Проведенные исследования показали, что УМЗЧ на полевых транзисторах при двухпетлевой отрицательной обратной связи сохраняет свои технические параметры при работе на разные динамические головки.

### **Литература**

4. *Дуглас Селф*. Схемотехника современных усилителей. ДМК Пресс, 2011. 529 с.
5. *Кубицкий А.А.* Методические указания по курсовому проектированию “Электрический расчет и компьютерное моделирование УМЗЧ”. М.: МТУСИ, 2009. 36 с.
6. *Мусатов К.В.* Моделирование работы акустических систем с помощью компьютера // Журнал “Радио”. 2013. №9-10.
7. *Амелина М.А., Амелин С.А.* Программа схемотехнического моделирования Micro-Cap, Версии 9, 10. Смоленск: Смоленский филиал НИУ МЭИ, 2013. 618 с.

# РАЗРАБОТКА И ИССЛЕДОВАНИЕ ПЕРЕДАТЧИКА С КВАДРАТУРНОЙ АМПЛИТУДНОЙ МОДУЛЯЦИЕЙ

*Тактакишвили В.Г.,  
МТУСИ, студент группы БРС1201  
Комаров С.Н.,  
МТУСИ, старший преподаватель*

Амплитудную модуляцию почти во всех областях радиосвязи давно вытеснили частотная, фазовая и однополосная модуляции. Всего две отрасли, где осталась АМ – это связь в авиации (поскольку АМ не искажается эффектом Доплера) и КСДВ радиовещание, где простота массового АМ радиоприемника и наличие большой зоны неуверенного приема, (где вполне удовлетворительно можно слушать информационные радиопередачи) определяют в значительной степени экономику отрасли. Однако, низкая помехозащищенность АМ и плохая энергетика радиолинии заставляют искать альтернативу. Как один из возможных вариантов модернизации АМ проф. Поляковым В. Т. была выдвинута идея квадратурной АМ.

Эта модуляция основывается на привычной АМ, но в спектре сигнала несущая имеет поворот фазы на 90 градусов относительно боковых. С этим изменением несущая и полезный сигнал становятся взаимно ортогональны. Амплитудный спектр сигнала в эфире при этом не меняется, но огибающая перестаёт опускаться ниже своего уровня в режиме молчания. Временная диаграмма показана на рисунке 1.

В рамках ВКР была произведена разработка и макетирование формирователя Аналоговой Квадратурной Амплитудной Модуляции — АКАМ, позволяющего получить сигнал для последующего усиления.

Разрабатываемый формирователь сигнала должен быть прост в настройке и обеспечивать возможность изменения рабочей частоты без значительных изменений в схеме. Для обеспечения требуемых характеристик решено сформировать сигнал на некоторой промежуточной частоте (далее "ПЧ") и затем переносить полученный сигнал при помощи преобразователя частоты в необходимую область спектра.

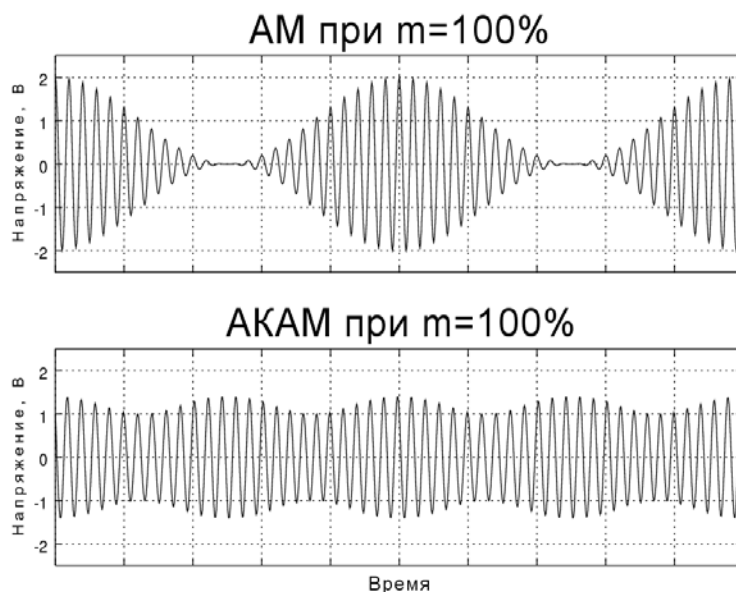
Для построения формирователя выбрана импортная микросхема AD835A, сочетающая в себе четырёхквadrантный перемножитель с двумя дифференциальными входами (X и Y) и сумматор с несимметричным входом Z. Эта микросхема имеет двуполярное питание  $\pm 5$  вольт и имеет возможность работы на низкоомную нагрузку в диапазоне частот от постоянного тока до 250 МГц. Применение этой микросхемы возможно как для построения модулятора, так и для переноса сигнала с ПЧ на требуемую частоту при помощи гетеродина (являющегося частью формирователя) с последующей фильтрацией зеркального канала, имеющего высокую частоту.

Так как у перемножителя входы имеют дифференциальную структуру, фазовращатель можно выполнить по схеме последовательной RC-цепочки, где несущая для перемножителя снимается с конденсатора, а напряжение с резистора поступает на суммирующий вход Z. Структурная схема формирователя показана на рисунке 2.

В итоге перемножитель получает сигнал, запаздывающий на 90 градусов относительно входного тока, а сумматор - сигнал, совпадающий по фазе со входным током. Учитывая большие входные сопротивления микросхемы (60 кОм и более) и малые ёмкости (около 2 пФ), неточностью поворота фазы, вызванной нагрузкой фазовращателя, можно пренебречь. Стоит отметить, что точная подстройка фазовращателя под рабочую частоту не требуется, так как разность фаз на входах перемножителя и сумматора постоянна и не зависит от частоты. От входной частоты зависит только соотношение амплитуд на входах перемножителя и сумматора, которое задаёт коэффициент модуляции.

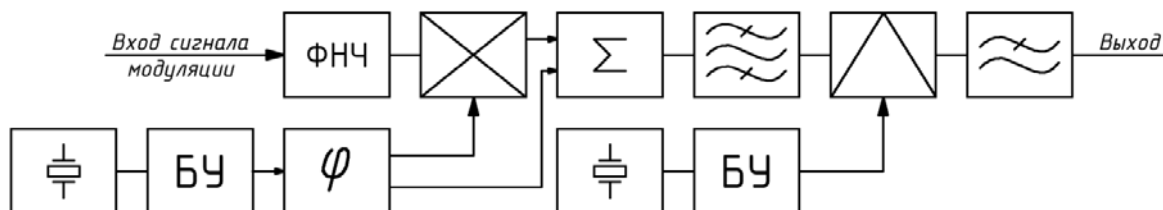
В отличие от АМ, в сигнале АКАМ не наблюдается перемодуляция, а КПД модуляции может быть значительно повышен путём увеличения коэффициента модуляции. Например, при  $m=1$  (максимальное значение для АМ, выше образуется перемодуляция) КПД модуляции равен 33.3%, а при использовании АКАМ с  $m=5$  он превышает 92%, что видно из рисунка 3.

При макетировании получена осциллограмма на рисунке 4. На ней вверху показан модулирующий сигнал на выходе ФНЧ, поступающий на перемножитель, а внизу — результирующий сигнал на выходе формирователя (после переноса с ПЧ на рабочую частоту и фильтрации при помощи ФНЧ 1 порядка). Амплитуда сигнала невелика и осциллограф вносит некоторые искажения в сигнал, что вызывает несимметричность огибающей. Можно заметить также небольшое отличие локальных максимумов огибающей — на отрицательной полуволне модулирующего сигнала локальный максимум ниже, чем на положительной.

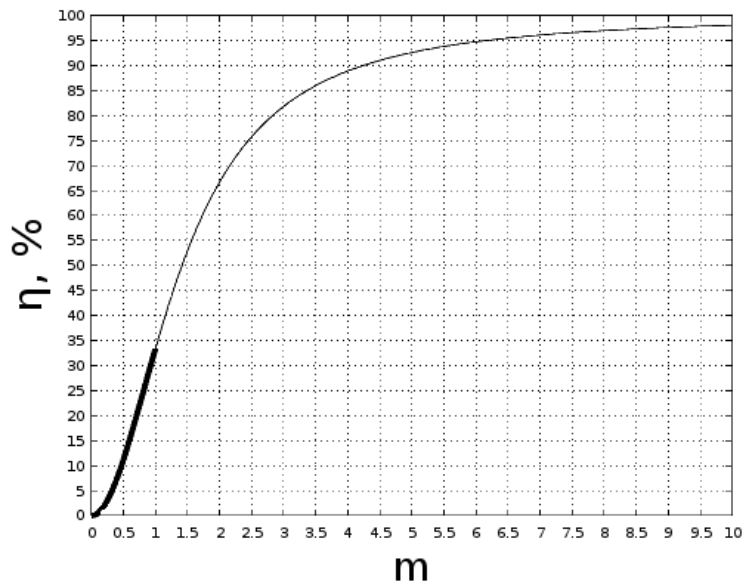


*Рис. 1. Временные диаграммы АМ и АКАМ*

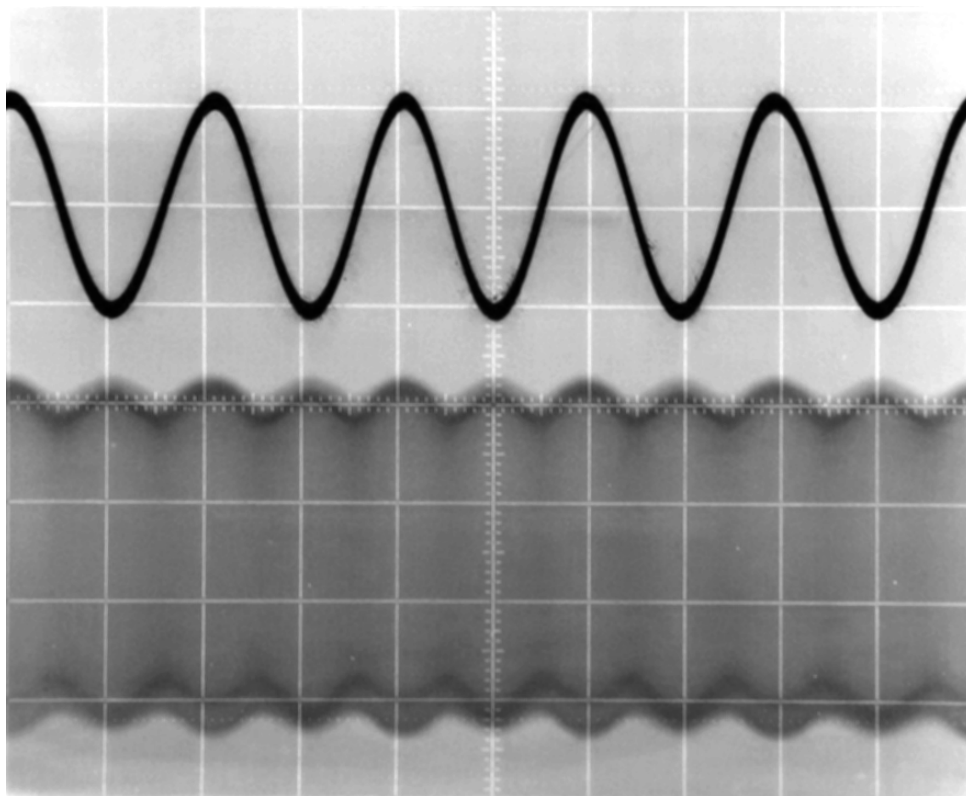
Это может быть вызвано неточностью установки рабочей точки перемножителя из-за несимметричности напряжений питания (по результатам замеров: на положительной шине 4.8В, на отрицательной -5.1В), несимметричности входных токов входного дифференциального каскада и других факторов. Эксперимент подтвердил теоретическую возможность формирования сигнала АКАМ и показал некоторые проблемы, не исследованные ранее, но проанализировать этот сигнал можно только создав приёмник. В будущем планируется сборка приёмника и исследование радиоканала на основе АКАМ.



*Рис. 2. Структурная схема формирователя*



*Рис. 3. Зависимость КПД модуляции АМ и АКАМ от коэффициента модуляции. Жирной линией показан КПД модуляции АМ*



*Рис. 4. Осциллограмма работы модулятора. Вверху исходный сигнал, внизу — сигнал на выходе модулятора*



## Литература

1. *Белов Л.А., Богачев В.М., Благовещенский М.В.* и др. Устройства генерирования и формирования радиосигналов. Учебник для вузов, под ред. Г. М. Уткина. М.: Радио и связь, 1994. 416 с.
2. *Момот Е.Г.* Проблемы и техника синхронного радиоприема. Под ред. В.И. Сифорова. Ленинград: Полиграфкнига, 1960. 178 с.
3. *Шахгильдян В.В.* Проектирование радиопередающих устройств/Шахгильдян В.В. [и др.]. М.: Связь, 1976. 432 с.
4. *Комаров С.Н., Морозова А.П.* Исследование усилителя мощности радиочастотных колебаний с многофазным возбуждением // Т-Сотт: Телекоммуникации и транспорт. 2014. Т. 8. № 10. С. 33-38.
5. *Комаров С.Н., Тertyшникова А.В.* Разработка и исследование многофазных синтезаторов радиочастот с мощным выходом // Т-Сотт: Телекоммуникации и транспорт. 2013. Т. 7. № 9. С. 97-99.

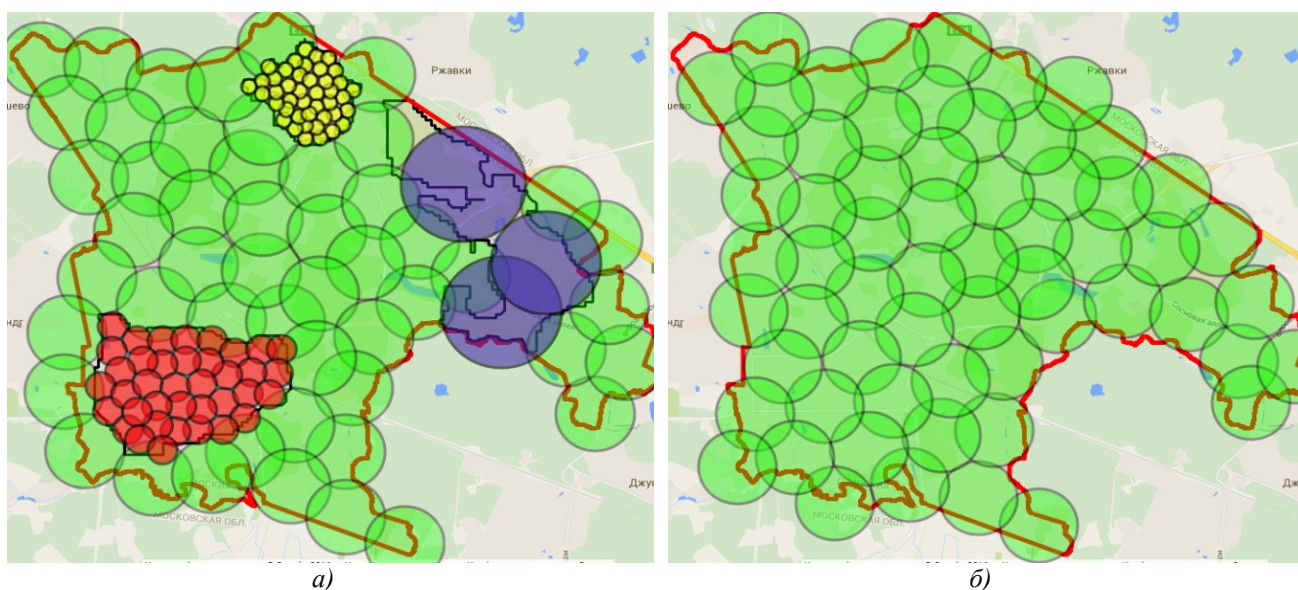
# ОЦЕНКА ВЛИЯНИЯ ВЫБОРА МОДЕЛИ СОТОВОГО РАДИОПОКРЫТИЯ НА ХАРАКТЕРИСТИКИ ФУНКЦИОНИРОВАНИЯ СЕТИ МОБИЛЬНОЙ СВЯЗИ

*Колпаков В.М., Синюхин В.С.,  
МТУСИ, студенты гр. БРМ1202  
Сорокин А.С.,  
МТУСИ, профессор, к.т.н.*

## **Постановка задачи и конкретные цели ее решения**

В наше время очень часто планирование сотовых сетей систем мобильной связи (СМС) осуществляется с помощью однородной модели, предполагающей одинаковые значения параметров по всей территории обслуживания (ТО) СМС [1, 2]. Однако, в реальных условиях такое предположение весьма оптимистично и может приводить к существенным ошибкам при планировании СМС. Последнее наиболее характерно для СМС, организуемых на территории больших городов (БГ), так как современный БГ имеет сложную пространственно-территориальную структуру, состоящую из значительного числа зон с различной плотностью населения и параметрами территории и прочими различиями. И, если не учитывать специфику каждой из таких зон, это может привести к невыполнению требований к качеству сотовой сети связи. Как в технической, так и в учебной литературе данной проблематике практически не уделено должного внимания, в связи с чем и было решено провести данное исследование.

Задача исследования заключается в том, чтобы путем численного анализа и графического представления полученных результатов показать влияние учета различного числа зон неоднородности (ЗН) на качество работы реальных сотовых СМС. На рис. 1 приведены в качестве иллюстрации виды реальной сотовой структуры и ее возможного однородного эквивалента (модели).



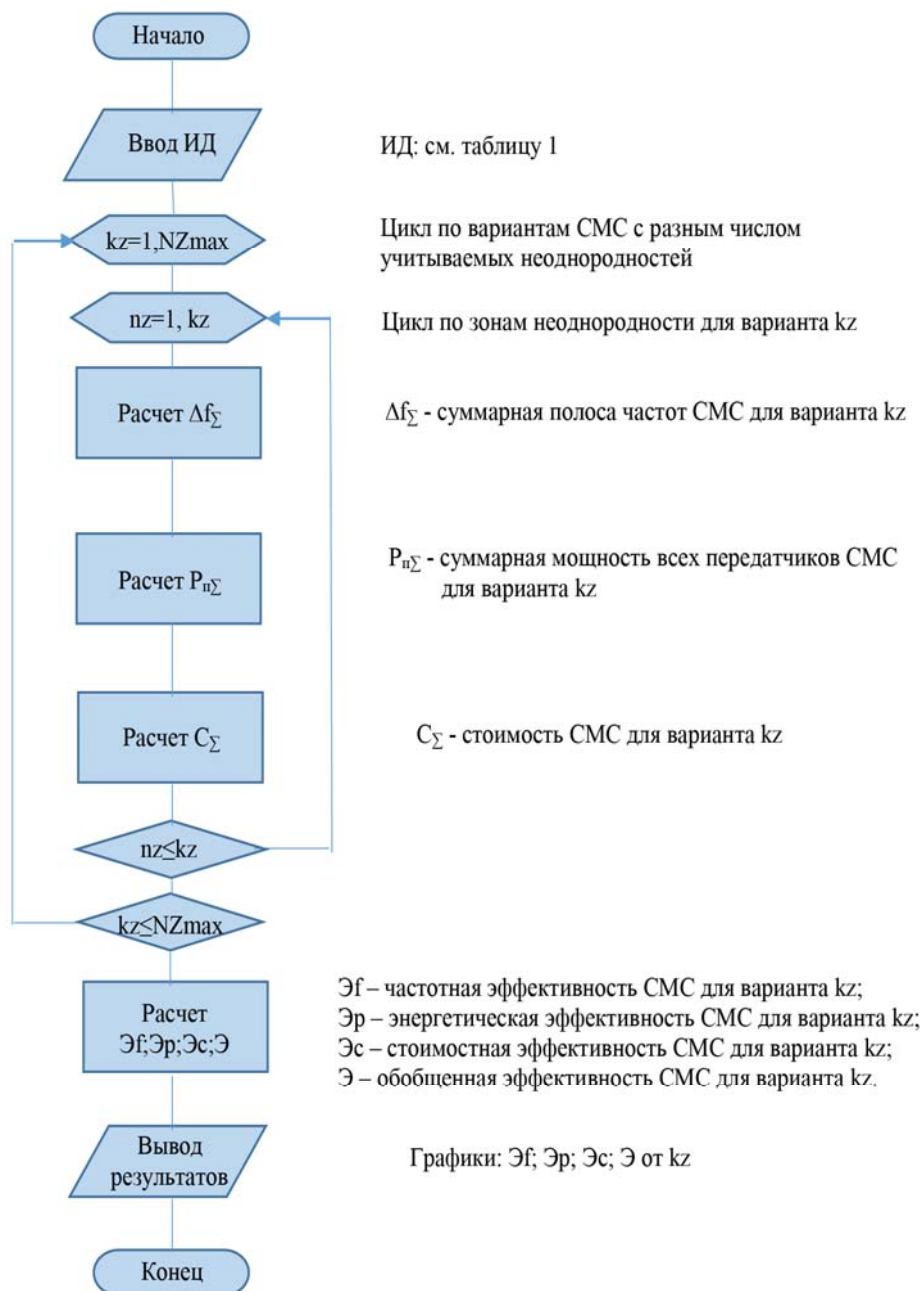
**Рис. 1.** Реальная структура сотового покрытия СМС (а)  
ее возможная однородная модель (б)

По результатам указанного выше анализа и оценки предполагается целесообразной разработка методики определения параметров эквивалентной однородной модели, что позволит обоснованное выполнение частотно-территориального планирования сотовых СМС на основе их однородных моделей.

### Алгоритм расчета

Для решения данной задачи был разработан алгоритм расчета, позволяющий графически сравнить стоимостную, частотную, энергетическую и общую эффективности при построении сетей СМС для однородной сотовой планировки и с выделением некоторых зон неоднородности.

Данный алгоритм был реализован в программной системе Matlab при структуре параметров моделей, приведенных в табл. 1. В этой же таблице приведены конкретные значения параметров однородной модели, соответствующей типовой модели сотовой СМС для территории БГ [2].



**Рис. 2.** Программный алгоритм расчета эффективности СМС при различном числе учитываемых зон неоднородности

### Расчетные модели неоднородной СМС с различным числом учитываемых зон неоднородности

Различных ЗН в реальной ситуации может быть выделено достаточно много, но в данной работе было решено ограничиться их максимальным числом, равным 15. Так как в данных моделях основную площадь занимают районы с высокой плотностью населения, можно утверждать, что такие модели соответствуют типу местности большой город.

Для получения результатов были использованы не только типовые параметры для оператора сотовой мобильной связи, но и создан универсальный набор типомодели местности [2], характеризующие особенности рельефа на соответствующей территории: – БГ, районы с высокой плотностью населения; – средний город (СГ), районы со средней плотностью населения; – малый город (МГ), районы с низкой плотностью населения; – сельская местность (СМ); – открытая местность (ОМ). Помимо перечисленных параметров типомodelей территории использовались следующие параметры ЗН: – площадь ЗН, км<sup>2</sup>; – число абонентов в каждой ЗН, тыс. абон.; – абонентская нагрузка от одного абонента в час наибольшей нагрузки ЧНН, Эрл.

На рис. 3 показана типизированная модель неоднородной сотовой СМС, на территории обслуживания которой выделено 15 специфических ЗН. Указанное максимальное число ЗН, принятое в данной работе, определено эмпирически путем анализа территорий значительного числа БГ.

На рис. 4 показана структура модели неоднородной сотовой СМС с пятью эквивалентными ЗН (ЭЗН), которые сформированы путем объединения территорий смежных ЗН, меньших размеров. Так, например, ЭЗН №2 получена объединением ЗН 2, 3, 5, 6, 8 и 10.

Таблица 1

**Типовые параметры однородной модели сотовой СМС на территории БГ**

Обозначение	Величина	Значение для БГ	Единица измерения
<i>Kto</i>	Тип территории в зонах неоднородности		
<i>Sto</i>	Площадь территории обслуживания зон неоднородности	600	км <sup>2</sup>
<i>Nab</i>	Число абонентов в зонах неоднородности	115	тыс. чел.
<i>NZmax</i>	Максимальное число учитываемых зон неоднородности	15	-
<i>f</i>	Диапазон рабочих частот СМС	2600	МГц
<i>Tcdop</i>	Допустимый ПНС в соте из-за замираний	1	%
<i>Tmdop</i>	Допустимый ПНС в соте из-за воздействия МС	2	%
<i>Tbldop</i>	Допустимый процент блокировки канала трафика	5	%
<i>A1</i>	Абонентская нагрузка от 1 абонента	0,07	Эрл
<i>Rab</i>	Минимальная гарантированная в ЧНН скорость передачи абонентских сигналов	10	Мбит/с
<i>dfl</i>	Ширина полосы пропускания 1 радиоканала	20	МГц
<i>idf</i>	Индекс учета дополнительной полосы частот	-	-
<i>Msec</i>	Число секторов в сотах	3	-
<i>M</i>	Модуляция 64-QAM	64	-
<i>Rrk</i>	Максимальная скорость передачи в радиоканале	100	Мбит/с
<i>pibsdBmax</i>	Допустимая ЭИИМ БС	25	дБВт
<i>ptasdBmax</i>	Допустимый уровень мощности АС	0	дБВт
<i>pcminbs0</i>	Чувствительность приемника БС для типового радиоканала	-160	дБВт
<i>dpcmin</i>	Различие чувствительностей приемников БС и АС	10	дБ
<i>gabs0</i>	Коэффициент усиления всенаправленной антенны для заданного диапазона рабочих частот	8	дБ
<i>Ccs</i>	Стоимость оборудования ЦС	1,5	млн. долл.
<i>Cbs1</i>	Стоимость 3-х секторной БС	0,1	млн. долл.
<i>Crr11</i>	Стоимость однопролетной СРПЛ	0,025	млн. долл.

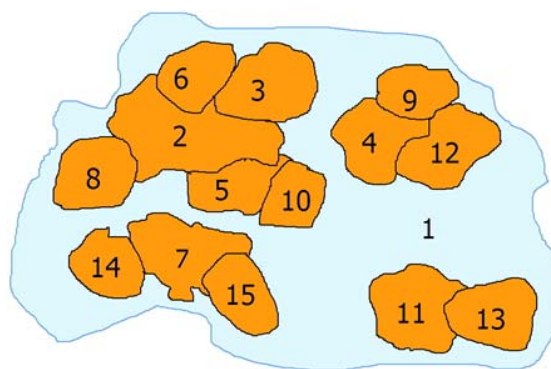


Рис. 3. Модель неоднородной СМС с выделением пятнадцати ЭЗН

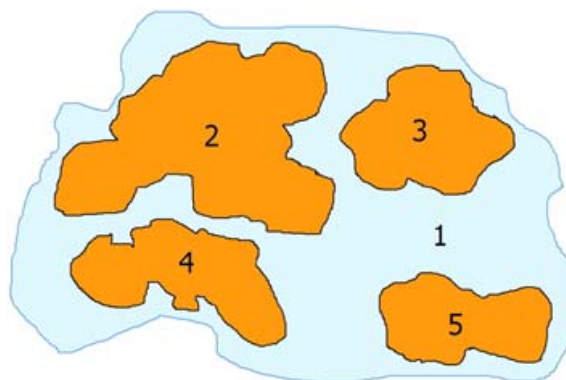


Рис. 4. Модель неоднородной СМС с выделением пяти ЭЗН

Таким образом были сформированы варианты неоднородных моделей с числом ЭЗН, равным 2, 5 и 10, а анализ проводился, соответственно для 5 вариантов модели сотовой СМС.

### Результаты расчетов

На рис. 5 приведены результаты расчета эффективности функционирования для указанных выше 5 вариантов моделей сотового покрытия территории БГ, различающихся числом ЭЗН, обозначенных на рисунках *kZ*. При этом на двух верхних рисунках и нижнем левом приведены зависимости от параметра *kZ* частных эффективностей: частотной эффективности для однородной модели  $Ef0pr$  (в %) и для вариантов неоднородной модели  $Efpr$  (в %); энергетической эффективности для однородной модели  $Ep0pr$  (в %) и для вариантов неоднородной модели  $Eppr$  (в %); стоимостной эффективности для однородной модели  $Ec0pr$  (в %) и для вариантов неоднородной модели  $Ecpr$  (в %). На правом нижнем рисунке приведена зависимость от параметра *kZ* обобщенной эффективности для однородной модели  $E0pr$  (в %) и для вариантов неоднородной модели  $Epr$  (в %).

Расчет указанных показателей функционирования проводился в соответствии с определениями и соотношениями, приведенными в [1].

### Выводы

Как можно видеть из полученных графиков, увеличение учитываемого числа зон неоднородностей приводит к снижению эффективности СМС относительно однородного варианта модели. При этом величина такого снижения составляет (10...50)% по разным видам эффективностей. Данное положение указывает на важность учета неоднородности территориально-информационной структуры сотовой СМС.

Поскольку полученные результаты имеют достаточно обобщенный характер, так как для их получения использовались типовые условия функционирования, то эти результаты можно использовать при анализе и выборе оптимального числа учитываемых зон неоднородности при планировании сотовых СМС на реальной территории. Так в соответствии с данными результатами, оптимальным числом зон неоднородности составляет 10, которое определяется точкой наибольшего снижения энергетической эффективности СМС (верхний правый график), так как при проектировании реальных сотовых СМС всегда учитывается наилучший вариант.

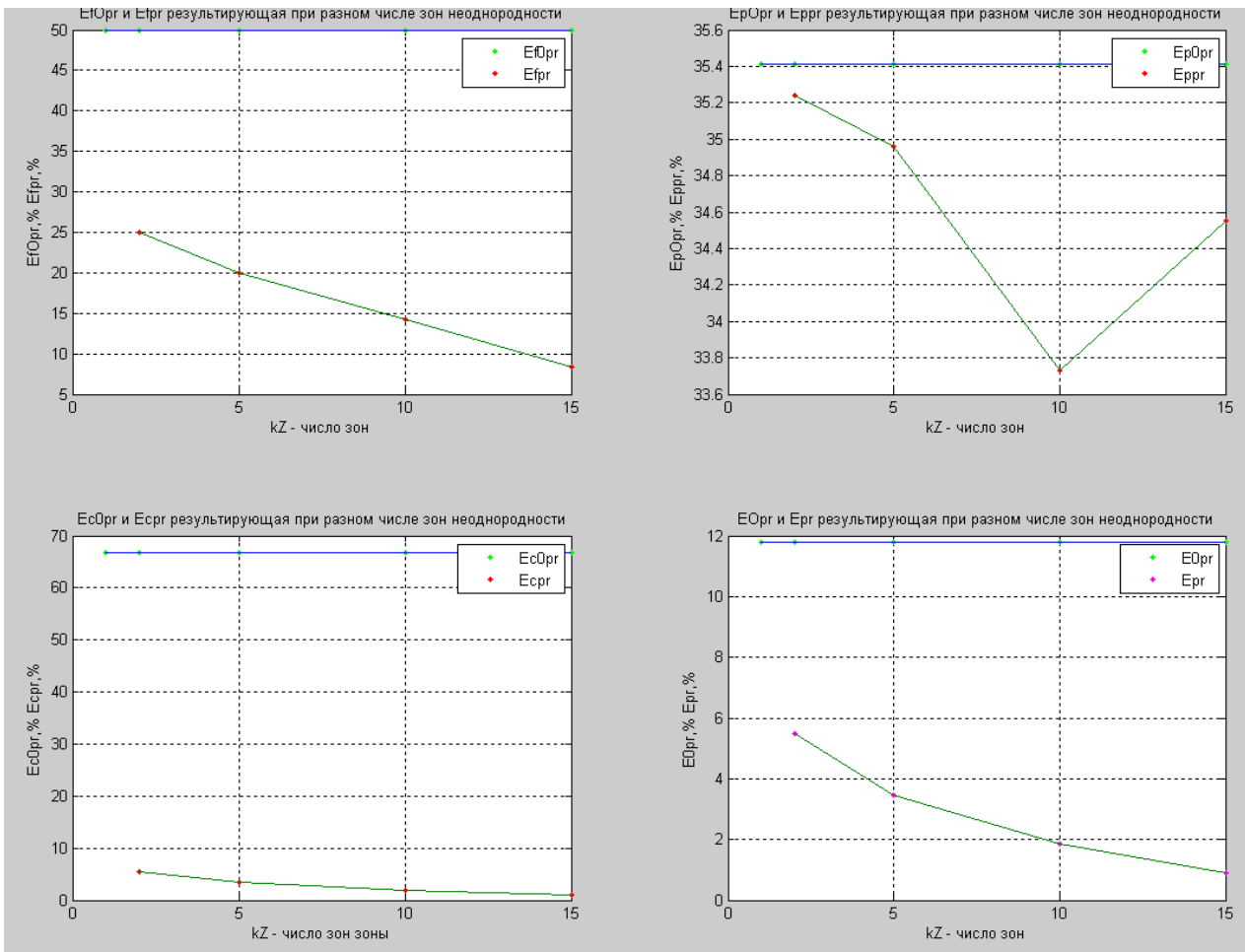


Рис. 5. Зависимость эффективности однородной и неоднородной сотовой СМСот числа учитываемых ЗН

## Литература

1. Сорокин А.С. Основы теории построения телекоммуникационных систем и сетей: Учебное пособие. М.: МТУСИ, 2012. 52 с.
2. Сорокин А.С. Сотовые системы радиосвязи. Курсовое проектирование: Учебное пособие. М.: МТУСИ, 2006. 31 с.

# РАЗВИТИЕ ЗАКОНОДАТЕЛЬСТВА О КОММЕРЧЕСКОЙ ТАЙНЕ

*Афонин С.П.,  
МТУСИ, студент группы ЗРС 1301  
Кубанков Ю.А.,  
МТУСИ, к.э.н.*

Целью статьи является исследование эволюции института коммерческой тайны и освещение современных проблем охраны коммерческой тайны.

До своего современного понимания коммерческая тайна прошла большой исторический путь. На Западе понятие «коммерческая тайна» появилось в Англии в 1817 году, однако истоки этого термина лежат в римском праве. В нашей стране впервые законодательно тайна и ответственность появились в начале XVIII века. В их числе также было сокрытие данных о коммерческих операциях, но отдельно оно не выделялось. Лишь к середине XIX века, наряду с различными профессиональными тайнами, такими как, например, адвокатская и врачебная, закрепилась отдельно и коммерческая тайна. Статьи 1187 и 1316 Уложения о наказаниях общего определения (1845 г) уголовно наказывали за преждевременную огласку торговой сделки или каких-то иных вредных для продавца сведений. Стоит отметить, что тогда не существовало как такового понятия «коммерческая тайна», только «промышленная тайна», которая включала в себя фабричную и торговую тайну. Лишь со временем термин пришел к современному названию.

В начале XX века, в связи с бурным развитием экономики в стране, конкуренция между соперничающими в одной отрасли предприятиями становилась острее. Все чаще недобросовестные дельцы шли на умышленный вред своим соперникам. Большую роль в разрешении этого вопроса сыграло Министерство торговли и Промышленности, образованное в 1905 году. Были выработаны определенные механизмы противодействия недобросовестной конкуренции. Один из них непосредственно был связан с закреплением понятия «коммерческая тайна». За выведывание сведений о чужих коммерческих сделках требовалось возместить ущерб, а если был доказан умысел нанести вред конкуренту, возбуждалось уголовное дело.

В Советской России понятие «коммерческая тайна» некоторое время продолжало существовать. В шестилетний период НЭПа(1921-1928г) советское правительство признавало частный бизнес, и вместе с ним право на охрану сведений о коммерческих сделках. Однако с окончанием НЭПа все предприятия национализировались, в связи с чем исчезла конкуренция между предприятиями, а следовательно различные сведения о коммерческой деятельности перешли в категорию государственной тайны.

Лишь накануне распада СССР, в период перестройки, снова возникла необходимость в этом юридическом понятии. Статья 33 «Закона о предприятиях в СССР» закрепляла коммерческую тайну как «не являющиеся государственными секретами сведения, связанные с производством, технологической информацией, управлением, финансами и другой деятельностью предприятия, разглашение (передача, утечка информации) которых может нанести ущерб его интересам» и назначала ответственность за нарушение, ссылаясь на другие законодательные акты. Однако, парадокс заключался в том, что эти акты так и не были приняты. Соответственно, юридически никакого наказания за нарушение коммерческой тайны в СССР период перестройки не было. Через шесть месяцев, в декабре 1990 г., в РСФСР был принят Закон "О предприятиях и предпринимательской деятельности", который запрещал предприятию разглашать сведения, которые могли бы составлять коммерческую тайну и назначил ответственным руководителя предприятия решать, какие сведения считать коммерческой тайной, а какие – нет. [1]

В процессе распада СССР и перехода от плановой экономики к рыночной возникла конкуренция между предприятиями, количество операций резко увеличилось – это требовало усиления законодательства, связанного с защитой коммерческой тайны. Первой ласточкой стала статья 2 Закона РСФСР «О собственности в РСФСР» среди других объектов интеллектуальной собственности были названы ноу-хау и торговые секреты. Стоит отметить, что уже тогда явно обозначились недостатки

законодательства, связанного с этой сферой. Так как экономика очень бурно развивалась, возникали все новые и новые виды операций, и законодатели попросту не успевали вовремя закреплять столь стремительно расширяющее понятие. Это создавало прецеденты для необоснованного засекречивания информации, которая по уму коммерческой тайной даже не являлась. Учитывая это, было принято Постановление Правительства РСФСР от 5 декабря 1991 г. № 35 «О перечне сведений, которые не могут составлять коммерческую тайну». [2]

Дальнейшая тенденция развития законодательства шла по пути именно окончательного закрепления, что входит в понятие коммерческая тайна, а что – нет.

Немаловажным этапом в этом вопросе стало принятие первой части Гражданского Кодекса Российской Федерации. Вопросы коммерческой тайны рассматриваются в ст. 139 ГК РФ. «Информация составляет служебную или коммерческую тайну в случае, когда информация имеет действительную или потенциальную коммерческую ценность в силу неизвестности ее третьим лицам, к ней нет свободного доступа на законном основании и обладатель информации принимает меры к охране ее конфиденциальности. Сведения, которые не могут составлять служебную или коммерческую тайну, определяются законом и иными правовыми актами. Лица, незаконными методами получившие информацию, которая составляет служебную или коммерческую тайну, обязаны возместить причиненные убытки. Такая же обязанность возлагается на работников, разгласивших служебную или коммерческую тайну вопреки трудовому договору, в том числе контракту, и на контрагентов, сделавших это вопреки гражданско-правовому договору».

В 2001 году было предпринята попытка проработать законодательство в области санкций, наступающих за нарушение коммерческой тайны. Была принята 183 статья Уголовного Кодекса РФ, которая назначала ответственность за «незаконные разглашение или использование сведений, составляющих коммерческую, налоговую или банковскую тайну». Если в результате возникли тяжелые последствия, грозило лишение свободы до 8 лет. Также вступила в силу Статья 13.14 Кодекса Российской Федерации об административных нарушениях, которая предусматривала за «разглашение информации, доступ к которой ограничен федеральным законом (за исключением случаев, если разглашение такой информации влечет уголовную ответственность)» штраф от 500 до 5000 рублей. Подпункт "в" п. 6 ч. 1 ст. 81 ТК РФ предусматривает возможность увольнения сотрудника в случае "разглашения охраняемой законом тайны (государственной, коммерческой, служебной и иной), ставшей известной работнику в связи с исполнением им трудовых обязанностей, в том числе разглашения персональных данных другого работника".[2]

Наряду с основным направлением развития законодательства института коммерческой тайны в российской правовой системе, возникло еще одно - информационно-правовое. Институт коммерческой тайны эволюционировал в ходе развитие института конфиденциальной информации. Первоначально этот институт закрепился в статье 41 Закона Российской Федерации «О средствах массовой информации», однако его деятельность ограничивалась сотрудничеством СМИ с источниками информации. Особую же роль именно в развитие коммерческой тайны сыграл Федеральный закон «Об информации, информатизации и защите информации». Часть 9 статьи 2 указанного закона определяла конфиденциальную информацию, как документированную информацию, доступ к которой ограничивается в соответствии с законодательством Российской Федерации.

В тоже время, гражданско-правовое направление также активно развивается. В 2004 году Государственной Думой был принят Федеральный Закон о коммерческой тайне, который «регулирует отношения, связанные с установлением, изменением и прекращением режима коммерческой тайны в отношении информации, которая имеет действительную или потенциальную коммерческую ценность в силу неизвестности ее третьим лицам.» Этот закон был призван окончательно закрепить коммерческую тайну как важную часть российского законодательства. В нем даются четкие критерии, что не является коммерческой тайной, какие критерии ее получения, права обладателя коммерческой тайны, назначалась ответственность за нарушение, и т.д.

С тех пор закон неоднократно изменялся и дополнялся. Статья 139 ГК РФ потеряла свою силу. Вместе с тем, 1 января 2008 года была принята ч. 4 ГК РФ, а также был изменен и дополнен ФЗ «О коммерческой тайне». В 2015 году утратила свою юридическую силу ст. 2 Закона «О коммерческой тайне»: «Законодательство Российской Федерации о коммерческой тайне состоит из Гражданского кодекса Российской Федерации, настоящего Федерального закона, других федеральных законов». По оценке адвоката Анцупова, коммерческая тайна была изъята из компетенции института гражданского права и причислена к трудовому законодательству. [3]

Несмотря на то, что сфера коммерческой тайны развивается достаточно бурно, многие специалисты отмечают, что в законе еще достаточно много пробелов и белых пятен. Например, на сего-



дняшний день нельзя заставить уволенного сотрудника подписать договор о неразглашении информации, которая была засекречена уже после того, как его уволили.

Таким образом, назрела необходимость корректировать федеральные нормы, регулирующие отношения в связи с коммерческой тайной.

## Литература

1. Яковец Е. Н. Своеобразие содержания и основные исторические этапы развития коммерческой тайны в России [Электронный ресурс] // Center-Bereg: Юридический портал. URL: <http://www.center-bereg.ru/h151.html> (дата обращения: 20.03.2016).

2. М. Кудрявцев Коммерческая тайна в России: Новый закон – новые вопросы [Электронный ресурс] //Юридическая Россия. Федеральный правовой портал. URL: <http://www.law.edu.ru/doc/document.asp?docID=1224659> (дата обращения: 15.04.2016).

3. М. Кудрявцев Коммерческая тайна в России : Актуальные изменения и проблемы коммерческой тайны за последнее время [Электронный ресурс] //zakon.ru – Первая социальная сеть для юристов. URL: [https://zakon.ru/blog/2016/5/6/aktualnye\\_izmeneniya\\_i\\_problemy\\_kommercheskoj\\_tajny\\_za\\_poslednee\\_vremya](https://zakon.ru/blog/2016/5/6/aktualnye_izmeneniya_i_problemy_kommercheskoj_tajny_za_poslednee_vremya) (дата обращения: 16.04.2016).

# АНАЛИЗ ТИПОВ ПРИЛОЖЕНИЙ В АВИАЦИОННОЙ СИСТЕМЕ ПЕРЕДАЧИ ДАННЫХ

Григорьев И.Д.,  
МТУСИ, магистрант группы М091501(72)  
Орлов В.Г.,  
МТУСИ, к.т.н., доцент

В начале 1980-х годов, ИКАО (Международная организация гражданской авиации) признала усиливающиеся ограничения существующей навигационной системы и потребность в улучшениях. В 1983 году ИКАО создала *FANS (Committee of Future Air Navigation Systems)* с целью изучения и внедрения новых концепций и методов. *FANS* занялась разработкой новых систем *CNS (Communications, Navigation and Surveillance)* для управления воздушным движением (*ATM*) для организации связи земля-воздух преимущественно средствами цифрового канала передачи данных. ИКАО разработала архитектуру систем связи, которая удовлетворяла бы нуждам пользователей (авиакомпаниям и государствам).

Основой для внедрения и работы такой системы служит авионика. Авионикой называются электронные системы, располагающиеся на борту авиационного судна. Она занимается коммуникацией, навигацией, управлением множеством систем и является неотъемлемой частью современных летательных аппаратов.

В настоящее время данная архитектура частично внедрена в эксплуатацию [1]. В системе существует 6 главных компонентов:

1. Интерфейс пилота должен быть эффективным и простым во взаимодействии с системой передачи данных, помимо этого, организация взаимодействия пилот-контроллер требуют наличия крайне быстрого механизма. Использование цифрового канала передачи для предоставления интерфейса требует разработки таких процедур, при работе которых были бы минимизированы системные ошибки ввода.

2. Электроника авиалайнера является основой для работы цифровой системы передачи данных. Оборудование должно обрабатывать данные, принимаемые от интерфейса пилота, сенсоров и системы управления полетом, и направлять информацию в цифровой канал передачи данных в реальном времени. Оборудование также должно обрабатывать входящие извне сигналы от авторизованных источников.

3. Канал передачи данных воздух-земля и земля-земля может быть как *ATN*-совместимым, так и спутниковым, либо другим, соответствующим требованиям для работы с системой передачи. При передаче системой авиалайнера будет выбран наиболее подходящий путь на основании изменяющихся во времени параметров: географическое положение, стоимость передачи, задержка, пропускная способность и доступность соединения [2].

4. Канал воздух-земля будет соединен с системой управления воздушным движением через существующую наземную сеть связи. Сеть должна быть совместима с набором протоколов, используемых в концепции *ATN (Air Traffic Navigation)*. Для передачи любых сообщений наземные *ATN*-маршрутизаторы должны выбрать наиболее подходящий канал.

5. Наземная система должна осуществлять связь и распространение информации о местоположении с минимальным участием диспетчера. Мониторинг совместимости, избегание коллизий, обработка ошибок и большое количество других функций, связанных с безопасностью и эффективностью управления воздушным трафиком выполняет наземная система.

6. Интерфейс взаимодействия с контроллером содержит требуемые инструменты для композиции сообщений канала воздух-земля. Провайдеры авиационных систем проектируют и разрабатывают интерфейсы контроллеров, которые бы удовлетворяли их нуждам.

В настоящее время термины *ATM* и *ATN* являются взаимозаменяемыми в рамках авиационных систем. Схематичная архитектура представлена на рис. 1.

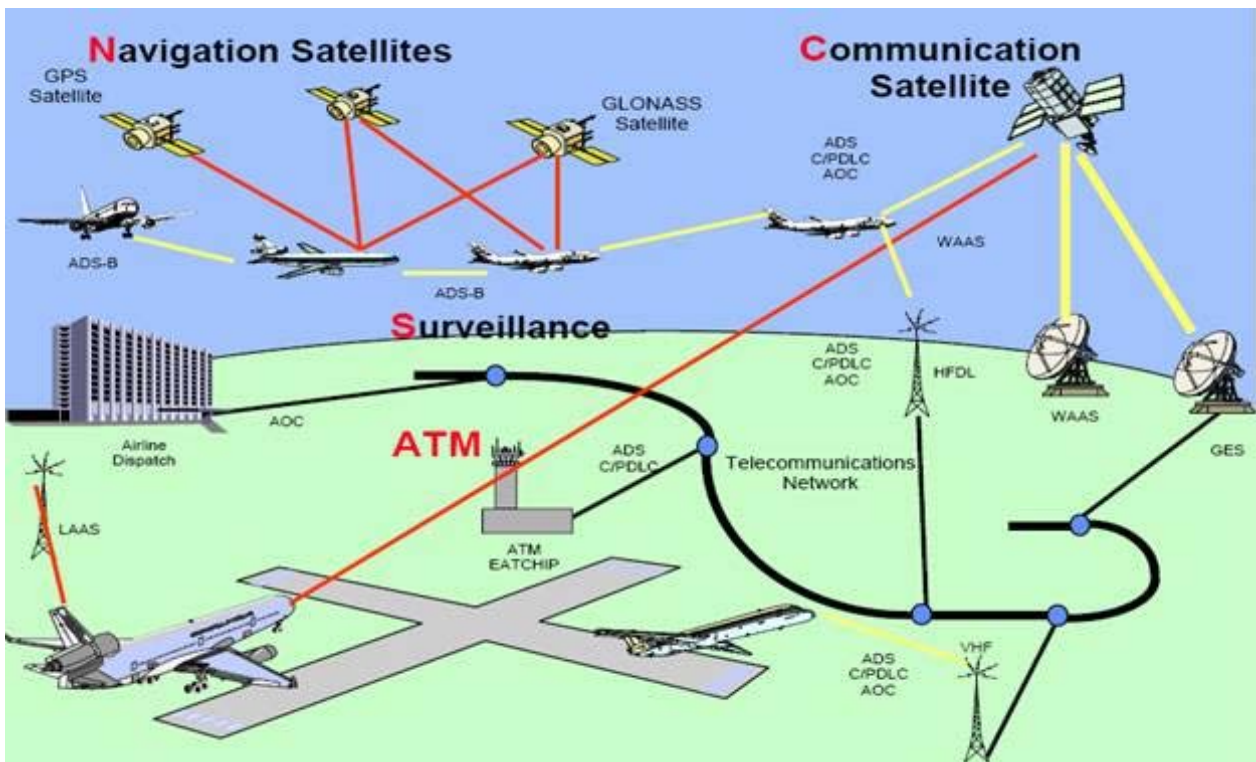


Рис. 1. Схематическое изображение архитектуры системы CNS/ATM

В рамках данной системы работает большое количество приложений, каждое из которых стандартизировано [3]. *Data Link Initiation Capability (DLIC)* предоставляет необходимую информацию для установки канала между наземной станцией и авиалайнером. Запуск приложения инициируется авиалайнером. Приложение выполняет следующие функции: вход в систему, выполнение инструкций по связи с наземными системами другого типа, локальное распространение информации, продвижение информации о входе в систему на земле.

*Automatic Dependent Surveillance (ADS)* – автоматическое зависимое наблюдение, данное приложение отвечает за наблюдение за положением воздушных судов, находящихся в океаническом пространстве и других зонах, где обеспечивается контроль управления трафиком без использования радара.

*Controller-Pilot Data Link Communications (CPDLC)* отвечает за организацию двустороннего обмена данными между авиадиспетчером и пилотом. Отправка *CPDLC* сообщения начинается с выбора адреса, выбора и, если необходимо, дополнения сообщения в отображаемом меню, либо другими способами, которые позволили бы организовать быструю компоновку сообщения и начало передачи. Сообщение может включать в себя разрешения на вход в воздушное пространство, ожидаемые разрешения, запросы, отчеты и иную информацию. Наиболее важным преимуществом работы данного приложения является снижение нагрузки на голосовой канал связи, поскольку эти данные в настоящее время передаются преимущественно с использованием аналогового голосового канала.

*Data Link Flight Information Services (FIS)* позволяет передавать информацию о полете, которая может быть предоставлена авиалайнеру в дискретной форме, например информация о погодных условиях, аэродромах и т.д. Такая информация поможет пилоту, усилив безопасность полета и улучшив ситуационное оповещение.

*ATS Interfacility Data Communication (AIDC)* обрабатывает информацию о приближающихся рейсах, границе района полетной информации. Также приложение занимается координацией условий пересечения границы и контролем трансфера.

*Automatic Dependent Surveillance – Broadcast (ADS-B)* - приложение наблюдения, занимающееся передачей таких параметров, как позиция и идентификация, широкоэмитерным каналом для приема таких данных любыми наземными или воздушными пользователями.

На основании теоретических данных о распределении трафика для каждого из видов приложений можно определить эталонные требования к качеству обслуживания, указанные в табл. 1. Вдобавок к указанным условиям, всем приложениям требуется:

- Вероятность потери отправленного сообщения должна быть меньше или равна  $10^{-6}$ ;
- Вероятность того, что после потери сообщения в системе, уведомление о потере не будет доставлено источнику должна быть меньше или равна  $10^{-9}$ ;
- Вероятность ошибочного направления сообщения к получателю должна быть равна или меньше чем  $10^{-7}$ ;

Таблица 1

### Требования к производительности для типов приложений

Приложение	Доступность (%)	Целостность	Надежность (%)	Непрерывность (%)
<i>DLIC</i>	99.9	$10^{-6}$	99.9	99.9
<i>ADS</i>	99.996	$10^{-7}$	99.996	99.996
<i>CPDLC</i>	99.99	$10^{-7}$	99.99	99.99
<i>FIS</i>	99.9	$10^{-6}$	99.9	99.9
<i>AIDC</i>	99.996	$10^{-7}$	99.9	99.9
<i>ADS-B</i>	99.996	$10^{-7}$	99.996	99.996

В табл. 2 отображены различные уровни производительности, которые могут быть выбраны для предоставления вышеописанных сервисов в рамках существующей системы. В зависимости от предоставляемого уровня сервиса, любое государство может определить производительность, требуемую в рамках данной области, руководствуясь такими факторами как распределение авиалайнеров в воздушном пространстве (эшелонирование), плотность трафика и пути его следования.

Таблица 2

### Требования ко времени задержки передачи

Уровни производительности	Рекомендованная задержка передачи в режиме «точка-точка», с	Задержка в 95% случаев в режиме «точка-точка», с	Задержка в 99.996% случаев в режиме «точка-точка», с
<i>A</i>	0.5	0.7	1
<i>B</i>	1	1.5	2.5
<i>C</i>	2	2.5	3.5
<i>D</i>	3	5	8
<i>E</i>	5	8	12.5
<i>F</i>	10	15	22
<i>G</i>	12	20	31.5
<i>H</i>	15	30	51
<i>I</i>	30	55	90
<i>J</i>	60	110	180

Можно также определить рекомендацию, при которой ни одно сообщение в соединении «точка-точка» не должно передаваться от источника к получателю дольше, чем 30 секунд. Исключением могут служить исключительные ситуации: природные катаклизмы, террористическая атака.

Развитием и разработкой *CNS/ATM* системы в настоящее время непрерывно занимаются ученые по всему миру, совершенствуя как аппаратную, так и программную базу, стандартизируя и внедряя новые элементы системы. Количество приложений, работающих в системе увеличивается, и в связи с этим, вопросы организации качества обслуживания выходят на первый план. Вышеописанное является достоверным признаком появления в ближайшем времени новых методов, которые позволят эффективнее и надежнее передавать данные в авиационных системах.

## Литература

1. ICAO Doc 9750-AN/963, Глобальный аэронавигационный план на 2013 — 2018 гг., издание четвертое, 2013.
2. ETSI EN 302 842-1 VHF air-ground and air-air Digital Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for aeronautical mobile (airborne) equipment, 2010.
3. ICAO Doc 9694-AN/955 Manual of Air Traffic Services Data Link Applications, 1st edition, 1999.
4. *Мазуркевич Д.О., Орлов В.Г.* Эволюция систем безопасности сетей сотовой связи разных поколений // Т-Comm: Телекоммуникации и транспорт. 2011. Т. 5. № 1. С. 38-40.
5. *Орлов В.Г., Мазуркевич Д.О.* Алгоритмы шифрования в публичных беспроводных сетях // Т-Comm: Телекоммуникации и транспорт. 2011. Т. 5. № 10. С. 62-64.

# ИССЛЕДОВАНИЕ И РАЗРАБОТКА МЕТОДА РЕСТАВРАЦИОННОЙ ОБРАБОТКИ СИГНАЛОВ ДЛЯ СИСТЕМ ЦИФРОВОГО ТЕЛЕВИДЕНИЯ

*Пушкина Е.В., Фролова М.М.,  
МТУСИ, студентки группы БРВ1201  
Власюк И.В.,  
МТУСИ, к.т.н. доцент*

Качество исходного сигнала существенно влияет на результат кодирования. Наличие искажений сказывается на работе кодера сжатия, принцип действия которого основан на внутрикадровой и межкадровой корреляции в структуре сигнала телевизионного изображения. Нарушение естественных связей между уровнями смежных пикселей из-за возникающих искажений фактически определяет изменение степени сходства смежных элементов, строк и кадров в структуре телевизионного сигнала. Наличие шума приводит к увеличению ( $\sim \sqrt{2}$  раз) межкадровой разницы на фоновых фрагментах сигнала изображения [1].

Следовательно, для эффективного сжатия необходимо устранить из исходного сигнала все факторы, препятствующие сокращению избыточности телевизионного изображения, для этого необходимо оптимизировать методы предварительной обработки видеосигнала, а именно методы противошумовой обработки. Результат в данном случае напрямую влияет на качество конечного изображения, получаемого зрителем.

Никаякая система регистрации не обеспечивает идеального качества изображений исследуемых объектов. Изображения в процессе формирования их системами обычно подвергаются воздействию различных случайных помех или шумов. Фундаментальной проблемой в области обработки изображений является эффективное удаление шума при сохранении важных для последующего распознавания деталей изображения. Сложность решения данной задачи существенно зависит от характера шумов.

Самыми распространёнными в вещательном телевидении являются шумы оборудования. Они возникают из-за неидеальности систем формирования, передачи и хранения информации. Подобные шумы представляют собой случайный белый шум и его разновидность - белый Гауссов шум.

В спектральной области, белый шум принимает постоянные значения на всех частотах и ограничивается только шириной полосы частот телевизионного канала. Однако, следует отметить, что оборудование в видеотракте может влиять на частотные свойства шума.

Пространственная частотно-контрастная характеристика (ЧКХ) датчика ТВ сигнала (ДТВС) имеет спад на нижних частотах. Экспериментально эта характеристика была построена в [2]. При этом распределение шумов светочувствительной матрицы является равномерным, следовательно, изменение отношения сигнал-шум на выходе системы будет определяться формой ЧКХ ДТВС. В следствии апертурной коррекции характеристики, в высокочастотной области возрастает уровень сигнала, при этом пропорционально возрастает уровень шумов.

Учитывая характер распределения шума, оптимальной для шумоподавления является высокочастотная фильтрация.

Блок высокочастотной фильтрации будем разрабатывать на основе метода коринга, который чаще всего применяется для наиболее распространенных видов помех, таких как, например, белый Гауссов шум.

Принцип коррекции по методу коринга заключается в разделении сигнала на две составляющие: высокочастотную и низкочастотную. При этом низкочастотную оставляют без изменений, а высокочастотную подвергают пороговой обработке. Если величина высокочастотной составляющей оказывается ниже порога, то она отбрасывается, если выше, то она передается на блок суммирования с компонентами низких частот. Метод коринга хорошо работает при низком уровне шума, так как все шумовые составляющие попадают в область ниже порога и полностью убираются, а все детали – в область выше порога и остаются неизменными.

При разработке блока противозумовой коррекции также следует учитывать особенности зрения человека. Известно, что ЧКХ зрительной системы имеет подъем в области низких частот. Следовательно, шум в этой области будет наиболее заметен для человека, поэтому помимо высокочастотной следует проводить и низкочастотную обработку сигнала.

Коррекция будет наиболее эффективной, если при обработке учитывать структуру изображения. Для этого рассмотрим возможные элементы алфавита при распознавании структуры, определим необходимые операции фильтрации и разработаем блок низкочастотной коррекции.

После анализа возможных структур изображения, был выбран следующий алфавит:

1) Фон

Характеризуется равномерным распределением (или по градиенту) уровня яркости. В результате выделения высокочастотной части фоновой блока будут получены близкие к нулю значения. Это свойство фона может быть использовано для его опознавания в блоке. Учитывая, что всегда на реальных изображениях имеются, по крайней мере, малоинтенсивные высокочастотные изменения, а также шумы, вносимые ДТВС и видеотрактом, необходима пороговая обработка, но значение порога должно быть достаточно малым.

2) Граница

Характеризуется резким перепадом уровня яркости в одном из направлений при относительно равномерной яркости в ортогональном направлении в пределах блока. Дифференцирование такой структуры в ортогональном направлении (относительно направления границы) даст однополярный импульс достаточно большой амплитуды, но малой длительности, которые определяются разностью уровней фона и объекта и крутизной перехода. Вдоль направления границы результат будет аналогичен случаю фона.

3) Линия

Характеризуется постоянным значением уровня яркости в блоке, за исключением узкой, в одном из направлений в пределах блока, полосы, имеющей другой уровень яркости. В результате дифференцирования структуры «линия» в ортогональном ей направлении получим биполярный импульс, среднее значение которого будет равно нулю. Дифференцирование вдоль направления линии приведет к результату, аналогичному структуре «граница».

4) Текстура

Характеризуется большим количеством высокочастотных компонент, которые имеют условно периодическую структуру, т.е. происходит повторение базового фрагмента или «зерна» текстуры в пространстве с пространственными изменениями этого фрагмента, типа «масштабирования», «поворота», «отображения» и т.д. Ввиду сложности определения этого типа структуры изображения, будем определять блок как текстурный, если он не относится к другим позициям алфавита, т.е. не является фоном, границей или линией.

Помимо опознавания структуры блока, требуется определить направление границ и линий. Для этого достаточно продифференцировать изображение в двух ортогональных направлениях, поскольку остальные являются линейно зависимыми, но в этом случае границы, расположенные под углом  $45^\circ$  к направлениям дифференцирования, могут быть опознаны как фон или текстура. Для устранения этого эффекта введем два дополнительных направления обработки – в направлении главной и побочной диагоналей. Таким образом обработку будем проводить в четырех направлениях: горизонтальном (Г), вертикальном (В), вдоль главной диагонали (ГД), вдоль побочной диагонали (ПД).

Операция интегрирования позволяет повысить отношение сигнала-шум без существенного снижения качества изображения. Рассмотрев все виды структур, определим для каждой из них направление процедуры обработки:

- Фон

Интегрирование во всех направлениях.

- Граница/линия

Интегрирование в направлении неизменной структуры, т.е. ортогонально направлению дифференцирования.

- Структура

Без обработки.

Таким образом, устройство шумоподавления будет состоять из блока распознавания структур и блока фильтрации.

Разработка блока распознавания структуры начнем с блока внутрикадровой обработки изображения. Вследствие такой обработки каждый кадр исходного изображения будет представлен мат-

рицей отсчетов, количество которых равно количеству направлений обработки исходного изображения, а количество «градаций» отсчета равно количеству позиций выбранного алфавита.

Структурная схема блока распознавания структуры представлена на рис. 1.

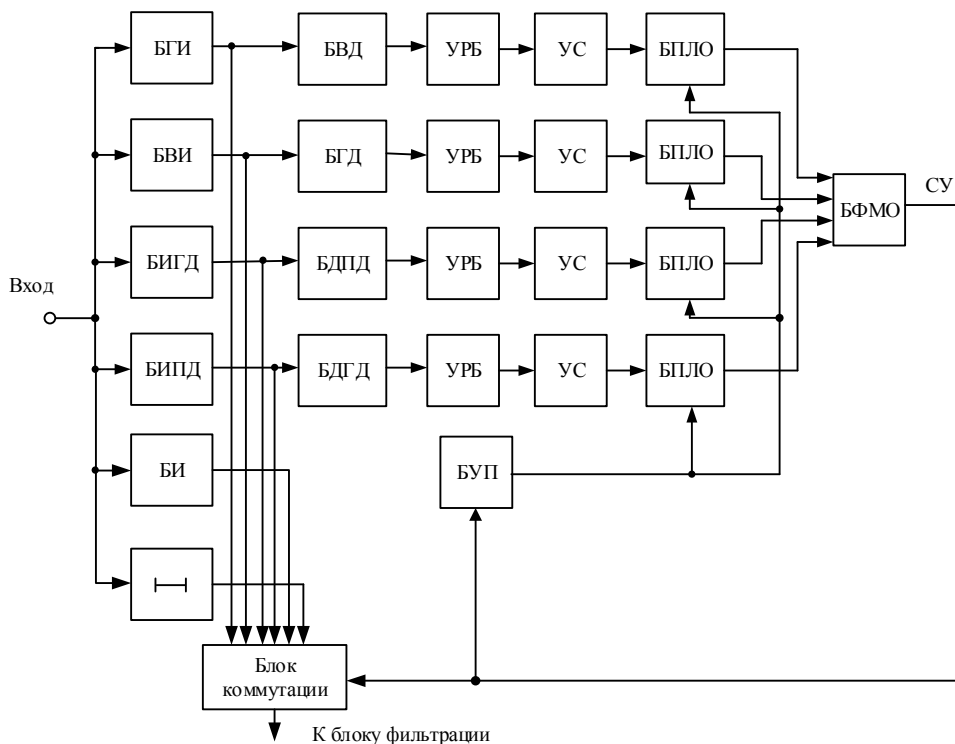


Рис. 1. Структурная схема блока распознавания структуры

Рассмотрим принцип работы блока внутрикадровой обработки:

1) В соответствии с масками фильтров, приведенными в таблице 1, последовательно во взаимно ортогональных направлениях осуществим интегрально-дифференциальную обработку изображений в каждом канале.

Таблица 1

**Маски фильтров для изотропной дифференциальной и интегральной обработки изображения**

		Направление обработки				Фон
		Вертикальное	Горизонтальное	Главная диагональ	Побочная диагональ	
Тип обработки	Интегральная	$\frac{1}{3} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}$	$\frac{1}{3} \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$	$\frac{1}{3} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\frac{1}{3} \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$	$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$
	Дифференциальная	$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}$	-

2) Разделим полученные изображения на блоки и проведем пороговую логическую обработку (БПЛО). Для этого вычислим сумму положительных  $S_p$  и отрицательных  $S_n$  значений элементов блока и проведем пороговую обработку (значение порога  $th$  выбирается экспериментально). В соответствии с таблицей 2 сформируем массив данных из изображений.



Таблица истинности для логической обработки сигнала

Оценка суммы элементов блока	Выходное значение			
	0 (00)	1 (01)	2 (10)	3 (11)
$S_p$	$<th$	$<th$	$\geq th$	$\geq th$
$S_n$	$<th$	$\geq th$	$<th$	$\geq th$

При использовании четырех направлений обработки и четырехпозиционного кодирования получается 256 различных вариантов состояния блока. Использовать такое количество позиций алфавита нецелесообразно, уменьшив его за счет объединения некоторых состояний блоков в одну позицию.

В результате анализа был выбран шестипозиционный алфавит:

0. Фон

$M = 0$ , если  $M_v = 0 \wedge M_h = 0 \wedge M_{md} = 0 \wedge M_{sd} = 0$ ,

где  $M$  – текущий элемент матрицы отсчетов на выходе,  $M_v$  – текущий элемент матрицы на выходе канала обработки в вертикальном направлении,  $M_h$  – в горизонтальном,  $M_{md}$  – в направлении главной,  $M_{sd}$  – в направлении побочной диагонали.

1. Граница / линия вертикального направления

$M_v = 0 \wedge M_h \neq 0$

2. Граница / линия горизонтального направления

$M_h = 0 \wedge M_v \neq 0$

3. Граница / линия направления главной диагонали

$M_{md} = 0 \wedge M_{sd} \neq 0$

4. Граница / линия направления побочной диагонали

$M_{sd} = 0 \wedge M_{md} \neq 0$

5. Текстура

В остальных случаях.

3) В блоке формирования матрицы отсчетов (БФМО) для каждого кадра входной последовательности в соответствии с выбранным алфавитом формируем матрицу отсчетов со значениями элементов равными номеру позиции алфавита.

После внутрикадровой обработки изображения сигнал управления (СУ) с БФМО поступает на блок установки порога (БУП) и блок коммутации, сигнал с которых идет к блоку фильтрации.

### Выводы

Существующие стандартные модели фильтров, как правило, не учитывают: характер распределения шума, особенности зрительной системы человека и структуру изображения. Такой подход к обработке не дает хороших результатов, поэтому для более эффективной коррекции требуется применение адаптивных фильтров. В процессе исследования были проанализированы стандартные модели шумов и фильтров, изучено влияние ДТВС на качество формируемого изображения, рассмотрены характеристики зрения человека. С учетом этих факторов было разработано устройство адаптивной фильтрации.

### Литература

1. Гонсалес Р. Вудс Р. Цифровая обработка изображений. М.: Техносфера, 2005. С. 1072.
2. Власюк И.В. Метод контроля пространственных характеристик телевизионных камер // Метрология и измерительная техника в связи, 2005. №6. С. 13.
3. Коржихин Е.О., Власюк И.В. Методы снижения пик+фактора в системах наземного цифрового телевизионного вещания стандарта DVB-T2 // Т-Сотт: Телекоммуникации и транспорт. 2012. Т. 6. № 9. С. 83-86.
4. Безруков В.Н., Власюк И.В., Канев С.А. Оценка относительного отклонения частоты при приеме сигналов наземных цифровых телевизионных вещательных систем // Т-Сотт: Телекоммуникации и транспорт. 2010. Т. 4. № 6. С. 9-11.
5. Власюк, И.В., Белов, А.С. Селекция нарушений с относительной пороговой оценкой структуры сигнала изображений в пределах дискретных фрагментов для телевизионных систем контроля безопасности объектов // Т-Сотт: Телекоммуникации и транспорт. 2010. Т. 4. № 9. С. 43-45.

# ИССЛЕДОВАНИЕ МЕТОДОВ КОДИРОВАНИЯ И ПРОТОКОЛОВ ПЕРЕДАЧИ СИГНАЛОВ ТЕЛЕВИЗИОННЫХ ИЗОБРАЖЕНИЙ В СИСТЕМАХ ИНТЕРНЕТ-ТЕЛЕВИДЕНИЯ ПО ТЕХНОЛОГИИ OTT-TV

Топильский С.А.,  
МТУСИ, студент группы МИТ 1502  
Власюк И.В.,  
МТУСИ, к.т.н., доцент

В настоящее время интернет-телевидение охватывает всё большее число абонентов, а появление такой технологии как *OTT (Over-the-top – «через голову»)* позволяет распространять телевизионный трафик не только среди весьма стационарных пользователей домашних телевизоров и компьютеров, но и среди постоянно перемещающихся пользователей смартфонов и планшетов. Данный момент позволяет вещателям не только увеличить аудиторию, но и снизить издержки на оборудования и при этом обеспечить увеличение абонентской базы, т.к. интернет есть практически везде[1].

Достоинствами данной технологии является встроенная поддержка многими устройствами, гибкость архитектура вещательного сервера, низкая зависимость от колебаний скорости передачи данных, широкая потенциальная аудитория, простота реализации интерактивных сервисов и низкие затраты на создание вещательной станции. Цель данной статьи – анализ современных методов кодирования и протоколов передачи телевизионного контента используемые в системах интернет-телевидения по технологии *OTT-TV*.

Для начала стоит разобраться что же такое *IP-сеть*, ведь она лежит в основе интернет-телевидения. *IP-сеть* – универсальный способ передачи цифровой информации, подразумевающий информационные сети различной структуры и топологии, поддерживающие работу по протоколу *TCP/IP*.

Теперь можно перейти непосредственно к методам передачи телевизионного трафика по *IP*. В зависимости от технологии передачи допускается использование трёх различных методов: *Unicast* (однаправленная), *Multicast* (групповая) и *Broadcast* (широковещательная).

Начать стоит с *Broadcast*, т.к. она применяется в основном либо в небольших локальных сетях, либо, когда серверу необходимо передать какую-нибудь информацию всем пользователям в сети. Главным минусом данного метода передачи стоит назвать то, что все пользователи будут получать один и тот же контент, а не тот, который они возможно хотели бы посмотреть. Также стоит учитывать, что большинство маршрутизаторов по умолчанию блокируют данный вид трафика. К плюсам можно отнести низкую нагрузку на сеть.

Следующий метод передачи телевизионного трафика активно применяется в *IPTV (Internet Protocol Television – IP-телевидение)* и называется *Multicast*. Данный метод предназначен для доставки данных группе абонентов и применяется при организации различных телетрансляций. При передаче телевизионного трафика создаётся поток, который затем передаётся только на те устройства, которые данный поток запрашивали. Главным плюсом является возможность существенно снизить требуемую полосу пропускания транспортной сети, т.к. не требуется создавать отдельный поток для доставки каждому запросившему контент зрителю. Минусом можно назвать то, что не все транспортные протоколы поддерживают данный метод передачи данных [2].

Наконец последним методом, применяемым в интернет-телевидении, является *Unicast*. Данный протокол осуществляет индивидуальную доставку трафика и используется для различных интерактивных сервисов или сервисов типа *VoD (Video on Demand – Видео по заказу)*. Плюсом данного метода является «личное» общение пользователя с сервером. Однако существенным минусом является сильная нагрузка на транспортную сеть, ведь передавая таким методам одинаковый трафик, мы создаём не один поток как в *Multicast*, а несколько одинаковых.

Принцип работы данных методов проиллюстрирован на рис. 1.

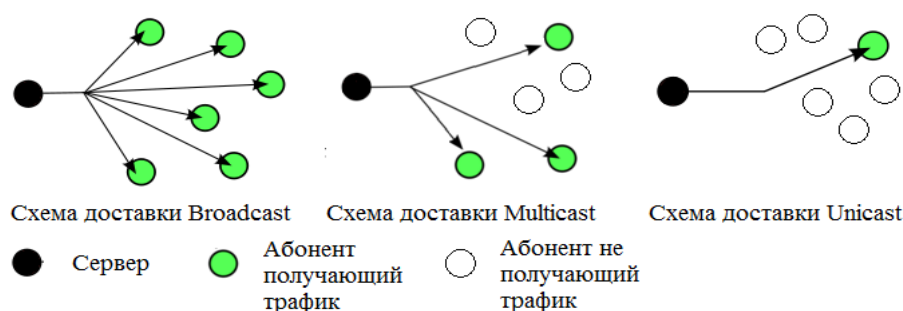


Рис. 1. Схема доставки контента разными методами

Теперь, разобравшись с методами доставки по сетям *IP* можно перейти непосредственно к технологии *OTT-TV* и методам, применяемых в ней.

*OTT* – формат вещания по неуправляемой сети Интернет, в котором активно применяются адаптивные протоколы вещания и методы масштабируемого кодирования. Так как средством доставки является Интернет, то каких-либо устройств для воспроизведения телевизионного трафика не требуется, необходимо только, чтобы устройство абонента поддерживало данную технологию или в некоторых случаях требуется скачивание видеоплеера.

Для распространения телевизионного контента между источником и получателем используется *HTTP*- (*TCP*) протокол, выполняющий транспортировку различного вида трафика от вещателя до потребителя в любой точке земного шара, где имеется возможность получить доступ в Интернет[1]. Существенным недостатком является то, что *TCP* не поддерживает передачу методом *Multicast*, поэтому используется *Unicast*, что существенно повышает нагрузку на транспортную сеть с каждым новым абонентом. Из-за этого многие интернет провайдеры негативно относятся к распространению данной технологии и планируют обязать платить *OTT* вещателей своеобразный налог на трафик, чтобы снизить издержки из-за высокой нагрузки на сеть. Поэтому в данной технологии активно внедряются новые методы видеокompрессии и во многих протоколах уже внедрена поддержка актуального на данный момент *H.265*.

Если рассматривать существующие и активно используемые в данный момент в *OTT* вещании технологии видеосжатия, то можно увидеть, что их не так много, это технология *H.264*, *H.265* и *VP9*. Последний на данный момент применяется в основном компанией *Google*, в то время как первые два поддерживаются основными протоколами адаптивного вещания.

Первым критерием при оценке видеокодера является качество видеокompрессии. Данный критерий показывает насколько сжатое изображение соответствует оригинальному, при оценке данного критерия обычно используют индекс структурного сходства (*SSIM* – *Structure similarity*). На рисунке 2 слева показана зависимость от битрейта индекса *SSIM*. [3]

Вторым критерием возьмём соотношение степени сжатия битрейта от времени кодирования. Он покажет сколько времени обычно требуется кодеру на обработку кадра, чтобы получить нужное качество. На рисунке 2 справа показано сравнение по данному критерию.

Как видно по графикам критерий *SSIM* у *VP9* и *H.265* примерно одинаков, однако также для них справедливо то, что с ростом степени сжатия существенно повышается требования к «железу» видеокодирующего устройства, в противном случае время сжатия для полноценного использования данных кодеков будет весьма существенным. При этом при схожем битрейте и относительно низкой степени сжатия пока что выгодней использовать хоть и более старый, но выигрывающий в данном плане *H.264*.

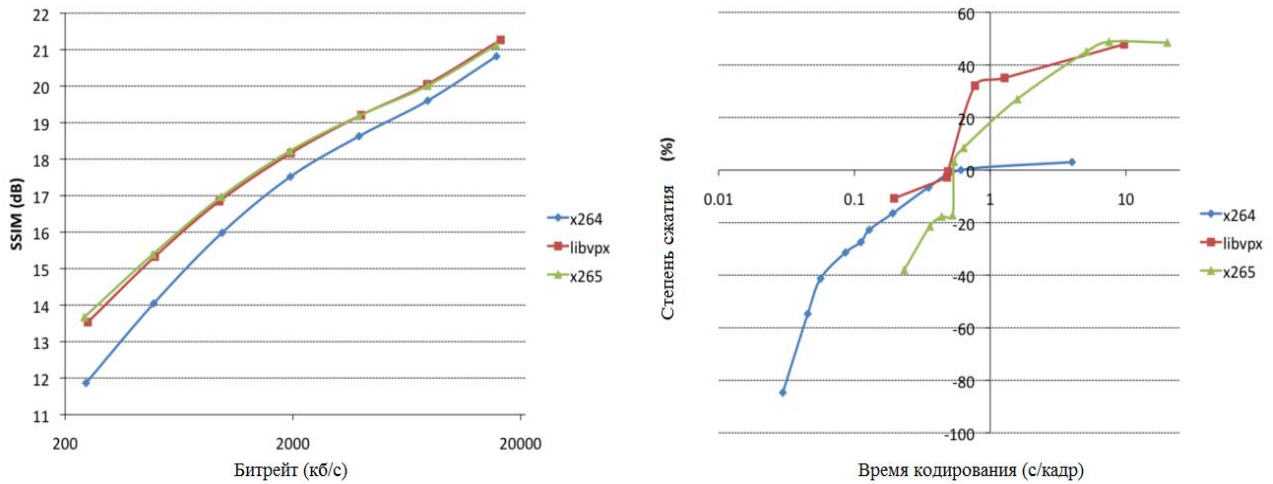


Рис. 2. Сравнение видеокодеков по SSIM (слева) и времени кодирования (справа)

Говоря о видеокомпрессии стоит помнить, что видеокодеки нужно также и декодировать, поэтому следующим пунктом будет сравнение кодеков по скорости декодирования. На рисунке 3 показано сравнение различных кодеков и их версий по скорости декодирования для обработки на одном ядром и для нескольких. Лучшие результаты показали уже хорошо оптимизированный кодек *H.264* и одна из реализаций кодека *VP9*, что явно говорит о том, что при необходимости декодирования в реальном времени. [3]

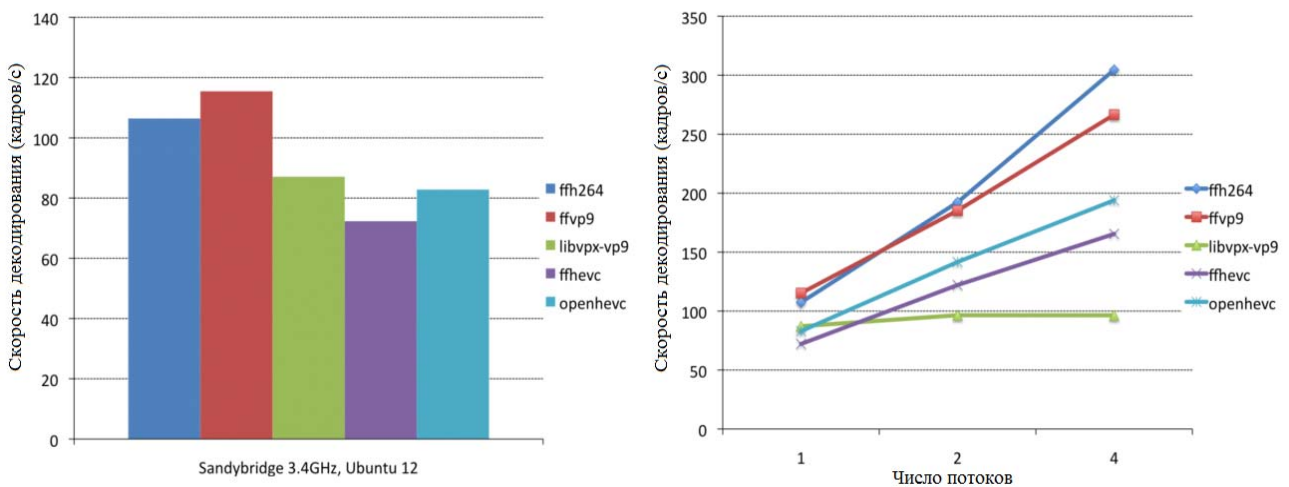


Рис. 3. Сравнение видеокодеков по скорости декодирования (слева) и скорости декодирования в зависимости от числа потоков (справа)

Проанализировав данные графики видно, что в ближайшее время кодек *H.264* вероятно будет оставаться актуальным, в то же время *VP9* хоть и показывает результаты лучше, чем *H.265*, но его популярность среди разработчиков адаптивных методов кодирования существенно ниже. *H.265* напротив уже поддерживается основными протоколами и при дальнейшей оптимизации кодеков и совершенствовании железа скорее всего плавно заменит популярный на данный момент *H.264*, что также позволит снизить нагрузки на транспортные сети провайдеров и уменьшить их претензии к *OTT* вещателям.

Теперь перейдём непосредственно к адаптивным протоколам вещания, применяемым при организации трансляций по технологии *OTT*. Сам принцип адаптивности описан на рис. 4.

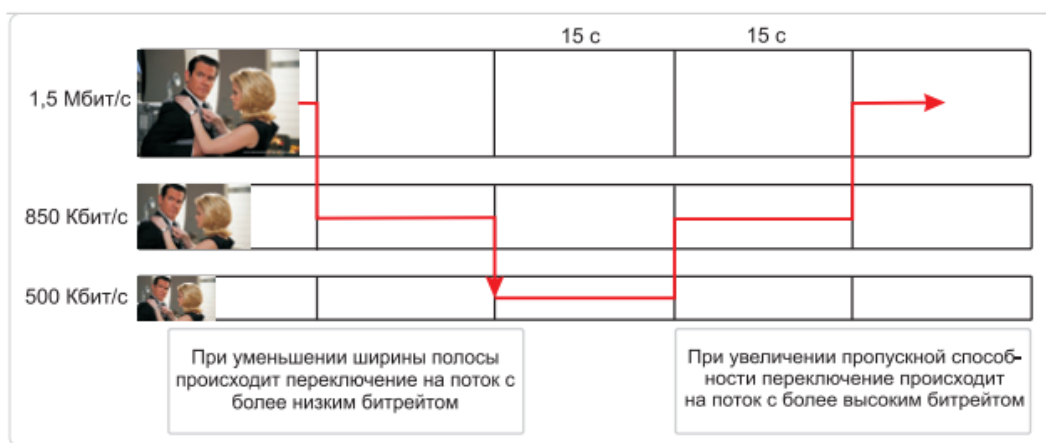


Рис. 4. Принцип адаптивного вещания

Существует пять протоколов адаптивного вещания: *Apple HLS (HTTP Adaptive Streaming)*, *Microsoft Smooth Streaming*, *Adobe HDS (HTTP Dynamic Streaming)*, *MPEG-DASH (Dynamic Adaptive Streaming over HTTP)*, *Google WebM*. Их принцип в основе своей схож, это формирование нескольких потоков видео, отличающихся качеством, передача пользователю плейлиста по которому пользователь запрашивает нужные ему фрагменты и при необходимости по нему же переключается на поток с более низким или высоким качеством в зависимости от текущей скорости абонента. Основными же отличиями является поддержка различных систем защиты контента (в основном каждая фирма продвигает свою технологию шифрования), платформ для воспроизведения и поддерживаемых видео-контейнеров и кодеков.

Например, протокол от *Microsoft* возможно воспроизвести только с помощью плеера их же разработки *Silverlight*. Аналогично протокол от *Adobe* нельзя воспроизвести без *Adobe Flash*, что существенно снизило их популярность по отношению к решениям, которые не привязаны к платформе и поэтому могут быть воспроизведены на практически любом проигрывателе.

Протокол от *Google*, ориентирован на сервисы на подобие *Youtube*, в тоже время наиболее популярные сейчас *Apple HLS* и *MPEG-DASH* поддерживает наиболее актуальные кодеки и контейнеры и могут быть использованы для любых трансляционных решений, что и сделало их популярными среди вещателей.

Сравнивать протоколы можно так же по поддерживаемым аудио- и видеокодекам, транспортным контейнерам и типам плейлист файлов можно увидеть в Таблице 1. По ней видно, что, не считая решений, привязанных к определённому проигрывателю, протоколы поддерживают либо *MPEG 2 TS*, либо у *WebM - Matroska*. Современные методы сжатия поддерживает опять же протоколы, не привязанные к определённой платформе. Но *WebM* продвигает собственные решения *Google* и при этом использует не требующие лицензионных отчислений форматы сжатия. Плейлист же у всех протоколов использует структуру *XML*, но отличается внутренним форматом.

Таблица 1

### Сравнение протоколов адаптивного вещания

Протокол вещания	Apple HLS	Microsoft Smooth Streaming	Adobe HTTP Dynamic Streaming	MPEG-DASH	Google WebM
Контейнер	MPEG-2 TS	PIFF	FLV	ISO, MPEG-2 TS, 3GP	Matroska
Видео и Аудио кодек	H.264, H.265 и AAC	H.264 и AAC	H.264 и VP6 и аудио кодеки AAC и MP3	H.264, H.265, VP8, VP9 и AAC	VP8, VP9 и Vorbis
Плейлист файл	m3u8	XML	XML	XML	-

Подводя итог можно сказать, что наиболее актуальными протоколами сейчас являются *Apple HLS* и *MPEG-DASH*, так как их принцип схож и у них нет жёстких требований к проигрывателю. Среди них стоит учитывать, что в отличие от протокола *Apple*, *MPEG DASH* поддерживает также не требующие лицензирования кодеки, что весьма полезно для небольших вещателей.

## Литература

1. *Broadcasting* Телевидение и радиовещание [Электронный ресурс]. – Режим доступа: <http://broadcasting.ru/articles2/Oborandteh/malenkie-sekrety-ott>. – Маленькие секреты *OTT*. – (Дата обращения: 17.04.2016).
2. Контур-М - мультисервисные сети передачи данных и телевидения: оборудование, проектирование и монтаж. [Электронный ресурс]. – Режим доступа: <http://www.konturm.ru/tech.php?id=iptvpe>. – Передача *Unicast*, *Broadcast* и *Multicast* трафика. – (Дата обращения: 18.04.2016).
3. *Ronald S. Bultje blog*. [Электронный ресурс]. – Режим доступа: <https://blogs.gnome.org/rbultje/2015/09/28/vp9-encodingdecoding-performance-vs-hevc-264/>. – *VP9 encoding/decoding performance vs. HEVC/H.264*. – (Дата обращения: 18.04.2016).
4. *Власюк И.В., Мазуркевич Д.О., Поташиников А.М.* Методика определения зависимости цветовой пороговой чувствительности от цветового тона // *Т-Comm: Телекоммуникации и транспорт*. 2009. № S1. С. 197-199.
5. *Безруков В.Н., Власюк И.В., Канев С.А.* Оценка относительного отклонения частоты при приеме сигналов наземных цифровых телевизионных вещательных систем // *Т-Comm: Телекоммуникации и транспорт*. 2010. Т. 4. № 6. С. 9-11.
6. *Власюк И.В., Сидорова А.И., Романова Е.П.* Особенности межкадрового кодирования видеоинформации по стандарту MPEG-4 // *Т-Comm: Телекоммуникации и транспорт*. 2010. Т. 4. № 9. С. 50-52.

# ИСПОЛЬЗОВАНИЕ СИСТЕМЫ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ MATLAB И SIMULINK В ЛАБОРАТОРНОЙ РАБОТЕ «ИССЛЕДОВАНИЕ ПАССИВНЫХ ЦЕПЕЙ ПРИ ГАРМОНИЧЕСКОМ ВОЗДЕЙСТВИИ НА ПОСТОЯННОЙ ЧАСТОТЕ»

*Кухтенко В.Ф.,  
МТУСИ, студент группы БИН1401  
Фриск В.В.,  
МТУСИ, к.т.н., доцент*

С развитием информационных технологий, в частности, вычислительной техники, у человека появляется все больше возможностей применить эти технологии и электронные вычислительные машины (ЭВМ) в различных сферах жизни и деятельности, тем самым значительно упростив себе жизнь. Не обошло это стороной и учебный процесс не только нашего ВУЗа, но и большинства учебных заведений по всему миру. С появлением интегральных микросхем и относительно дешевых и доступных ЭВМ стал возможен машинный анализ электрических цепей. Появились программы машинного моделирования электрических цепей, например, Micro-Cap, P-CAD, CircuitMaker и другие. В данной статье рассмотрена одна из таких программ – MATLAB и интегрированная в нее подпрограмма Simulink, которая позволяет моделировать электрические цепи.

MATLAB (MATrix LABoratory) – это высокоуровневый язык и интерактивная среда для программирования, численных расчетов и визуализации результатов. MATLAB позволяет относительно просто работать с матрицами действительных, комплексных и аналитических типов данных и с таблицами.

С помощью программ MATLAB и Simulink была выполнена лабораторная работа «Исследование пассивных цепей при гармоническом воздействии на постоянной частоте» [1]. Цель работы – рассчитать комплексные токи и падения напряжения в цепи, состоящей из резистивных и реактивных элементов при гармоническом воздействии, а также провести моделирование данной цепи. Сравнить результаты моделирования с аналогичными характеристиками, полученными расчетным путем.

Порядок выполнения лабораторных работ по ТЭЦ, следующий: студент, выполняющий лабораторную работу, начинает с изучения основных положений теории цепей по заданной теме, которые представлены в соответствующей литературе. Вся нужная литература представлена в конце описания каждой лабораторной работы. После внимательного изучения заданной темы, студент отвечает на контрольные вопросы (для самопроверки), которые также представлены в описании лабораторной работы, и приступает к ее выполнению. Выполнение работы состоит из следующих этапов: предварительный расчет, моделирование, сравнение результатов, полученных в предыдущих этапах и составление выводов о проделанной работе.

Цель, преследуемая данной статьей – показать возможность использования современной системы компьютерного моделирования MATLAB и Simulink в вышеназванной лабораторной работе, а также сравнение данной системы с похожей программой – Micro-Cap.

В первом пункте предварительного расчета требуется рассчитать в  $RC$ -цепи величины комплексных значений тока и падений напряжений на резистивном и реактивном элементах цепи, а также рассчитать их модули и фазы.

Рассмотрим электрическую цепь, состоящую из последовательно соединенных резистора  $R$  и конденсатора  $C$  ( $RC$ -цепь – резистивно-конденсаторная цепь). Ток  $I$  в такой  $RC$ -цепи рассчитывается согласно комплексному закону Ома для одиночного контура.

Закон Ома для комплексных величин для такой цепи рассмотрен в [1]. Комплексные падения напряжения на резисторе  $R$  и конденсаторе  $C$  вычисляются по формуле, приведенной в [1].

Программа MATLAB предоставляет возможность ввода формул любой сложности с мгновенным вычислением результата. Это значительно ускоряет расчет. Реализуется следующим образом: применительно к конкретной задаче имеется некоторый набор исходных данных, которые необходимо ввести в программу, присвоив численные значения этих данных буквенным обозначениям. Рассмотрим отрывок программного кода:

```
% RC-цепь
% Исходные данные
% ЭДС, В
E=1;
% Сопротивление, Ом
R=100;
% Частота, Гц
f=4*10^3;
% Емкость, Ф
C=0.3*10^-6;
% Емкостное сопротивление, Ом
XC=1/(2*pi*f*C);
% Ток в нагрузке, А
I=E./(R-j*XC)
% Модуль тока, А
MODI=abs(I)
% Фаза тока, град.
Fi=angle(I)*180/pi
% Фаза тока, рад.
Fi_rad=angle(I)
```

Символ «%» – обозначает комментарий, который выделяется зеленым цветом и хорошо виден в остальном программном коде, что позволяет, добавляя комментарии к определенным строкам кода, быстро в нем ориентироваться. Символ «=» – присваивает буквенному обозначению заданное численное выражение, либо число. Символ «^» – обозначает возведение в степень. Символ «\*» – умножение, остальные же символы имеют те же значения, хорошо известные нам из математики («+», «-», «/» – соответственно сложение, вычитание, деление). Как видно из данного отрывка кода формулы для расчета составляются из комбинаций вышеперечисленных символов, буквенных обозначений и численных значений.

В результате данного расчета получили: комплексный ток в RC цепи  $I=0,0036+j0,0048$  А, модуль комплексного тока  $|I|=0,006$  А, фаза комплексного тока  $\varphi_I = 52,9844^\circ = 0,9248$  рад. Все эти данные выводятся на экран ЭВМ (могут и не выводиться, это можно регулировать символом «;» в конце строки, результаты которой выводить не требуется) после ввода исходных данных и ввода формул.

Аналогичные расчеты следует провести для комплексных падений напряжений на элементах цепи: на резисторе  $U_R$  и на конденсаторе  $U_C$ . Далее необходимо построить графики данных величин. Построим график зависимости тока в RC-цепи от времени  $i(t)=I_m \sin(2\pi ft + \varphi)$ . Для этого введем соответствующую формулу в программу MATLAB, и при тех же исходных и уже найденных данных, используя специальную команду *plot(...)*, получим на экране графики интересующих нас значений.

Результаты всех проведенных расчетов заносятся в таблицу отчета в графу «По предварительному расчету». После проведения предварительного расчета и заполнения таблицы в соответствии заданием, можно переходить к проведению эксперимента.

В качестве эксперимента смоделируем данную цепь в программе Simulink и сравним результаты эксперимента с результатами предварительного расчета.

Simulink предоставляет широкие возможности для моделирования электрических цепей. В программе предусмотрена возможность установить измерительные приборы, такие как вольтметр, амперметр, фазометр. Вольтметр и амперметр – измеряют модуль комплексных значений соответственно тока и напряжения на элементах цепи. Необходимо отметить, что такая возможность имеется не во всех инженерных программах. В частности в программе Micro-Cap ее нет.

После сборки схемы с заданными величинами сопротивления, емкости и напряжения источника ЭДС, а также частоты (в соответствии с исходными данными) проводится эксперимент, в котором измеряются модули комплексных значений тока и напряжения. С помощью виртуального осцил-



лографа на экран ЭВМ выводятся графики напряжений и токов. Полученные данные заносятся в графу таблицы отчета «Получено экспериментально», а графики – в соответствующий раздел отчета.

Так как в результате эксперимента были получены модули комплексных величин тока и напряжений в цепи, то нужно провести обработку этих данных. Это делается снова в MATLAB, но так как Simulink – это подсистема MATLAB, то прибегать к использованию сторонних программ не требуется и обработка проводится быстро.

Для того чтобы получить комплексное значение тока и напряжений по их модулям – переведем значение тока из показательной формы в алгебраическую, для чего воспользуемся формулой Эйлера из [2].

В MATLAB для этого в качестве исходных данных требуется задать модуль комплексных значений тока и напряжений в цепи, а также их фазы, полученные при проведении эксперимента. После этого вводится формула Эйлера и программа выведет значение комплексного тока и напряжений.

В данном случае при заданных исходных данных было получено значение тока:  $I=0,0036+j0,0048$  А, которое нужно занести в таблицу отчета в графу «Рассчитано по экспериментальным данным». Те же вычисления проводятся и для напряжений на резисторе и конденсаторе.

После того как заполнена вся таблица отчета, можно сравнить данные, которые были получены в результате эксперимента и его обработки, и данные, полученные в предварительном расчете.

Если данные совпадают, то можно сделать вывод, что работа выполнена правильно, ошибок не допущено. Однако необходимо учесть, что результаты не обязательно будут в точности совпадать. Допускается небольшое расхождение в числах. Это объясняется в первую очередь погрешностью измерений (при эксперименте) и погрешностью вычислений. Но в целом больших расхождений быть не должно. Как видно из приведенного примера – данные совпали, а значит, работа была выполнена правильно.

Для сравнения тот же эксперимент был проведен с помощью программы Micro-Cap (Micro-Cap 11 Evaluation 11.0.1.6). Как уже отмечалось, в этой программе нет возможности установить измерительные приборы, такие как вольтметр и амперметр для синусоидального тока. А значит получить значения модулей комплексных токов и напряжений в цепи нельзя, поэтому здесь были получены только графики. Графики совпали с теми, которые получены в Simulink. Они хорошо иллюстрируют изменение значений тока и напряжений в цепи во времени. Однако к плюсам данной программы следует отнести простоту и быстроту сборки схемы. Эта программа сделана проще и может быть понятней, чем Simulink.

В результате проделанной работы были рассчитаны комплексные величины тока и напряжений в цепи, построены графики для наглядного представления тока и напряжения в цепи. Полученные результаты в предварительном расчете и результаты экспериментального расчета совпадают, отсюда вывод – работа выполнена верно.

Подведем итог сравнения Simulink и Micro-Cap. Simulink является более сложной системой моделирования электрических цепей. В Simulink имеется возможность установить вольтметр и амперметр, и измерять комплексные напряжения и токи, а также их модули и фазы. Так как Simulink[3] – подсистема MATLAB, то в ней имеется возможность производить математические расчеты и сразу получать результат, не прибегая к использованию сторонних программ. В Simulink сложнее выбрать нужный элемент схемы, так как представлено множество библиотек для одного и того же элемента. MATLAB и Simulink – мощные системы, решающие инженерные задачи, поэтому университету стоит продлить лицензию на их использование.

## Литература

1. Фриск В.В., Практикум № ML 1. Расчет и моделирование частотных характеристик RL-цепи в MATLAB и Simulink. М.: Брис-М, 2015. 32 с.
2. Фриск В.В., Основы теории цепей. М.: РадиоСофт, 2002. 288 с.
3. Дьяконов В.П., MATLAB R2007/2008/2009 для радиоинженеров. М.: ДМК Пресс, 2010. 976 с.
4. Frisk V. Researches of electric processes in a single wire line and in the flat condenser // T-Comm: Телекоммуникации и транспорт. 2014. Т. 8. № 1. С. 58-59.
5. Фриск В.В. Передача речи по однопроводной линии // T-Comm: Телекоммуникации и транспорт. 2015. Т. 9. № 7. С. 97-99.

# МЕТОД СПЕКТРАЛЬНОГО АНАЛИЗА АУДИО-СИГНАЛА FFT (БЫСТРОЕ ПРЕОБРАЗОВАНИЕ ФУРЬЕ)

Панкратов Ф.С., Жуков И.А.,  
МТУСИ, студенты группы БИК1404  
Ганин В.И.,  
МТУСИ, к.т.н., доцент

Дискретное преобразование Фурье (ДПФ, англ. *FFT*), на сегодняшний день, является одним из самых распространенных инструментов анализа аудио-сигнала. Однако, до появления компьютеров, ДПФ использовалось редко, поскольку вычисление только лишь 32-точечного ДПФ требует 1024 операции комплексного умножения и сложения. Первая программная реализация алгоритма БПФ была осуществлена в начале 60-х годов XX века Джеймсом Кули в вычислительном центре *IBM* под руководством Джона Тьюки. Это стало большим толчком к развитию алгоритма БПФ, многие программисты стали соревноваться в эффективной реализации алгоритма. Цель работы – изучение метода спектрального анализа *FFT*, программная реализация этого метода и создание устройства для визуализации результатов работы алгоритма.

Выражение для дискретного преобразования Фурье:

$$S(k) = \sum_{n=0}^{N-1} s(n) \cdot \exp\left(-j \cdot \frac{2\pi}{N} \cdot n \cdot k\right), \quad k = 0 \dots N - 1.$$

(1)

ДПФ  $N$  отсчетов сигнала  $S(n)$ ,  $n=0 \dots N-1$  (в общем случае комплексного) ставит в соответствие комплексных спектральных отсчетов  $S(k)$ ,  $k=0 \dots N-1$ . Для вычисления одного спектрального отсчета требуется  $N$  операций комплексного умножения и сложения. Таким образом, вычислительная сложность алгоритма ДПФ составляет  $N^2$  операций комплексного умножения и сложения.

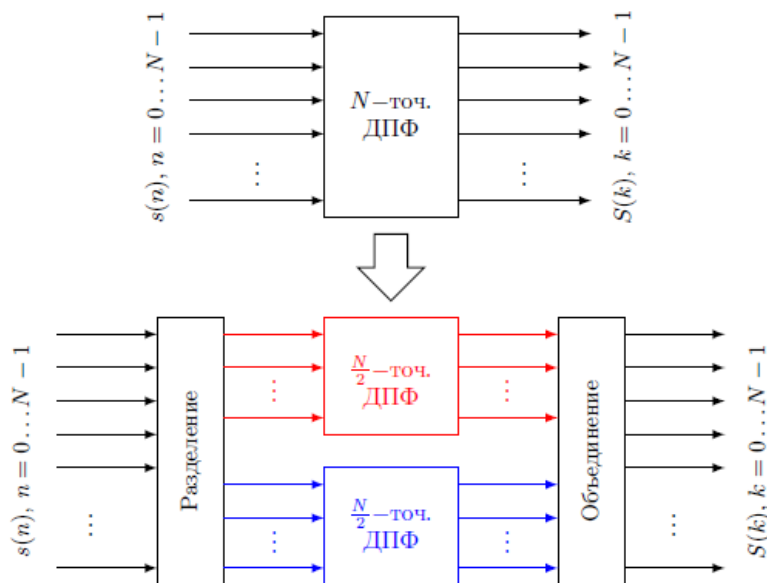


Рис. 1. Замена точечного ДПФ двумя точечными ДПФ

Можно достичь существенного ускорения вычисления если нам удастся свести расчет  $N$ - точечного ДПФ к двум точечным ДПФ, как это показано на рисунке 1.

Замена одного  $N$ - точечного ДПФ двумя  $\frac{N}{2}$  —точечными ДПФ приведет к уменьшению количества операций в 2 раза, но дополнительно требуются операции разделение последовательности на две и объединение двух  $\frac{N}{2}$  —точечных ДПФ в одно  $N$ -точечное.

При этом каждое из  $\frac{N}{2}$  точечных ДПФ также можно вычислить путем замены  $\frac{N}{2}$  точечного ДПФ на два  $\frac{N}{4}$  —точечных, которые в свою очередь можно рассчитать через  $\frac{N}{8}$  —точечные ДПФ. Эту рекурсию можно продолжать пока возможно разбить входную последовательность на две.

В нашем случае, если  $N = 2^L$ ,  $L$  это положительное целое, мы можем разделить последовательность пополам  $L$  раз. Для  $N=8$  ( $L=3$ ) такое разделение представлено на рис. 2.

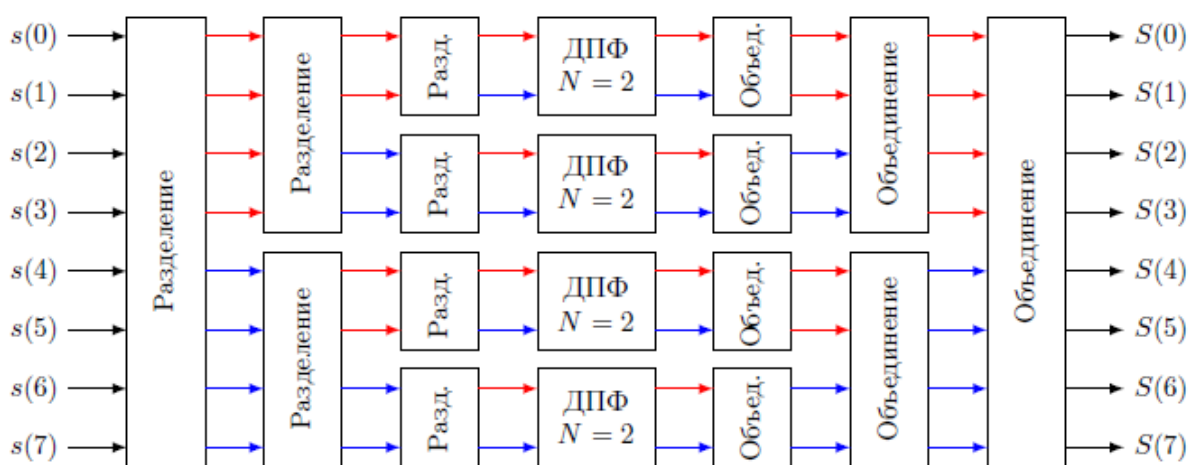


Рис. 2. Разделение и объединение последовательности для  $N=8$

Однако, у алгоритма ДПФ имеется одна важная особенность: если частота тона совпадает с одной из частот сетки ДПФ, то спектр будет выглядеть "идеально": единственный острый пик укажет на частоту и амплитуду тона. Если же частота тона не совпадает ни с одной из частот сетки ДПФ, то алгоритм "соберёт" тон из имеющихся в сетке частот, скомбинированных с различными весами. График спектра при этом размывается по частоте. Такое размытие обычно нежелательно, так как оно может закрыть собой более слабые звуки на соседних частотах.

Как будет выглядеть спектр тона на ДПФ-анализаторе? Оказывается, это зависит от частоты тона. Известно, что ДПФ раскладывает сигнал не по тем частотам, которые на самом деле присутствуют в сигнале, а по фиксированной равномерной сетке частот. Например, если частота дискретизации равна 48 кГц и размер окна ДПФ выбран 4096 отсчётов, то ДПФ раскладывает сигнал по 2049 частотам: 0 Гц, 11.72 Гц, 23.44 Гц, ..., 24000 Гц.

Чтобы уменьшить эффект размытия спектра, сигнал перед анализом умножается на весовые окна – гладкие функции, спадающие к краям интервала. Они уменьшают размытие спектра за счёт некоторого ухудшения частотного разрешения. Если рассматривать ДПФ как набор полосовых фильтров, то весовые окна регулируют взаимное проникновение частотных полос.

Простейшее окно – прямоугольное: это константа 1, не меняющая сигнала. Оно эквивалентно отсутствию весового окна. Одно из популярных окон – окно Хэмминга. Оно уменьшает уровень размытия спектра примерно на 40 дБ относительно главного пика.

Весовые окна различаются по двум основным параметрам: степени расширения главного пика и степени подавления размытия спектра ("боковых лепестков"). Чем сильнее подавляются боковые лепестки, тем шире будет основной пик. Прямоугольное окно меньше всего размывает верхушку пика, но имеет самые высокие боковые лепестки.

Для большинства задач не так важно, какой именно вид весового окна использовать. Главное, чтобы оно было. Использование весового окна уменьшает зависимость формы спектра от конкретной частоты сигнала и от её совпадения с сеткой частот ДПФ.

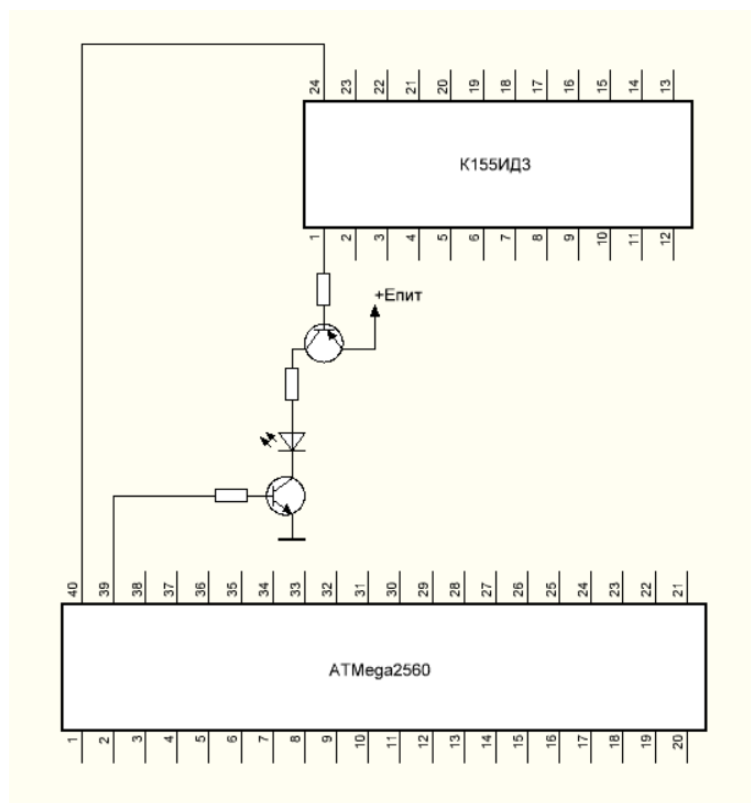
Чтобы компенсировать расширение пиков при применении весовых окон, можно использовать более длинные окна ДПФ: например, не 4096, а 8192 отсчета. Это улучшит разрешение анализа по частоте, но увеличит время вычисления.

Часто возникает необходимость проследить, как спектр сигнала меняется во времени. ДПФ-анализаторы помогают сделать это в реальном времени при воспроизведении сигнала. Однако в ряде случаев оказывается удобна визуализация изменения спектра во всём звуковом отрывке сразу. Такое представление сигнала называется спектрограммой. Для её построения применяется оконное преобразование Фурье: спектр вычисляется от последовательных окон сигнала, и каждый из этих спектров образует столбец в спектрограмме.

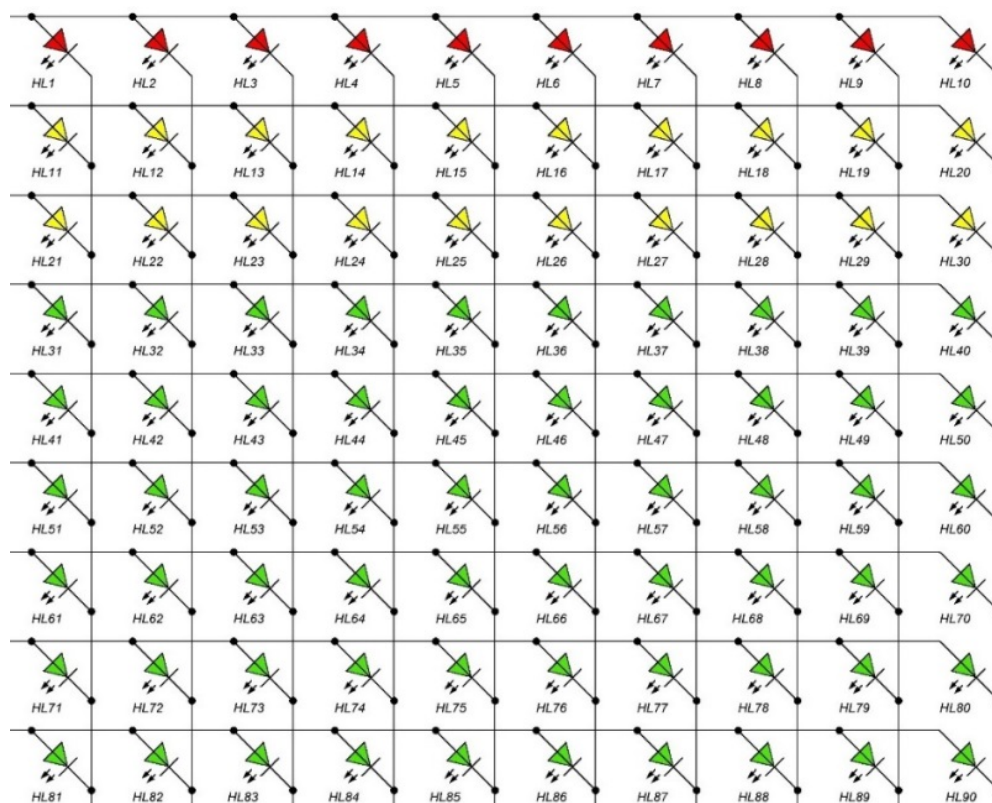
Особенно полезна спектрограмма при анализе быстро меняющихся сигналов. По ней легко определить такие характеристики голоса, как частота и глубина vibrato, его форма и ровность, наличие певческой форманты. По изменению высоты основного тона и гармоник прослеживается исполняемая мелодия. Современные средства реставрации звука, такие как программа *iZotope RX*, активно используют спектрограмму для редактирования отдельных частотно-временных областей в сигнале. С помощью этой техники можно найти и подавить такие нежелательные призвуки, как звонок мобильного телефона во время важной записи, скрип стула пианиста, кашель в зрительном зале и т.п.

Спектрограмма часто позволяет найти в записи дефекты, которые неочевидны при прослушивании, но могут сказаться при последующей обработке. Например, паразитная наводка от ЭЛТ-видеомонитора на частоте 15–16 кГц может ускользнуть от уха пожилого звукорежиссёра. Однако спектрограмма ясно покажет её в виде горизонтальной линии и позволит уточнить частоту для настройки режекторного фильтра.

Разработанное устройство состоит из двух частей: светодиодной матрицы и микроконтроллера *Arduino* на базе микропроцессора *AVR ATmega 2560*, производящего анализ входного аудиосигнала. Микроконтроллер позволяет выполнить спектральную развёртку по 64 частотам с 16 уровнями на каждой частоте. Светодиодная матрица состоит из 1024 светодиодов, управление которыми обеспечивают четыре дешифратора К155ИД3, собранных в каскад при помощи микросхемы К155ЛН1. Упрощённая схема управления одним светодиодом:



Светодиодная матрица вынесена на отдельную плату:



## Литература

1. *Лукин Алексей*: Спектроанализатор – что мы на нем видим. 2013 г. Режим доступа: <http://prosound.ixbt.com/education/spektr-analys.shtml>.
2. *Гольденберг Я.М., Матюшкин Б.Д., Поляк М.Н.* Цифровая обработка сигналов. 1995.

# ФИЛЬТРЫ В MATLAB

Тихомирова Е.О., Барков А.С.,  
МТУСИ, студенты группы БИН1408  
Степанова А.Г.,  
МТУСИ, ст. преподаватель

Одной из часто возникающих на практике задач является создание фильтров, пропускающих сигналы в определенной полосе частот и задерживающих остальные частоты. При этом различают:

- фильтры нижних частот (ФНЧ), пропускающие частоты, меньшие некоторой частоты среза  $\omega_0$ ;
- фильтры верхних частот (ФВЧ), пропускающие частоты, большие некоторой частоты среза  $\omega_0$ ;
- полосовые фильтры (ПФ), пропускающие частоты в некотором диапазоне  $\omega_1 \dots \omega_2$  (они могут также характеризоваться средней частотой  $\omega_0 = (\omega_1 + \omega_2)/2$  и шириной полосы пропускания  $\Delta\omega = \omega_2 - \omega_1$ );
- режекторные фильтры (РФ), пропускающие на выход все частоты, кроме лежащих в некотором диапазоне  $\omega_1 \dots \omega_2$ .

Идеальная форма АЧХ фильтров этих четырех типов показана на рис. 1.

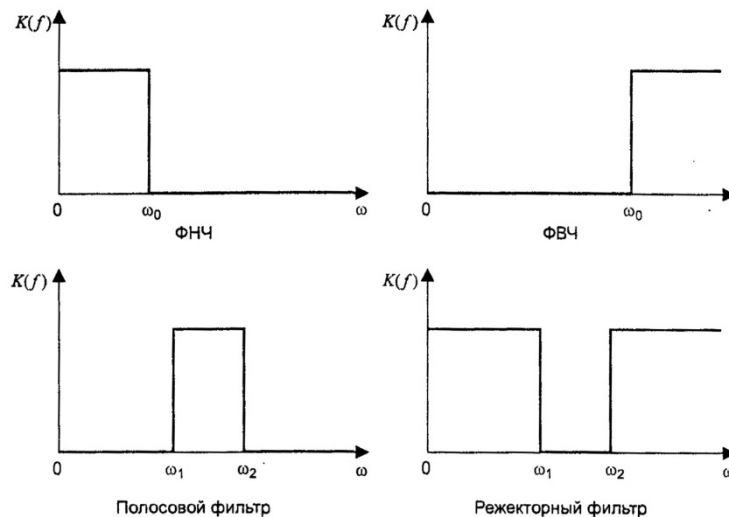


Рис. 1. АЧХ фильтров различного типа

## Фильтр Баттерворта

Функция передачи фильтра-прототипа Баттерворта не имеет нулей, а ее полюсы равномерно расположены на  $s$ -плоскости в левой половине окружности единичного радиуса (см. далее рис.2, а).

Благодаря такому размещению полюсов формула для АЧХ фильтра Баттерворта оказывается очень простой:

$$K(\omega) = \frac{1}{\sqrt{1 + \left(\frac{\omega}{\omega_0}\right)^{2n}}}$$

Коэффициент передачи на нулевой частоте равен 1, а на частоте среза независимо от порядка фильтра составляет  $1/\sqrt{2} \approx 0,707 \approx -3$  дБ. При  $\omega \rightarrow \infty$  АЧХ стремится к нулю.

АЧХ фильтра Баттерворта (см. рис.3) является *максимально плоской* при  $\omega = 0$  и  $\omega \rightarrow \infty$ . Это означает, что в данных точках равны нулю  $2n - 1$  производных АЧХ по частоте. В целом АЧХ монотонно спадает от единицы до нуля при изменении частоты от нуля до бесконечности.

В MATLAB расчет аналогового фильтра-прототипа Баттерворта производится с помощью функции `butter`:  $[z, p, k] = \text{butter}(n)$ ;

Входной целочисленный параметр  $n$  — это порядок фильтра.

На рис. 2 показано расположение полюсов фильтра Баттерворта 5-го порядка на комплексной плоскости, а также его АЧХ и ФЧХ:

```

» [z, p, k] = buttap(5);           % нули и полюсы прототипа
» plot(p, 'x')                   % расположения полюсов
» axis equal
» axis([-1.5 1.5 -1.5 1.5])
» w = 0:0.01:5;
» [b, a] = zp2tf(z, p, k);       % коэффициенты функции передачи
» h = freqs(b, a, w);           % комплексный коэффициент передачи
» figure
» plot(w, abs(h))                % график АЧХ
» grid
» figure
» plot(w, unwrap(angle(h)))      % график ФЧХ
» grid

```

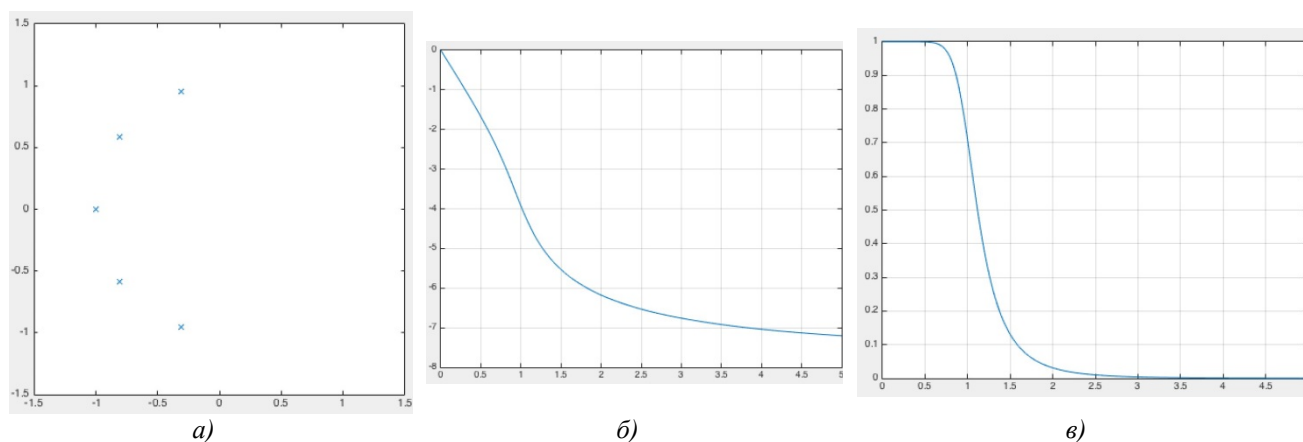


Рис. 2. Характеристики фильтра Баттерворта 5-го порядка: а – расположение полюсов на комплексной плоскости; б – АЧХ; в – ФЧХ

### Фильтр Чебышева

Функция передачи фильтра Чебышева также не имеет нулей, а ее полюсы расположены в левой половине эллипса на  $s$ -плоскости (см. далее рис. 3, а).

$$K(\omega) = \frac{1}{\sqrt{1 + \varepsilon^2 T_n^2(\omega / \omega_0)}}$$

АЧХ фильтра Чебышева описывается следующим образом:

Здесь  $\omega_0$  — частота среза,  $T_n(x)$  — полином Чебышева  $n$ -го порядка,  $n$  — порядок фильтра,  $\varepsilon$  — параметр, определяющий величину пульсаций АЧХ в полосе пропускания.

Полином Чебышева  $T_n(x)$  при  $|x| < 1$  колеблется в диапазоне  $-1 \dots +1$ , а при  $|x| > 1$  неограниченно возрастает по абсолютной величине. Поэтому АЧХ фильтра Чебышева первого рода в полосе пропускания (при  $|\omega| \leq \omega_0$ ) колеблется между значениями  $1/\sqrt{1 + \varepsilon^2}$  и 1, а вне полосы пропускания (при  $|\omega| > \omega_0$ ) монотонно затухает до нуля (см. далее рис. 3, б).

Коэффициент передачи на нулевой частоте равен 1 при нечетном порядке фильтра и  $1/\sqrt{1 + \varepsilon^2}$  — при четном. На частоте среза коэффициент передачи фильтра равен  $1/\sqrt{1 + \varepsilon^2}$ , то есть уровню пульсаций АЧХ в полосе пропускания. При  $\omega \rightarrow \infty$  АЧХ стремится к нулю.

По сравнению с фильтром Баттерворта того же порядка фильтр Чебышева обеспечивает более крутой спад АЧХ в области перехода от полосы пропускания к полосе задерживания.

Значение параметра  $\varepsilon$  и уровень пульсаций  $R_p$  (в децибелах) связаны следующим образом:

$$R_p = 20 \lg(\sqrt{1 + \varepsilon^2}) = 10 \lg(1 + \varepsilon^2) \text{ дБ,}$$

$$\varepsilon = \sqrt{10^{R_p/10} - 1}$$

При  $\omega \rightarrow \infty$  АЧХ фильтра Чебышева является максимально плоской.

В MATLAB фильтр-прототип Чебышева рассчитывается с помощью функции `cheb1ap`: `[z, p, k] = cheb1ap(n, Rp)`

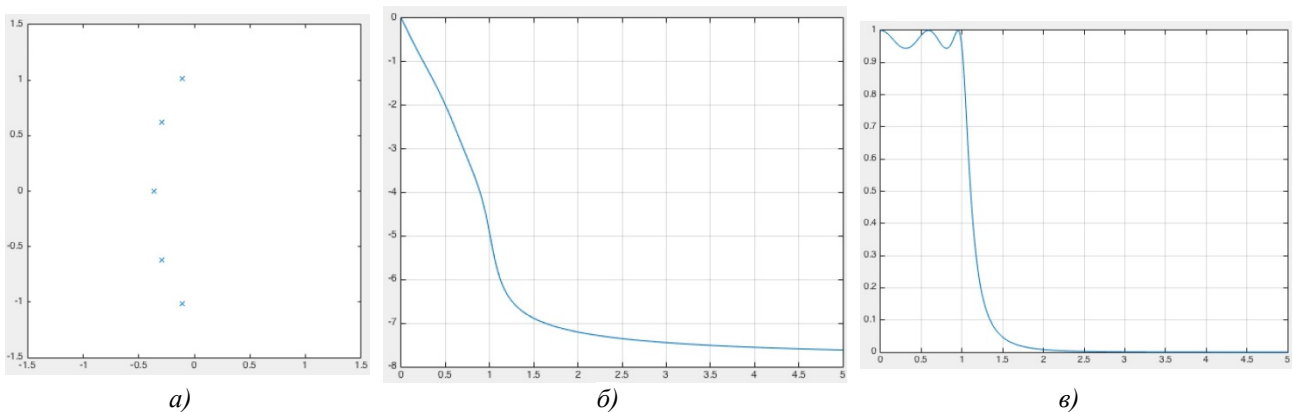
Здесь  $n$  — порядок фильтра,  $R_p$  — уровень пульсаций в полосе пропускания (в децибелах).

На рис. 3 показано расположение на комплексной плоскости полюсов фильтра Чебышева первого рода 5-го порядка с уровнем пульсаций 0,5 дБ, а также его АЧХ и ФЧХ.

```

» [z, p, k] = cheb1ap(5, 0.5)           % нули и полюсы прототипа
» plot(p, 'x')                          % график расположения полюсов
» axis equal
» axis([-1.5 1.5 -1.5 1.5])
» w = 0:0.01:5;
» [b, a] = zp2tf(z, p, k)               % коэффициенты функции передачи
» h = freqs(b, a, w);                   % комплексный коэффициент передачи
» figure
» plot(w, abs(h))                        % график АЧХ
» grid
» figure
» plot(w, unwrap(angle(h)))              % график ФЧХ
» grid

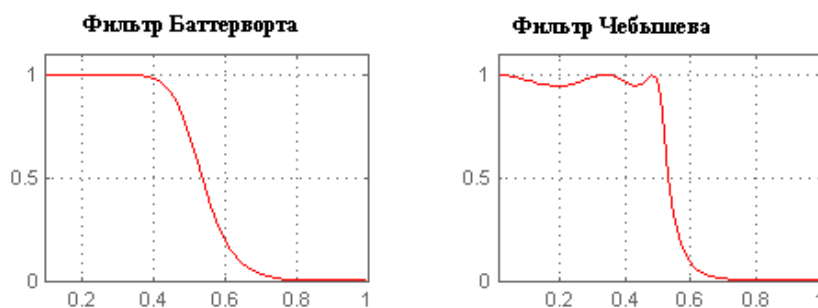
```



**Рис. 3.** Характеристики фильтра Чебышева первого рода 5-го порядка с уровнем пульсаций в полосе пропускания 0,5 дБ: а – расположение полюсов на комплексной плоскости; б – АЧХ; в – ФЧХ

### Вывод

АЧХ фильтра Баттерворта в сравнении с фильтром Чебышева одинакового (пятого) порядка (рис. 4).



**Рис. 4.** АЧХ фильтров

Видно, что спад АЧХ фильтра Баттерворта медленнее, однако он имеет и более гладкую АЧХ на частотах полосы пропускания.



## **Литература**

1. *Сергиенко А.Б.*, Цифровая обработка сигналов. СПб.: Питер, 2003. 608 с.

# ПРИМЕНЕНИЕ HIGH-K ПЛЁНОК В СОВРЕМЕННОЙ ЭЛЕКТРОНИКЕ

*Каледина А.В., Чехов А.С.,  
МТУСИ, студенты группы БИН1404  
Каравашкина В.Н.,  
МТУСИ, к.т.н., доцент*

В настоящее время, среди большого количества типов транзисторов особое внимание уделяют двум классам: *p-n* – биполярные транзисторы (переходные) и полевые или МДП (металл – диэлектрик – полупроводник) – транзисторы с изолированным полупроводниковым затвором. Принцип работы полевого транзистора заключается в возможности управлять протекающим через транзистор током при помощи электрического поля.

На сегодняшний день микроэлектроника вплотную подошла к атомарному рубежу. Это означает, что типичный размер транзистора составляет всего несколько десятков атомных слоев. С каждым днём уменьшение размеров транзисторов любого типа становится с каждым разом более проблематичным, что заставляет большинство компаний активно заниматься изучением и разработкой альтернативных технологий

Стартом развития всей электронной промышленности считается 1959 год, когда создали первую интегральную микросхему. С самого начала учёные стремились к увеличению производительности процессоров и пришли к выводу, что единственный возможный способ повысить скорость работы процессоров является уменьшение геометрических размеров транзистора, что позволяет понизить напряжение питания, вследствие чего сокращается энергопотребление и снижается рассеиваемое тепло, а также увеличение количества транзисторов в одной микросхеме открывает нам возможности в создании новых, более производительных процессоров.

Уменьшение размеров транзисторов влияет на такие его характеристики, как длина затвора, ширина затвора, толщина слоя диэлектрика. Толщина слоя диэлектрика определяет рабочее напряжение транзистора и его скоростные характеристики, поэтому нельзя уменьшать размеры транзистора при этом не изменяя толщину слоя диэлектрика в транзисторе.

Формула, определяющая слой диэлектрика между затвором и кремнием:

$$C_{ox} = \frac{\epsilon_0 \epsilon S}{t_{ox}}$$

где  $S$  – площадь затвора,  $t_{ox}$  – толщина слоя диэлектрика,  $\epsilon$  – диэлектрическая проницаемость слоя диэлектрика.

Емкость слоя диэлектрика определяется величиной тока, проходящего между стоком и истоком. Силу тока между стоком и истоком определяют при помощи следующей формулы:

$$I = \frac{1}{2} \mu C_{ox} \frac{W}{L_g} (V_{DD} - V_T)^2$$

где  $\mu$  – подвижность носителей заряда,  $W$  и  $L_g$  – ширина и длина канала проводимости,  $V_{DD}$  – управляющее напряжение затвора,  $V_T$  – напряжение срабатывания,  $C_{ox}$  – емкость подзатворной области.

Из формулы мы видим, что, если увеличивать емкость подзатворной области уменьшая при этом толщину слоя диэлектрика, то позволительно уменьшать управляющее напряжение на транзисторе (вследствие чего сократится энергопотребление), при этом оставляя силу тока между истоком и стоком неизменной.

Следовательно мы видим, что увеличение емкости подзатворной области за счет уменьшения толщины слоя диэлектрика положительно отражается на рабочих характеристиках транзистора, но уменьшение толщины слоя диэлектрика порождает и негативные последствия, которые являются причиной возникновения токов утечки.

При 65-нм технологическом процессе производства применение диоксида кремния в роли подзатворного диэлектрика ещё является возможным, однако, последующие уменьшения размеров транзисторов станет уже невозможным при применении данного материала

Для решения данной проблемы корпорация *Intel* произвела замену диоксида кремния на слой диэлектрика с высокой диэлектрической проницаемостью, который получил название – *High-K*-диэлектрик. Это позволило значительно понизить токи утечки.

Применение *High-K*- материалов предоставляет возможность применять более толстые слои диэлектрика при этом обеспечивая требуемые значения емкости подзатворного конденсатора.

При выполнении данного условия емкость затвора с диэлектриком из оксида кремния будет равна емкости затвора с *High-K*-диэлектриком:

$$\frac{\epsilon_o \epsilon_{ox} S}{d_{ox}} = \frac{\epsilon_o \epsilon_{high-K} S}{d_{high-K}}$$

Из полученной формулы следует, что для обеспечения равной емкости эквивалентная толщина на *High-K*-диэлектрика (*Equivalent Oxide Thickness, EOT*) равна:

$$EOT = d_{high-K} = \left( \frac{\epsilon_{high-K}}{\epsilon_{ox}} \right) \cdot d_{ox}$$

Поэтому, использование альтернативных материалов с более высокой диэлектрической проницаемостью позволяет во столько раз повысить толщину слоя диэлектрика по сравнению с толщиной диоксида кремния, во сколько раз диэлектрическая проницаемость вещества больше диэлектрической проницаемости диоксида кремния. А увеличение слоя диэлектрика ведёт к уменьшению токов утечки.

Почти 10 лет во всем мире проводились многогранные исследования свойств границы раздела большинства из этих оксидов с кремнием: влияние на пороговое напряжение, подвижность носителей в канале и пр. В результате сравнения диэлектриков наиболее перспективным было признано семейство материалов на основе гафния.

Прежде всего, была разработана технология получения однородных пленок с контролируемой толщиной, получившая название *ALD* — *Atomic Layer Deposition* (последовательное атомное осаждение). Этапы такой технологии иллюстрируются на примере *HfO2*. Смысл данной технологии заключается в том, что за один цикл обработки, который длится всего несколько секунд, на поверхности образуется только один слой молекул, благодаря чему толщину наносимой пленки можно регулировать почти с максимальной точностью просто числом таких циклов. Каждый цикл включает в себя две стадии осаждения: газовую фазу прекурсоров, которые являются химическими предшественниками осаждаемого вещества, и две продувки для удаления излишков. Прекурсоры выбираются так, чтобы всего один слой прекурсоров мог прилипнуть к подложке для первого осаждения или к предыдущему слою для последующих шагов процесса. На 3-м шаге ранее нанесенный монослой *HfCl4* обрабатывался парами воды, образовавшиеся при этом молекулы *HCl* испарялись с поверхности и на ней оставался очередной монослой двуокиси гафния.

Однако при использовании *HfO2* в роли подзатворного диэлектрика возник целый ряд проблем. Главная состояла в том, что данный материал оказался, не совместим с традиционным поликремниевым затвором, из-за сравнительно низкой температуры рекристаллизации, в результате чего возникал значительный разброс между различными транзисторами ИС по токам утечки и напряжению отсечки.

Теоретические и эмпирические исследования показали, что в *HfO2* и других *high-k* материалах внутри каждой молекулы усиливаются связи между металлом и кислородом, которые определяют высокую степень поляризации, в результате чего возникает резонанс, приводящий к резкому падению подвижности носителей в канале.

Для решения данной проблемы при использовании *high-k* материалов в роли подзатворного диэлектрика необходимо в затворе заменить поликремний на металл, в котором концентрация носителей намного больше. В этом случае описанный резонанс не наступает. Однако для *p +* или *n +* канальных МДП-транзисторов пришлось использовать различные металлы с «правильными» работами выхода.

При нанесении двух металлов возрастает число требуемых литографических циклов, а технологический маршрут усложняется и становится более дорогим. Первоначально было предложено два варианта такого маршрута. Представители *Intel* вначале формируют простой поликремниевый затвор, который работает в качестве маски для создания стока и истока, далее вытравливают его, затем осаждают сплав для *p*-каналов, после чего удаляют его из *n*-транзисторов, потом осаждают сплав для *n*-каналов и добавляет ко всем затворам алюминиевый наполнитель. Описанный вариант получил на-

звание «затвор последним» или «замещающий затвор». *IBM* и сотрудничающие с ней фирмы до последнего времени использовали другую технологию, которая называется «затвор первым».

Этот метод повышает процент выхода годных изделий.

Сейчас благодаря *high-k* плёнкам, открываются огромные возможности в разработке и производстве центральных процессоров. Например компания *Intel* в своём новейшем семействе чипов *Skylake* использует уже 14 нм технический процесс, что позволяет снизить Термопакет (*TDP*) (величина, которая показывает, на отвод какой тепловой мощности должна быть рассчитана система охлаждения процессора или другого полупроводникового прибора). За счет этого процессоры могут широко использоваться в портативных устройствах.

Также в процессорах таких ведущих компаний как *AMD* и *Qualcomm* используется 14 нм технический процесс. А, например, компания *IBM* уже говорит, о разработке новейшего вида сверхплотного чипа, который позволит обеспечить производительность в несколько раз выше той, что обеспечивают нынешние процессоры. Сообщается, что в *IBM* нашли способ производства 7-нм транзисторов.

*High-k* материал активно используется в накопителях нового поколения. В наши дни емкость общедоступных жестких дисков достигает нескольких терабайт, в своё время как емкость *SSD* составляет лишь 250–500 Гбайт. Нынешняя планарная флеш-память *NAND* во многом исчерпала свой потенциал. Главная проблема *NAND* заключается в том, что уменьшать размеры кристалла с каждым разом становится все труднее. На смену ей разрабатывается и внедряется технология «вертикальной» флеш-памяти – *3D NAND*.

Особенностями *3D NAND* является то, что ячейки размещаются не только в плоскости, но еще и слоями. Благодаря этому чип получает трехмерную структуру и имеет возможность вмещать внушительно больше информации на единицу площади, в отличие от двухмерных кристаллов. Компания *Samsung* продемонстрировала концепт трехмерной памяти *V-NAND*, включающая в себя 24 слоя ячеек флеш-памяти. Технология изготовления *V-NAND* практически ничем не отличается с технологией изготовления *Bit-Cost Scalable Flash (BiCS)* разработки *Toshiba*. *BiCS* задерживает электрические заряды на записываемом слое из нитрида кремния, который окружен двумя слоями окисной пленки. В свою очередь ячейка окружена управляющим затвором, который выводит электрический заряд со слоя, производящий запись.

*V-NAND* работает на таком же принципе, но материалы слоев существенно другие: записывающий слой окружен алюминиевым диэлектриком, который выполнен по технологии *high-k*. По заявлению *Samsung*, использование нитрида тантала в управляющем затворе позволяет быстрее стирать ячейки.

*3D V-NAND* память может похвастаться заметно лучшей скоростью работы. Это достигается с помощью упрощения алгоритма записи в ячейку. Вместо трех операций теперь выполняется всего одна. Упрощение алгоритма сделалось возможным по причине уменьшенной интерференции между ячейками. Планарная память требует дополнительный анализ перед записью из-за возможных помех между соседними ячейками. Вертикальная память освобождена от этой проблемы, и запись спокойно выполняется всего лишь за один шаг.

Так же *3D V-NAND* память намного меньше подвержена износу благодаря тому, что для записи информации в ячейку не требуется высокое напряжение. Например, для размещения данных в ячейку с планарной памятью применяется напряжение около 20 В. В свою очередь для трехмерной памяти этот показатель ниже. На надёжности памяти благоприятно отразился тот факт, что создание трехмерной флеш-памяти не требует тонких технологических норм. Сейчас уже третье поколение *3D V-NAND* с 48 слоями изготавливается по 40 нм техпроцессу.

## Литература

1. Денисенко В.В. Новые физические эффекты в нанометровых МОП-транзисторах.
2. R. Chau et al. Gate Dielectric Scaling for High-Performance CMOS: from SiO<sub>2</sub> to HighK.
3. Сергей Пахомов. Нанотехнологии на службе Intel.
4. Елизаров А.А., Каравайкина В.Н., Нестерова Д.А., Шаймарданов Р.В. Моделирование микроволнового излучателя на основе коаксиального ребристого стержня // Т-Comm: Телекоммуникации и транспорт. 2014. Т. 8. № 10. С. 24-26.
5. Елизаров А.А., Каравайкина В.Н. Конструктивно-технологические особенности устройств на замедляющих системах с аномальной дисперсией // Т-Comm: Телекоммуникации и транспорт. 2013. Т. 7. № 9. С. 77-81.

# РАЗРАБОТКА СИСТЕМЫ ФОРМИРОВАНИЯ ГРУПОВОГО СЕАНСОВОГО КЛЮЧА ДЛЯ ГРУППЫ ОБЪЕКТОВ VDL-4

Дуплищева Я.В.,  
МТУСИ, студентка группы БЗС1201  
Шаврин С.С.,  
МТУСИ, д.т.н., профессор

В последнее время проблема обеспечения безопасности в авиационной сети связи особенно актуальна. Для решения данной и иных проблем была предложена новая система управления полетами – АЗН-В (автоматическое зависимое наблюдение-вещание). Одной из реализацией данной системы является протокол VDL Mode 4.

VDL Mode 4 является цифровым каналом передачи данных, функционирующим в УКВ-диапазоне частот. Он обеспечивает цифровую связь между мобильными станциями (летательными аппаратами, автомобилями в аэропорте) и стационарными наземными станциями. Его уникальной характеристикой является то, что доступное время передачи делится на большое число кратких временных слотов, синхронизированных на основе глобального времени. Каждый слот может быть использован радиотранспондером для передачи данных. Точная синхронизация слотов и запланированное их использование для передачи известны всем пользователям, находящимся в зоне видимости друг от друга, что делает эффективным использование канала передачи данных, и пользователи не ведут передач одновременно. В результате для работы в режиме VDL Mode 4 не требуется никакой наземной инфраструктуры, а потому система может обеспечивать связь и другие функции, как в режиме «борт-борт», так и режиме «земля-борт» [1].

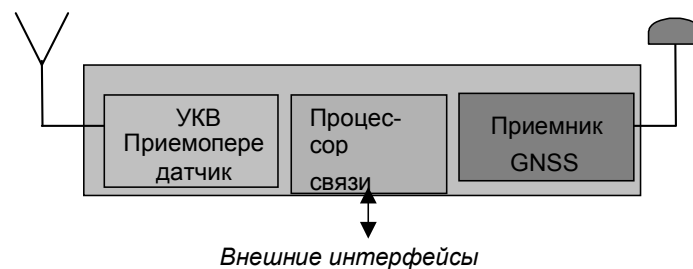


Рис. 1. Транспондер VDL Mode 4

На рис.1 показана структурная схема базового транспондера VDL Mode 4, которая одинакова для бортовых и для наземных пользователей.

Кратко формат сигнала VDL Mode 4 можно описать следующим образом. Интервал времени длительностью 1 секунда делится на 75 слотов, длительностью 13,33 мс. Из них часть времени уходит на нарастание и затухание мощности сигнала. Доля информационной части сообщения уходит на передачу 24 бит синхронизации и по 8 бит уходит на флаги, обозначающие начало и конец сообщения (см. рис. 2).

В рамках протокола VDL Mode 4 была поставлена задача обеспечить организацию стабильного защищенного канала связи между группой объектов. В рамках исследования рассматривается его организация по подобной схеме:

1. Формирование сеансового ключа с помощью ассиметричного алгоритма обмена ключами.

2. Шифрование абонентами данных с помощью симметричного алгоритма шифрования полученным ключом и установление сеанса связи.

Основной проблемой использования иных схем организации защищенного канала связи (например, использование организации связи между объектами по принципу «каждый с каждым») является использование и резервация экстремально большого количества слотов для передачи ключей, что ведет к большим задержкам и, в худшем случае, к разрыву сеанса связи. Отсюда можно сделать вывод, что для разработки системы формирования группового сеансового ключа основным критерием выбора алгоритма обмена ключами будет являться максимальное количество абонентов, которым может быть передано сообщение за одну секунду.

#### ФОРМАТ СИГНАЛА VDL MODE 4

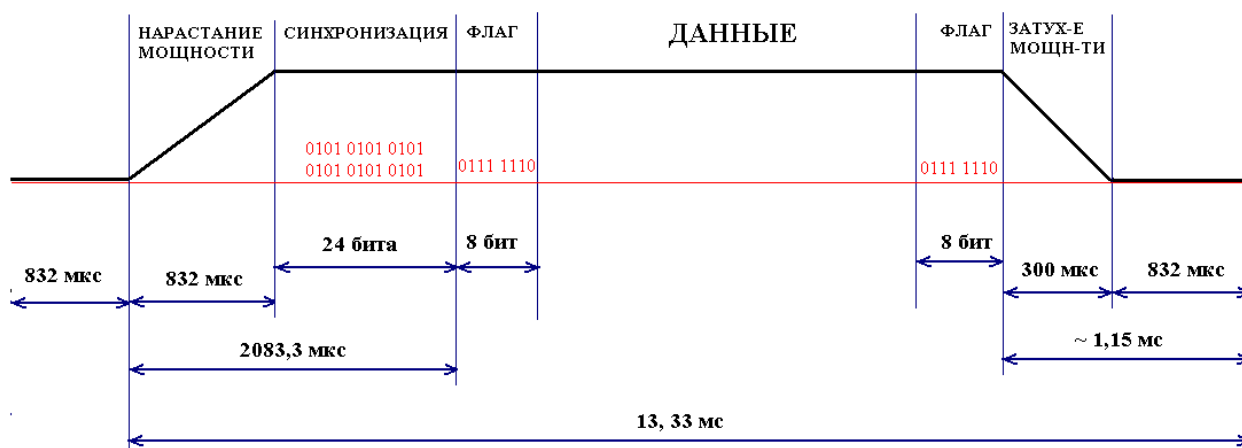


Рис. 2. Формат сигнала VDL Mode 4

Используя структуру секундного фрейма, можно вывести эмпирическую формулу, описывающую основной критерий для расчета количества слотов при передаче одного сообщения:

$$160 + n \cdot 256 = K, \quad (1)$$

где  $n + 1$  – количество занятых слотов,  $K$  – длина ключа (бит). Преобразуем формулу (1) и определим количество занятых слотов для передачи, к примеру, 2048-битного ключа:

$$160 + n \cdot 256 = 2048, \quad n = \frac{2048 - 160}{256},$$

$n = 7.375 \approx 8$ , тогда количество занятых слотов при передаче 2048-битного сообщения в одну сторону будет равняться 9.

Адрес станции VDL Mode 4 является ее универсальным идентификатором. Он состоит из 24-битного введенного в ICAO адреса воздушного судна и трех добавочных битов. Поскольку каждый идентификатор регистрируется в ICAO, не возникает необходимости использовать громоздкие алгоритмы обмена ключами с проверкой подлинности, т.к. проверку подлинности идентификатора станции можно устроить при ее начальном обнаружении.

В процессе выбора оптимального алгоритма формирования ключа были подробно проанализированы самые известные из них [2]. Для дальнейшего анализа были выбраны те, которые возможно реализовать в условиях протокола VDL Mode 4. Результат сведен в табл. 1.

Среди всех алгоритмов наиболее привлекательной выглядит одна из вариаций алгоритма Диффи-Хеллмана – алгоритм Хьюза. Ведущий (абонент, раздающий ключ) вычисляет ключ до установления соединения и передает ключ каждому абоненту сети по отдельности после прохождения процедуры идентификации. Именно этот алгоритм и будет использован в дальнейшем при создании ключа.

**Результат расчета количества слотов VDL-4, задействованных при передаче 2048-битного ключа между тремя абонентами**

Название алгоритма	Количество занятых слотов, задействованных при передаче 2048-битного ключа
Diffie-Hellman	72
Hughes	54
Обмен ключом без обмена ключом	56
Протокол «точка-точка»	72
Трехпроходный протокол Шамира	56
Расширенный протокол ЕКЕ (А-ЕКЕ)	>75

Считая абонента А ведущим, приведем подробное описание алгоритма Хьюза:

1. Выбираются два больших открытых целых числа  $g$  и  $n$ , причем  $1 < g < n$ . Важно, чтобы числа  $n$  и  $\frac{(n-1)}{2}$  были простыми, и чтобы число  $g$  являлось примитивным корнем по модулю  $n$ .

2. Абонент А выбирает случайное большое целое число  $x$  и генерирует:

$$k = g^x \bmod n.$$

3. Абонент В выбирает случайное большое целое число  $y$ , вычисляет и посылает абоненту А:

$$Y = g^y \bmod n.$$

4. Абонент А вычисляет и посылает абоненту В:

$$X = Y^x \bmod n.$$

5. Абонент В вычисляет:

$$z = y^{-1} \bmod (n-1),$$

$$k' = X^z \bmod n.$$

В итоге у обоих абонентов имеется один и тот же ключ  $k = k' = g^x \bmod n$ . Схема реализации алгоритма Хьюза для двух абонентов приведена на рис. 3.

Доказательство, что все абоненты группы (В и С) получают от ведущего А один и тот же ключ:  
Для абонента В с закрытым ключом у:

$$k = g^x \bmod n, Y = g^y \bmod n.$$

$$X = Y^x \bmod n \rightarrow X = (g^y \bmod n)^x \bmod n.$$

$$\begin{aligned} z = y^{-1} \bmod (n-1) \rightarrow k &= X^z \bmod n = ((g^y \bmod n)^x \bmod n)^z \bmod n = \\ &= ((g^y \bmod n)^x \bmod n)^{y^{-1} \bmod (n-1)} \bmod n = (g^{yx} \bmod n)^{y^{-1} \bmod (n-1)} \bmod n \\ &= \\ &= g^{yxy^{-1} \bmod (n-1)} \bmod n = g^x \bmod n. \end{aligned}$$

Для абонента С с закрытым ключом d:

$$k = g^x \bmod n, Y = g^d \bmod n.$$

$$X = Y^x \bmod n \rightarrow X = (g^d \bmod n)^x \bmod n.$$

$$\begin{aligned} z = d^{-1} \bmod (n-1) \rightarrow k &= X^z \bmod n = ((g^d \bmod n)^x \bmod n)^z \bmod n = \\ &= ((g^d \bmod n)^x \bmod n)^{d^{-1} \bmod (n-1)} \bmod n = (g^{dx} \bmod n)^{d^{-1} \bmod (n-1)} \bmod n \\ &= \\ &= g^{dx d^{-1} \bmod (n-1)} \bmod n = g^x \bmod n. \end{aligned}$$

В итоге у всех абонентов группы имеется один и тот же ключ  $k = g^x \bmod n$ .

Основными достоинствами выбранного алгоритма являются:

1. Наименьшее количество задействованных слотов.
2. Возможность отправки ключа всем абонентам в любом порядке.
3. Независимость ключа, созданного ведущим, от закрытых параметров других абонентов.

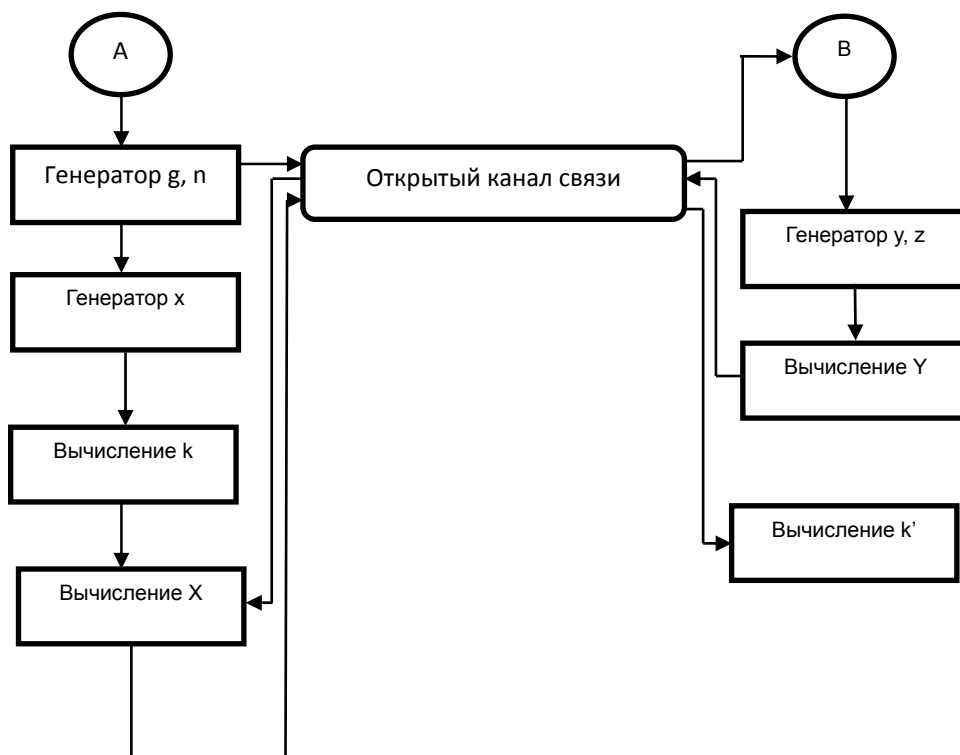


Рис. 3. Схема реализации алгоритма Хьюза для двух абонентов

В качестве процессора связи в радиотранспондере VDL Mode 4 выбран сигнальный процессор фирмы Analog Devices ADSP-2181. В ПЗУ у каждого абонента хранится база подходящих по критериям значений  $n, g, x$  для использования их в процессе создания сеансового ключа. При включении транспондера в ОЗУ процессора загружается необходимая информация и выполняется модуль программы в соответствии с выполняемой на данный момент ролью (раздает станция ключ в этот момент времени или принимает).

Оптимальная длина создаваемого сеансового ключа для достаточного удовлетворения потребности в защищенности канала VDL Mode 4 составляет 64 бита, поэтому в процессе написания программы алгоритма Хьюза используется расширенное поле Галуа  $GF(2^n)$ . Это позволяет существенно упростить процедуры возведения в степень, умножения и нахождения обратного по модулю числа.

## Литература

1. Manual on VHF Digital Link (VDL) Mode 4 – ICAO Doc 9816 AN/448, First Edition, 2004. - 406 p.
2. Schneier B., Applied cryptography: protocols, algorithms, and source code in C. Second edition. John Wiley & Sons. 1996.
3. Клёсова Ю.В., Шаврин С.С. Компьютерное моделирование метода распознавания речи для организации речевого канала в авиационных системах // Т-Comm: Телекоммуникации и транспорт. 2016. Т. 10. № 5. С. 54-59.
4. Душин С.В., Шаврин С.С. Влияние параметрических изменений эхотракта на работу взаимокорреляционного эхокомпенсатора // Т-Comm: Телекоммуникации и транспорт. 2011. Т. 5. № 8. С. 47-49.



# МАКЕТ ФРАГМЕНТА СЕТИ ДОСТУПА GEAPON БАЗОВОЙ КАФЕДРЫ "РОСТЕЛЕКОМ" МТУСИ

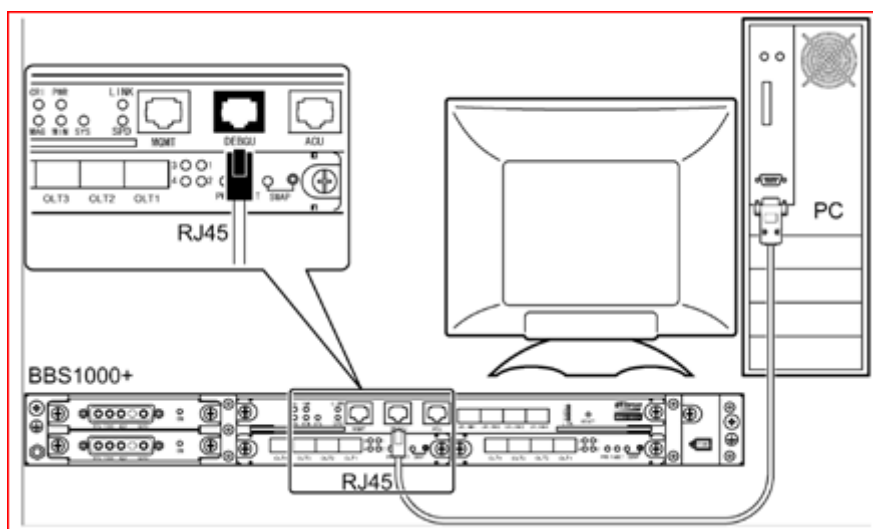
*Демин И.П., Жуков В.Ю., Киселев А.С., Смирнов С.И.,  
МТУСИ, студенты группы БОС1301  
Джалалов И.К.,  
МТУСИ, к.т.н., доцент*

Пассивная оптическая сеть Gigabit Ethernet (Gigabit Ethernet Passive Optical Network, GEAPON) является вариантом оптоволоконной сети до дома (FTTH - Fiber To The X), обеспечивающей передачу данных сети Ethernet со скоростью до 1 Гб/с, а также, дополнительно, обмен видео- и голосовой информацией по протоколу IP (VoIP).

Система GEAPON использует одно физическое волокно для передачи данных в обоих направлениях. Исходящий трафик передается на длине волны 1490 нм, а входящий на длине волны 1310 нм. Дополнительно длина волны 1550 нм зарезервирована для возможных дополнений в виде кабельного телевидения и спектрального уплотнения сигналов (WDM).

Оборудование фирмы NEC, смонтированное в лаборатории базовой кафедры ПАО «Ростелеком», обеспечивает симметричные решения GEAPON для создания мультисервисных услуг внутри систем оптоволоконной сети типа FTTH.

Основными компонентами системы GEAPON являются OLT и ONU. Вся передача данных выполняется между OLT и ONU. ONU оснащено системой временного уплотнения сигналов (TDM) с разделением по времени доступа к среде.



*Рис. 1. Схема подключения для настройки головной станции OLT*

При изучении макета были получены навыки работы с оборудованием пассивных оптических сетей доступа. Для проведения экспериментов, необходимо было предварительно подключить и настроить головную станцию OLT (optical line terminal).

Затем подключить абонентские станции ONU (optical network unit) к головной станции OLT по оптической сети.

И уже после этого имея возможность подключения в разрыв оптической линии ONU различных приборов, проводить измерения, полезные в лабораторных работах кафедры МТС.

Для разрыва оптической линии воспользуемся оптической розеткой и подключим через пачкорды оптический аттенуатор:

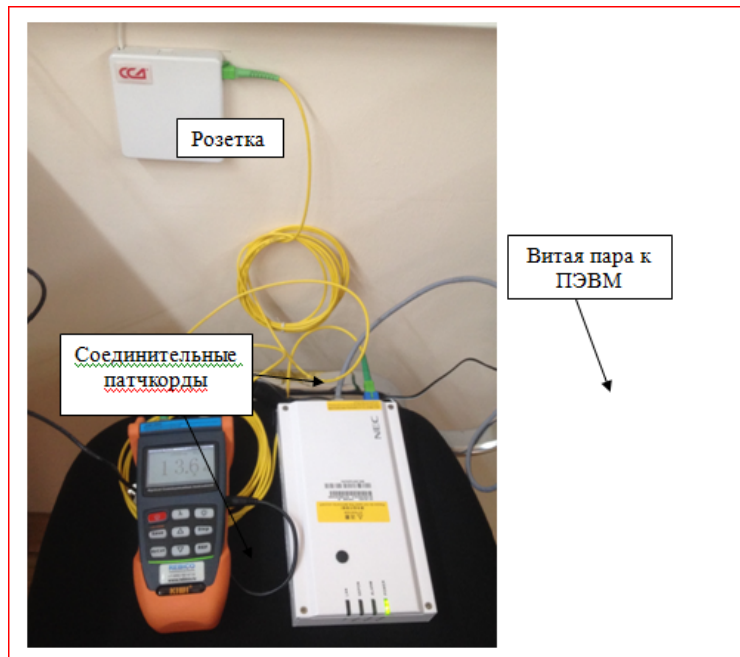


Рис. 2. Пример подключения оптического аттенюатора в разрыв линии

Установив аттенюатор в режим минимального ослабления, включаем тестовый трафик. На ПЭВМ открываем командную строку (cmd.exe) и включаем тестовый поток данных командой:

```
#ping 192.168.100.1 -l 500 -t
```

Если канал в рабочем состоянии, то приходит ответ от устройства с IP адресом 192.168.100.1.

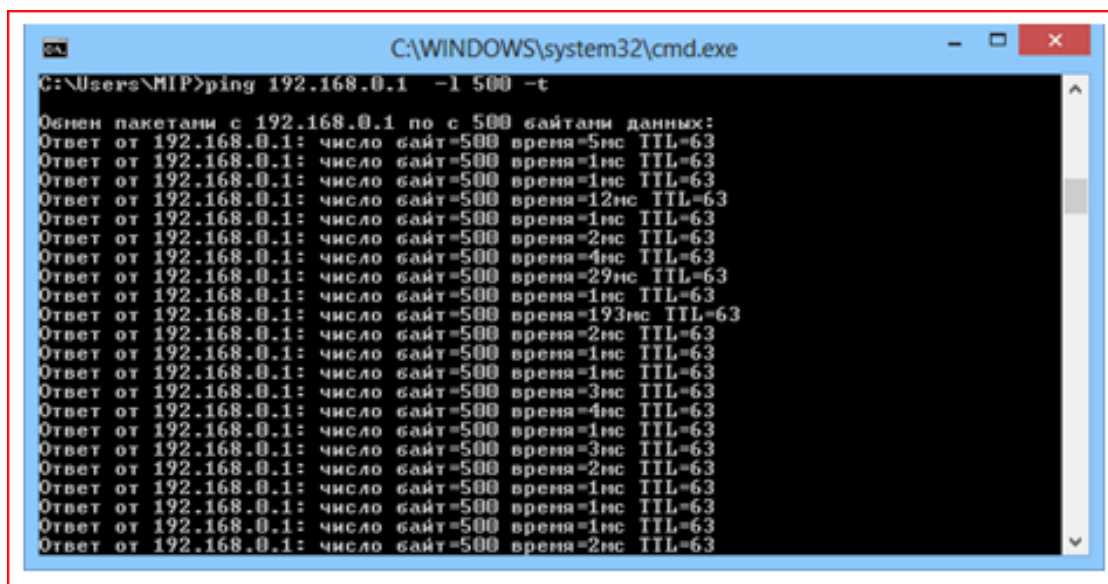


Рис. 3. Пример ответа от удаленного устройства

Увеличивая затухание аттенюатора, добиваемся прекращения прохождения сигнала, т.е. ответ от удаленного устройства перестанет приходить.



# ПОСТАНОВКА ЛАБОРАТОРНЫХ РАБОТ НА БАЗЕ ОБОРУДОВАНИЯ SI3000 КОМПАНИИ ISKRATEL

Вилков А.Р., Касанов К.В.,  
МТУСИ, студенты группы БСС1201  
Маликова Е.Е.,  
к.т.н., доцент

Данная публикация посвящена постановке лабораторных работ на макете компании *Iskratel*, который установлен в лаборатории Л-411 кафедры Сетей связи и систем коммутации. Данный макет предназначен для изучения студентами нашего университета принципов построения сетей следующих поколений и получения практических навыков при работе на этих сетях.

Функциональная схема макета на базе *Iskratel SI3000 IMS* представлена на рис. 1.

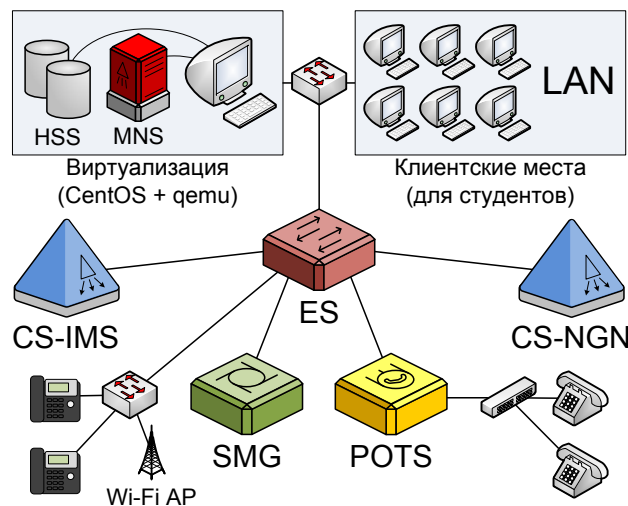


Рис. 1. Функциональная схема лабораторного макета

Серверное оборудование *Iskratel* в данном макете представлено следующими платами:

- Плата *POTS* для подключения аналоговых телефонных аппаратов, к ней подключены два аналоговых телефонных аппарата производства *Avaya*;
- Плата транкингового шлюза *SMG* для взаимодействия с другими коммутационными узлами по протоколам ОКС №7 и *DSS1*;
- Плата *NGN*, являющаяся программным коммутатором сети *NGN*;
- Плата *IMS*, являющаяся call-сервером подсистемы *IMS*;
- Плата *Ethernet*-коммутатора, которая осуществляет коммутацию вышеописанных узлов сети, также к ней подключаются два телефонных аппарата *SIP* производства *Yealink*.

Для управления *call*-серверами используются компьютеры в локальной сети и сервер менеджмента *Lenovo*, на котором установлена система управления *SI3000 MNS*, а также сервер пользовательских данных (*HSS – Home Subscriber Server*) с открытым исходным кодом из проекта *Open-source IMS Core*, разработанного в институте открытых телекоммуникационных систем *Fraunhofer* в Германии.

Для возможности подключения дополнительных абонентов была добавлена *Wi-Fi* точка доступа, благодаря которой можно осуществить подключение к стенду смартфонов, планшетов и иных устройств с целью использования их в качестве *SIP*-клиентов.

На данный момент на стенде нами было реализовано три лабораторные работы:

- «Основы управления и взаимодействия с *SI3000 IMS*»;
- «Добавление абонентов *SIP* и совершение вызовов»;
- «Изучение *Ethernet*-коммутатора *Iskratel ES*».

Первая работа заключается в ознакомлении с системами *MNS* (менеджмента) и сервером пользовательских данных (*HSS*).

Студенты со своих рабочих мест должны произвести вход в эти системы, ознакомиться с основными элементами управления, просмотреть списки абонентов, их свойства и понять различия между концепциями *NGN* и *IMS* с точки зрения конфигурации этих сетей. На рисунке 2 в левом верхнем углу представлено основное окно системы менеджмента, далее представлено приложение *NEM*, с помощью которого можно просматривать и добавлять абонентов подсистемы *NGN*, а также веб-интерфейс *FOKUS HSS*, использующийся для управления списком абонентов и опциями подсистемы *HSS*.

Вторая работа заключается в добавлении *SIP*-абонентов в списки call-серверов *NGN* и *IMS* и совершении проверочного вызова с беспроводного клиента (смартфона). Для этого на своем мобильном телефонном аппарате студенты устанавливают программу *Zoiper* – для *Android/iOS/Windows Phone*, *SIPdroid* – для *Android*, либо подходящие аналоги для совершения вызовов. Ход работы (приемительно к *IMS call-серверу*) представлен на рис. 3.

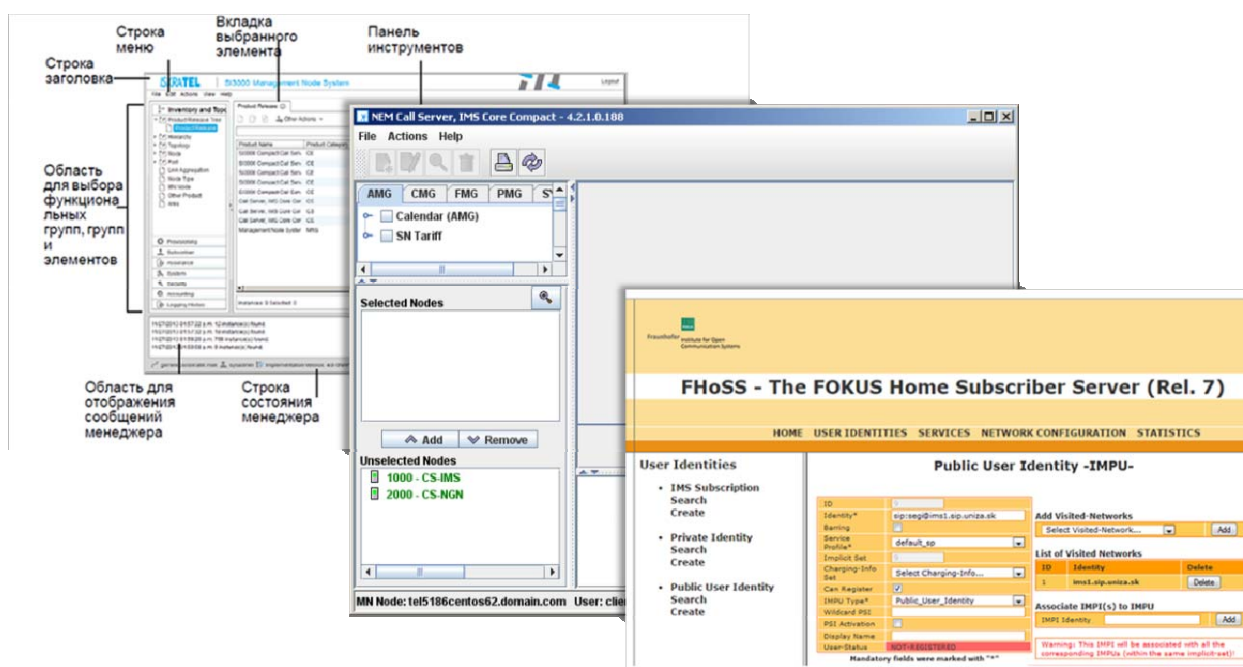


Рис. 2. Изучаемые в лабораторной работе №1 системы лабораторного макета

Студенты должны воспользоваться средствами *MNS* для создания новых абонентов в подсистеме *NGN*. Для этого студенты создают абонента и закрепляют за ним телефонный номер вида **8-495-XXX-XXXX**.

Для создания нового абонента *IMS* студенты должны воспользоваться интерфейсом *FOKUS HSS*, создать три идентификатора (*IMPI*, *IMPU*, *IMSU*) и связать их между собой. В этом случае идентификация абонента будет происходить по SIP-логину вида [user@domain.com](mailto:user@domain.com).

После этого студенты должны произвести вход в созданные ими учетные записи абонентов с помощью мобильного клиента (перечень указан выше) и произвести вызов на один из телефонных аппаратов стенда, чтобы проверить корректность настройки созданного абонента. Если студенты смогли совершить вызов, то данная работа считается выполненной. Таким образом студенты получают практические навыки создания и редактирования списка абонентов, а также узнают различия между механизмами идентификации абонентов в сетях *NGN* и *IMS*.



Рис. 3. Добавление SIP-абонента IMS и пример настройки мобильного SIP-клиента

Третья работа заключается в ознакомлении с *Ethernet*-коммутатором *Iskratel*, расположенном в серверной стойке.

Плата центрального *Ethernet*-коммутатора (*ES*) представляет собой модульную плату. На задней стороне платы размещены интерфейсы *Ethernet*, предназначенные для подключения других плат корпуса (до 20 *GE*-портов). На передней стороне платы расположены порты *Ethernet*, используемые для подключения сетевых устройств на уровне агрегирования (до 4 *GE* «медных» и/или до 4 *GE* оптических портов). Плата *ES* агрегирует трафик, поддерживает различные телекоммуникационные услуги и обеспечивает их обработку.

При выполнении данной лабораторной работы студенты должны получить от коммутатора информацию о конфигурации портов, их состоянии, соединениях с другими платами станда (*POTS*, *SMG*, *CS*, т.е. направление *Downlink* и конфигурацию в направлении транспортной пакетной сети - *Uplink*). В данном макете имеется две виртуальные локальные сети *VLAN* (*Virtual Local Area Network*). Одна из них используется для трафика управления, а вторая для передачи речи по протоколу *IP* и сигнальной информации. Студенты должны определить номера этих виртуальных локальных сетей (*VLAN*). После этого учащиеся составляют схему топологии сети (рис. 4) и вносят туда все полученные данные.

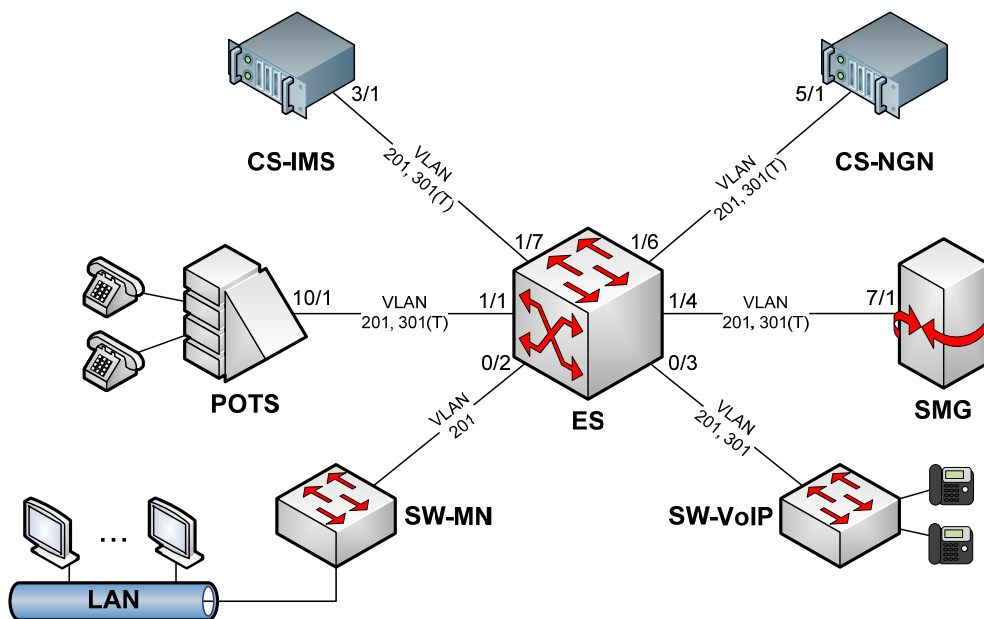


Рис. 4. Пример составления топологии сети

Следующим этапом нашей работы будет постановка лабораторных работ по предоставлению различных дополнительных услуг пользователям на базе платформы *IMS*, например услуг переадресации, параллельного вызова, конференций, видео-конференций и др.

### **Литература**

1. *Деарт В.Ю.* Мультисервисные сети связи. 4.2: Протоколы и системы управления сеансами (Softswitch/IMS) // М.: Брис-М, 2011.
2. Официальный сайт International Telecommunication Union <http://www.itu.int/ITU-T>.
3. Официальный сайт 3rd Generation Partnership Project (3GPP) [www.3gpp.org](http://www.3gpp.org).

# РАЗРАБОТКА ЛАБОРАТОРНЫХ РАБОТ НА БАЗЕ СИМУЛЯТОРА HUAWEI ENSP

*Зеленский П.В., Сазонов А.С.,  
МТУСИ, студенты группы БСС1201,  
Маликова Е.Е.,  
МТУСИ, к.т.н., доцент*

В настоящее время в телекоммуникациях реализуется концепция сетей связи следующего поколения (Next Generation Networks). Происходит переход от традиционных сетей с коммутацией каналов к сетям с коммутацией пакетов на базе IP-технологий.

Сеть NGN должна предоставлять широкий набор услуг с гибкими возможностями по управлению, персонализации и созданию новых наборов услуг за счет унификации сетевых решений. Концепция предполагает использование универсальной транспортной сети с распределенной коммутацией и вынесенными за пределы сети функциями представления услуг, а также интеграцию с традиционными сетями связи. Каждый уровень сети представлен конкретными устройствами.

Функциональная модель NGN представляет из себя несколько уровней (рис 1.):

1. Уровень доступа, на котором происходит концентрация трафика от большого числа разнообразного терминального оборудования к транспортной сети.
2. Транспортный уровень обеспечивает прозрачную передачу данных всех приложений через единую сетевую структуру.
3. Уровень управления сеансами предназначен для обработки сигнальной информации, маршрутизации вызовов и управления сеансами.
4. Уровень управления услугами обеспечивает представление услуг в соответствии с конкретным сценарием их реализации.



*Рис. 1. Функциональная схема сети следующего поколения*

В современных условиях студент должен не только иметь теоретическое представление о работе сетевых устройств на каждом уровне, но и уметь работать с ними, настраивать важнейшие телекоммуникационные устройства – коммутаторы и маршрутизаторы и другие устройства.

**Цель работы:** в условиях отсутствия реального оборудования, научить студентов с помощью эмуляторов работать с актуальным и современным оборудованием IP-сетей.

Для разработки лабораторных работ был выбран эмулятор HUAWEI eNSP. Причины выбора данного эмулятора следующие:

- компания Huawei является крупнейшей телекоммуникационной компанией в мире, которая производит самое современное телекоммуникационное оборудование;



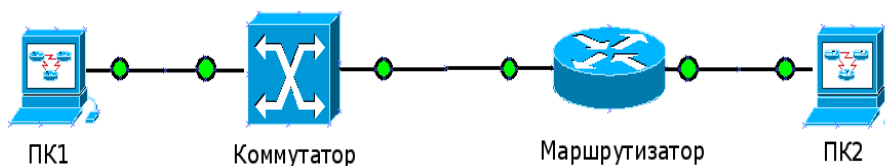
- данный эмулятор является открытым программным обеспечением, что позволяет его устанавливать на компьютеры ВУЗов и на личные компьютеры учащихся, не нарушая лицензионного соглашения;
- он имитирует работу коммутаторов, маршрутизаторов, облачных и других устройств;
- имитирует функции конфигурирования устройств в командной строке, что полностью соответствует работе на реальном оборудовании;
- моделирует сети, содержащие до 200 различных устройств;
- имитирует захват пакетов на интерфейсе с помощью программы Wireshark для наглядного воспроизведения процесса взаимодействия протоколов.

**Результат работы:** разработаны 3 лабораторные работы с методическими и практическими указаниями, охватывающие работу основных сетевых протоколов.

Ниже приведены описания разработанных лабораторных работ.

В первой лабораторной работе студент знакомится с такими протоколами, как ARP и IP. Протокол IP (Internet Protocol) является одним из важнейших сетевых протоколов в сети Интернет и других компьютерных сетях. Он работает на сетевом уровне стека протоколов TCP/IP и является основой этого стека. Протокол ARP (Address Resolution Protocol) – это протокол канального уровня, использующийся для установления соответствия между физическим и логическим адресом. Протокол ARP может применяться для разных типов адресов, но, в основном, используется для перевода IP-адресов в MAC-адреса. В данной лабораторной работе студенту предлагается настроить IP адреса и проверить сетевую связность.

### Лабораторная работа № 1 “Построение простейшей компьютерной сети”

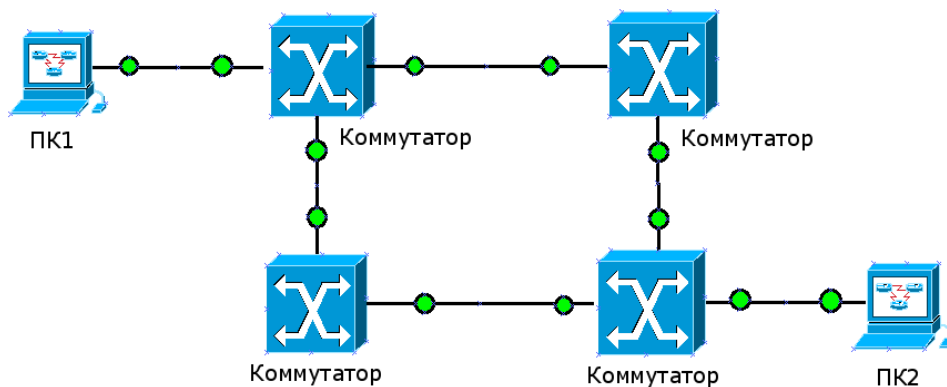


*Рис. 2. Топология сети для лабораторной работы № 1*

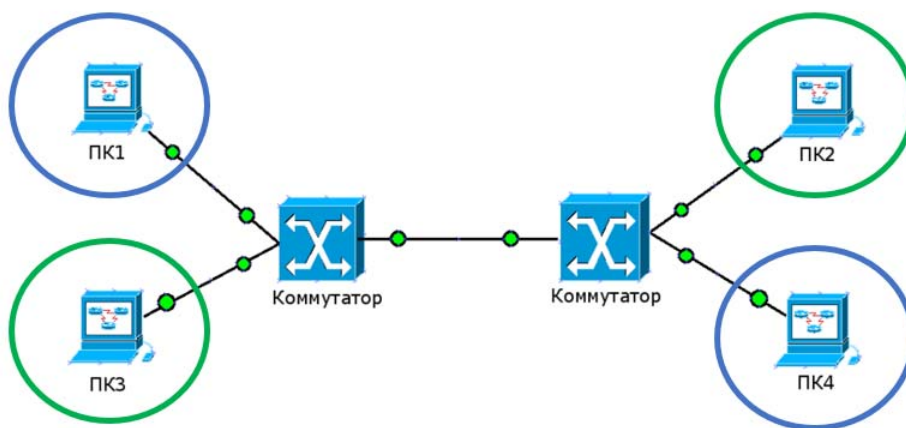
В первой лабораторной работе студент знакомится с такими протоколами, как ARP и IP. Протокол IP (Internet Protocol) является одним из важнейших сетевых протоколов в сети Интернет и других компьютерных сетях. Он работает на сетевом уровне стека протоколов TCP/IP и является основой этого стека. Протокол ARP (Address Resolution Protocol) – это протокол канального уровня, использующийся для установления соответствия между физическим и логическим адресом. Протокол ARP может применяться для разных типов адресов, но, в основном, используется для перевода IP-адресов в MAC-адреса. В данной лабораторной работе студенту предлагается настроить IP адреса и проверить сетевую связность.

### Лабораторная работа № 2 “Базовая настройка коммутатора”

Вторая лабораторная работа делится на две части. В первой части (рис. 3.) студент изучает такие протоколы канального уровня, как STP, RSTP. Протокол STP (Spanning Tree Protocol) является протоколом покрывающего дерева, который предназначен для автоматического удаления петель коммутации на канальном уровне в Ethernet-сетях. Протокол RSTP (Rapid Spanning Tree Protocol) является версия протокола STP с ускоренной реконфигурацией дерева. Студент проверяет работу этих протоколов, например, наблюдает восстановление сети после обрыва. Во второй части (рис. 4.) лабораторной работы студент изучает технологию VLAN (Virtual Local Area Network). Эта технология позволяет создавать изолированные сегменты сети путем логического конфигурирования устройств, не прибегая к изменению физической структуры. Например, на рис. 4 сети VLAN настроены так, что устройства “ПК1” и “ПК4” находятся в одной локальной сети, хотя физически находятся в разных сетях и удалены друг от друга.



*Рис. 3. Топология сети лабораторной работы № 2 (часть 1)*



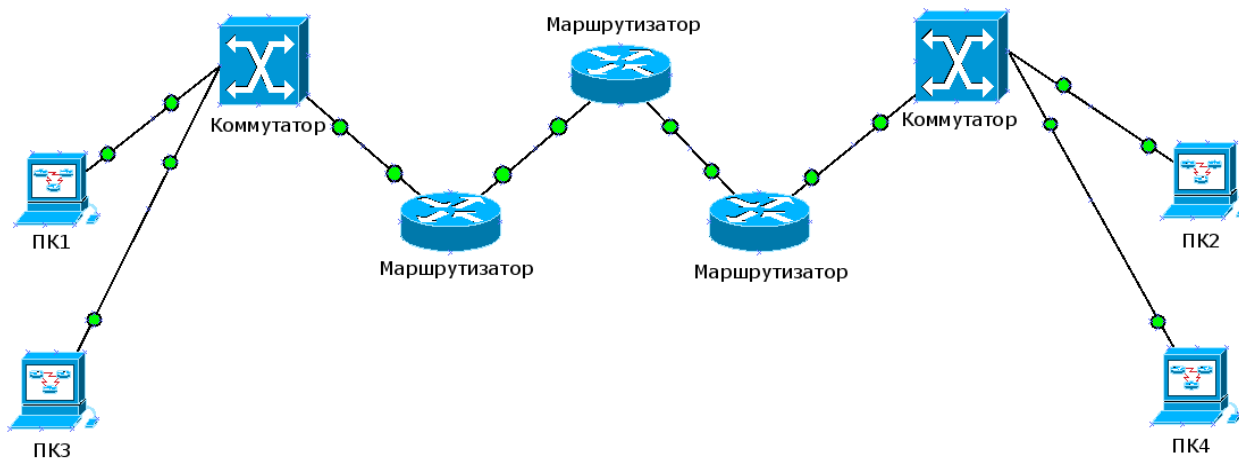
*Рис. 4. Топология сети лабораторной работы № 2 (часть 2)*

### Лабораторная работа № 3 “Настройка протоколов DHCP и OSPF”

Третья лабораторная работа посвящена изучению принципов работы таких протоколов как OSPF и DHCP. Протокол OSPF (Open Shortest Path First) – это протокол динамической маршрутизации, основанный на технологии отслеживания состояния канала и использующий для нахождения кратчайшего пути алгоритм Дейкстры.

Протокол DHCP (Dynamic Host Configuration Protocol) является протоколом динамической настройки узла, который позволяет компьютерам автоматически получать IP-адреса и другие параметры, необходимые для работы в сети.

Студенту предлагается настроить топологию сети (рис. 5). При этом оконечные устройства сети (ПК1, ПК2, ПК3, ПК4) подключаются к коммутаторам и с помощью протокола DHCP получают IP-адреса. Далее используется протокол динамической маршрутизации OSPF, с помощью которого компьютеры обмениваются между собой сообщениями. Протокол OSPF распространяет информацию о доступных маршрутах в данной сети.



*Рис. 5. Топология сети лабораторной работы № 3*

### Литература

1. Олифер В. Г., Олифер Н. А., Компьютерные сети. Принципы, технологии, протоколы. 4-е изд. СПб.: Питер, 2010. 944 с.
2. Деарт В. Ю., Мультисервисные сети связи. Транспортные сети и сети доступа. М.: Брис-М, 2014. 189 с.

# ПОСТАНОВКА ЛАБОРАТОРНЫХ РАБОТ ПО ИССЛЕДОВАНИЮ МОБИЛЬНЫХ ТЕЛЕФОНОВ НА ЛАБОРАТОРНОМ СТЕНДЕ

Каменков М.А.,  
МТУСИ, студент группы БРВ1201  
Синюхин В.С.,  
МТУСИ, студент группы БРМ1202  
Часовников Е.Д.,  
МТУСИ, к.т.н., доцент

Мобильная связь уверенно вошла в нашу повседневную жизнь. Без мобильной связи трудно представить общение с друзьями, коллегами по работе.

При передаче голосовых сообщений и *SMS* мы часто подвергаемся действию мошенников. Для решения проблем с кибермошенниками в области мобильной связи необходима консолидация усилий по нескольким направлениям. Одним из первых этапов в этом направлении является комплекс лабораторных работ по изучению студентами как программного обеспечения, так и аппаратной части, что позволяет комплекс лабораторных работ по изучению среды *Flowcode*.

Лабораторная установка включает в себя компьютер с установленным программным обеспечением и стенд, ядром которого является микроконтроллер.

Программирование микроконтроллеров состоит из двух основных этапов:

- создание программного кода на компьютере;
- запись программы в микроконтроллер.

Среда *Flowcode* позволяет просто создавать приложения для микроконтроллеров с помощью элементов блок-схем. Программное обеспечение позволяет пользователям, не знакомым с программированием, просто и быстро создавать приложения. Созданный проект может быть запущен в режиме моделирования на компьютере или отлажен и записан в чип.

Среда *Flowcode* работает с множеством внешних для микроконтроллера элементов: ЖК-экраны, светодиоды, переключатели, модули средств связи и др.

Написанная программа может быть отлажена в режиме симуляции прямо без записи в чип. После успешного выполнения симуляции программа может быть скомпилирована в микроконтроллер.

Студентам предлагается выполнить в этой среде 4 лабораторные работы, связанных с изучением работы мобильных телефонов: «Изучение среды *Flowcode*» [2], «Изучение макросов *Flowcode* для работы с *GSM*-модулем»[1], «Исследование работы *GSM*-модема в голосовом режиме»; «Исследование работы *GSM*-модема в режиме передачи коротких сообщений». Вид интерфейса среды «*Flowcode*» представлен на рис. 1

В верхней части окна представлена панель элементов. С ее помощью к проекту подключаются эмуляции различных периферийных устройств. Например, в нижней правой части экрана, на панели, можно наблюдать эмуляции клавиатуры, дисплея и восьми светодиодов.

В выполняемых работах делается большой упор на использование макросов – шаблонных процедур с некоторыми входными и выходными параметрами, позволяющих быстро составить требуемый алгоритм работы чипа даже человеку без серьезного опыта программирования микроконтроллеров. Знакомству с макросами посвящена первая лабораторная работа - «Изучение среды *Flowcode*», в которой студент знакомится с инструментами, предоставляемыми средой, учится использовать встроенные макросы, а также создавать собственные.

В лабораторной работе «Изучение макросов *Flowcode* для работы с *GSM*-модулем» студент знакомится с инструментами управления *GSM*-модема, такими как макрос *Initialize* для начальной инициализации *GSM*-модуля, *Check Network Status* – для проверки подключения модуля к сети *GSM*, *Send Command* – для отправки *AT*-команды, заданной в параметре *Command* в виде строки, в *GSM*-модуль, и *AT*-командами *GSM*-модема, позволяющими получить полную информацию о производи-

теле (*AT+CGMI*), модели (*AT+CGMM*), *IMSI*-номере *SIM*-карты (*AT+CIMI*), а также значении уровня принимаемого сигнала (*RSSI*) и коэффициент ошибок (*BER*) (*AT+CSQ*) монтированного в стенд модема. Для получения результатов работы *AT*-команд необходимо использовать макрос *ReadString*, возвращающий последнее информационное сообщение или подтверждение о выполнении команды от *GSM*-модуля.

В лабораторной работе «Исследование работы *GSM*-модема в голосовом режиме» студент должен написать алгоритм и с помощью лабораторного стенда совершить голосовой вызов на мобильный телефон, а потом и принять вызов с телефона. Для этого используются макросы *Check for Incoming* (макрос для проверки незатребованного входящего сообщения от *GSM*-модуля, например, входящего вызова или *SMS*-сообщения.), *Dial Number* (макрос позволяет осуществить исходящий вызов на номер, указанный в параметре *Number*), *Answer Incoming Call* (макрос позволяет ответить на входящий голосовой вызов) и *Hang Up Call* (макрос разрывает текущий разговор) [3].

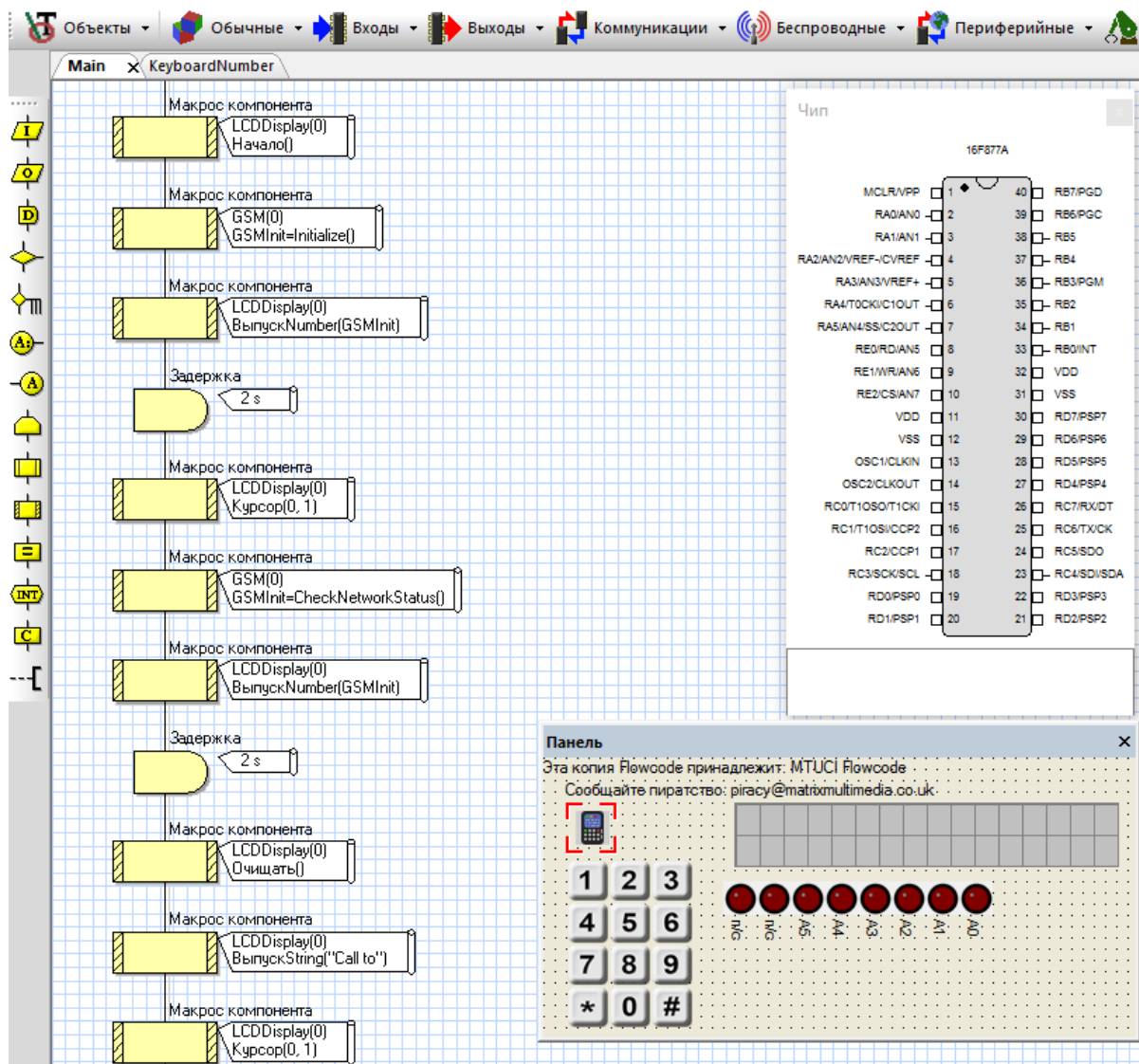


Рис. 1. Интерфейс среды «Flowcode» с фрагментом программы

В лабораторной работе «Исследование работы *GSM*-модема в режиме передачи коротких сообщений» студент должен написать алгоритм и с помощью лабораторного стенда отправить *SMS*-сообщение на мобильный телефон, а потом и принять *SMS*-сообщение с телефона. Для этого используются макросы *Check for Incoming*, *Send Text Message* (макрос отправляет *SMS*-сообщение *Message* на номер *Number*), *Get Text Number* (макрос считывает и возвращает номер телефона, с которого поступило входящее *SMS*-сообщение, обнаруженное ранее с помощью макроса *Check for Incoming*) и

*Get Text String* (макрос считывает и возвращает содержимое входящего SMS-сообщения, обнаруженного ранее с помощью макроса *Check for Incoming*).

В среде *Flowcode* предусмотрена возможность компиляции программы для записи в чип, либо в *C*-файл (программа на языке *C*) или *HEX*-файл (программа на языке ассемблера в шестнадцатеричном формате).

## МАКРОСЫ ОСНОВНЫХ КОМПОНЕНТОВ СРЕДЫ FLOWCODE

### Макросы компонента *LCD Display* (ЖК-дисплей).

*Start* – макрос инициализации дисплея. Этот макрос должен быть использован в начале программы. Без него вывод текста на ЖК-дисплей будет невозможен.

*Clear* – макрос очищает дисплей. При этом удаляются все ранее выведенные на дисплей символы.

*Command* – макрос отправляет значение управляющего сигнала непосредственно в ЖК-дисплей. Управляющий сигнал может быть использован для настройки дисплея и отправки команд непосредственно на дисплей. В лабораторных работах данные сигналы не используются.

*Cursor* – макрос перемещает курсор в позицию, заданную координатами  $x, y$ . После перемещения курсора все выводимые на экран символы будут начинаться в позиции  $x, y$ . Макрос не выводит символ на экран в заданной позиции, а только перемещает курсор в необходимое место. Для вывода символов на экран необходимо использовать другие макросы. Например, чтобы переместить курсор в начало второй строки, необходимо указать значения  $x=0$  и  $y=1$ . Позиция 0, 0 соответствует левому верхнему углу дисплея.

*Print Number* – макрос выводит на дисплей целое число *Number* (*INT*). Можно выводить не только константы, но и значения числовых переменных. Вывод осуществляется на текущую позицию курсора. По окончании вывода курсор остается на следующей позиции.

*Print ASCII* – макрос выводит на дисплей *ASCII*-символ *Character* (*BYTE*). Можно выводить не только символы-константы, но и значения символьных *ASCII*-переменных. Поддерживается вывод сразу нескольких символов. Вывод осуществляется на текущую позицию курсора. По окончании вывода курсор остается на следующей позиции.

*Print String* – макрос выводит на дисплей строку текста *String* (*STRING*). Можно выводить не только строки-константы, но и значения строковых переменных. Вывод осуществляется до тех пор, пока в строке не встретится символ разрыва. Например, если строковая переменная из 20 символов *Str1[20] = "Hello"*, то на экран будет выведено не 20 символов, а 5. Вывод осуществляется на текущую позицию курсора. По окончании вывода курсор остается на следующей позиции.

*Scroll Display* – макрос прокручивает дисплей влево или вправо на число позиций, указанных в переменной *Num\_Positions* (*BYTE*).

*Clear Line* – макрос очищает всю строку с номером *Line\_Number* (*BYTE*) и возвращает курсор в ее начало.

*RAM Write* – макрос позволяет сохранить во внутренней памяти дисплея до 8 пользовательских символов.

### Макросы компонента *LED Array* (блок светодиодов).

*LEDOn* – макрос включения светодиода. Включает светодиод, указанный в переменной *WhichLED*.

*LEDOff* – макрос выключения светодиода. Выключает светодиод, указанный в переменной *WhichLED*.

### Макросы компонента *Switch Bank* (блок переключателей).

*Read State* – макрос считывает состояние одного переключателя, указанного переменной *WhichSwitch*.

*Wait Until High* – макрос приостанавливает выполнение программы до тех пор, пока не нажата кнопка переключателя, указанного в переменной *WhichSwitch*.

*Wait Until Low* – макрос приостанавливает выполнение программы до тех пор, пока не отпущена кнопка переключателя, указанного в переменной *Which Switch*.

Лабораторная установка подготовлена к работе. По всем четырём работам подготовлены контрольные отчёты.

## Литература

1. *Попов В.И.* Основы сотовой связи стандарта *GSM*. М.: Эко-Трендз, 2005. 296 с.
2. <http://flowcode.info>. – Русскоязычный ресурс, посвященный данной среде разработки.
3. *Кааранен Х.* и др. Сети *UMTS*. Архитектура, мобильность, сервисы. М.: Тухносфера, 2007.

# ПРОЕКТИРОВАНИЕ МАСШТАБИРУЕМОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ С ВОЗМОЖНОСТЬЮ ИНТЕГРАЦИИ ОБРАБОТКИ ДАННЫХ В МОБИЛЬНОЕ ПРИЛОЖЕНИЕ

Стрельников В.Г., Шалаумов Э.В.,  
МТУСИ, студенты группы БВТ1301  
Яшина М.В.,  
МТУСИ, д.т.н., профессор

В настоящее время огромная роль в организации большинства процессов по обмену информацией, к примеру, на предприятиях, в сфере образования и т.д., возложена на информационные системы (ИС).

Для каждой сферы жизнедеятельности, в которых необходим постоянный обмен данными, строятся свои уникальные информационные системы. Однако процесс проектирования ИС возможно значительно сократить, создав типизированную масштабируемую модель, центральная структура которой будет обеспечивать возможность внедрения дополнительных ресурсов без изменения уже целостной структуры данных.

Итак, что же такое информационная система? Это система, предназначенная для хранения, поиска и обработки информации и соответствующие организационные ресурсы, которые обеспечивают и распространяют информацию.

Передо мной стояла задача разработки такой ИС, которая удовлетворяла бы ряду следующих критериев:

- Наличие клиентского приложения – конечный пользователь должен иметь доступ к ИС без установки какого-либо дополнительного ПО, а также хранить необходимые данные на сервере;
- Внешнее хранилище данных – для устранения дополнительной нагрузки на сервер загружаемые пользователем данные хранить в облачных сервисах;
- Автоматизированность – обеспечить работу пользователей с ИС в автоматизированном режиме, то есть на базе центральной структуры сформировать процесс обработки данных практически без вмешательства администратора системы. Почему “практически без”? Это связано с тем, что часть информации может устаревать значительно быстрее, чем другая — как минимум решение проблемы актуальности выводимых данных;
- Масштабируемость – организация центральной структуры системы таким образом, чтобы наращивание дополнительных ресурсов не приводило к её пересмотру/реструктуризации;
- *API* – реализация одних из основных составляющих ИС, а именно: обработку и поиск информации, с возможностью дальнейшей интеграции в мобильное приложение.

Таким образом, я пришел к выводу, что для реализации поставленной задачи отличным решением будет: многослойная клиент-серверная распределенная ИС.

Она представляет собой сайт, который и будет являться клиентским приложением. БД и СУБД находятся на сервере. Также на сервере размещено и файловое хранилище – *FTP*. *FTP* обеспечивает доступ к клиентскому приложению посредством просмотра веб-страниц, на которых и будут происходить процессы поиска и обработки данных.

Правильней сказать, что пользователь отправляет *HTTP*-запрос через браузер на сервер, который принимает его, обрабатывает и возвращает содержательный ответ – веб-страницу.

Запрашиваемая страница может содержать *PHP*-сценарии, которые сервер будет обрабатывать *PHP*-интерпретатором. Также фрагменты *PHP* кода могут содержать *MySQL*-инструкции, которые интерпретатор, в свою очередь, передает процессору (или серверу) базы данных *MySQL*.



Для реализации поставленной задачи мною были выбраны следующие инструменты:

- Основа сайта будет выполнена на скриптовом языке веб-программирования *PHP v.5.4.* *PHP*-сценарии будут обеспечивать обработку запрашиваемых данных, а также выполнять обращение к БД сервера для загрузки необходимой информации.
- В некоторых сценариях будет разумно использовать фоновый обмен данными между браузером и веб-сервером. В этом поможет асинхронный *JavaScript* и *XML – (AJAX)*
- Хранение и обработка части данных будет осуществляться СУБД – *СУБД MySQL v.5.1.*
- Для *MySQL* существует графический веб-интерфейс – *phpMyAdmin v.4.5.2.*
- Для хранения некоторых файлов будет использовано облачное хранилище — Яндекс.Диск, который обеспечивает разработчиков собственным *API*.
- Графический интерфейс сайта будет реализован с помощью фреймворка – *Bootstrap v. 3.3.6.*

Так совпало, что мне довелось начать свою работу не с чем-то общим, а на конкретном примере: необходимо было разработать ИС для кафедры МКИИТ.

Итак, свою исследовательскую деятельность я начал с изучения предметной области (кафедры). Необходимо понять, какие группы пользователей будут включены в ИС, с какими правами доступа, а также возможностями взаимодействия с ней. Из этого и будет вытекать базовый набор функций ИС.

ИС проектируется как масштабируемая, что достигается за счет составления такой структуры определенных данных (информации о дисциплинах, преподавателях, группах, студентах, расписании и т.д.), которые обеспечивают некое “ядро” для дальнейшего развития. То есть любые дополнения не будут приводить к необходимости изменения изначальной структуры данных. Дополнительные модули лишь используют связи с уже имеющимся набором информации.

На подготовительном этапе было выявлено 4 группы пользователей, между которыми будут происходить взаимодействия:

1. Администратор или редактор сайта;
2. Преподаватели. Включают: преподавателей, сотрудников учебно-вспомогательного персонала, ассистентов;
3. Студенты. Включают: аспирантов, магистров, бакалавров;
4. Сторонние посетители.

Каждая группа пользователей выполняет определенные роли в реальной жизни — образовательный процесс. Именно эти роли и ложатся в основу определенных прав для доступа к ИС, из которых вытекают конкретные наборы возможностей по взаимодействию с ней. Помимо этого подхода к определению инструментов необходимых в ИС также ставились конкретные задания руководителем кафедры — Мариной Викторовной — для обеспечения упрощения образовательного процесса.

Итак, реализованы следующие разделы:

- Новости кафедры, расписание, информация о дисциплинах, преподавателях и студентах, контактная информация

Доступ к данным разделам имеют все группы пользователей. То есть информация является открытой для всех. Изменения данной информации доступны только администратору сайта.

- Личный кабинет

Доступ к личному кабинету есть только у зарегистрированных пользователей, каждый из которых имеет свой уникальный идентификатор, который выдается на кафедре. Без данного кода попасть в закрытую часть сайта невозможно.

В личном кабинете пользователь может: изменять информацию о себе; добавлять собственные публикации, статьи; просматривать личные сообщения.

- Группы

Для упрощения информирования студентов будет введен раздел группы. Преподаватели по выбранной группе могут отсылать сообщения, то есть какие-либо новости, либо файлы, которые видны всей группе. Можно делать пометки о важности сообщения.

У студентов данная “лента новостей” едина для всех преподавателей.

- Электронная библиотека

Доступ у зарегистрированных пользователей. У каждого участника ИС есть возможность загрузить книгу, журнал, методические указания и др., просмотреть уже загруженный им материал и при необходимости изменить введенные данные, а также осуществить поиск по уже сформированной базе данных, то есть непосредственно по электронной библиотеке.

- Электронный журнал

Доступ у преподавателей. Электронный журнал реализован для учета успеваемости, посещаемости студентов. Для него реализовано первое *API*.

- Работы

Студент имеет возможность загрузить дипломную, курсовую работу или реферат на сервер для проверки преподавателем.

Каждая работа загружается для конкретного преподавателя, который курирует студента. После загрузки работы преподавателю приходит оповещение в личный кабинет. Работу можно оценить: либо поставить оценку, отметку о проверке, либо “вернуть” на доработку с комментариями.

До отправки преподавателю программ, написанных на языке *C++*, студенты могут скомпилировать их прямо на сайте. После компиляции будет выведено сообщение либо об успехе, либо об ошибке с указанием на нее.

Изначальная задача заключалась в возможности интеграции обработки, поиска информации в мобильное приложение. Данный подход реализуется с помощью *API* сайта.

Для тестовой версии был использован инструмент ИС – электронный журнал.

Мобильное приложение, которое использует *API* сайта, разработано на языке *Java* под ОС *Android*. Принцип работы заключается в отправке *POST*-запросов на сервер с необходимым запросом конкретных данных. Сервер в свою очередь начинает обработку полученной информации. При необходимости делает запрос к БД. После обработки возвращает либо текстовый ответ, либо объекты в формате *JSON* — универсальная структура данных, которая реализует текстовый формат обмена данными.

Для разработанного приложения были проведены тесты с целью определения количества потребляемого трафика интернета

Для определения эффективности работы приложения был проведен тест, который заключался в анализе объема передаваемых данных по сети Интернет как посредством приложения, так и мобильной версией сайта. Для анализа был проделан ряд идентичных последовательных итераций, по результатам которых были вычислены объемы трафика. Для мобильного приложения трафик составил – 33 Кбайта, а для мобильной версии сайта – 638 Кбайта. Таким образом, становится очевидным, что объем передаваемых данных через мобильное приложение значительно ниже (в 19 раз), чем через мобильную версию сайта. Это связано с тем, что через *API* запрашиваются конкретные данные, а в мобильной версии сайта необходима загрузка полной веб-страницы, которая включает в себя дополнительный код, стили и изображения, что существенно влияет на размер данных.

Таким образом, на данный момент было сформировано центральное “ядро” ИС, которое обеспечивает соответствие поставленной задаче:

- Наличие клиентского приложения — создан сайт;
- Внешнее хранилище данных — данные сайта хранятся частично на сервере, частично в облачном сервисе;
- Автоматизированность – организуется за счет неизменности некоторой части информации, а именно: дисциплины, преподаватели, студенты, расписание. Безусловно, есть исключения, к примеру, студент может перевестись на другой факультет, но такие изменения единичны. В целом что-то глобально меняется раз в семестр;
- Масштабируемость – обеспечивается за счет определенного структурирования данных;
- *API* – в тестовом режиме реализовано для электронного журнала.

## Литература

1. Робин Никсон. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5, O'Reilly, 2011. 497 с.
2. Бретт Маклафлин. PHP и MySQL. Исчерпывающее руководство, O'Reilly, 2016. 544 с.
3. Зигард Медникс, Лайрд Дорнин. Программирование под Android, O'Reilly, 2012. 496 с.
4. Буслаев А.П., Проворов А.В., Яшина М.В. Современные подходы к исследованию поведения связанного потока частиц с мотивацией // Т-Comm: Телекоммуникации и транспорт. 2011. Т. 5. № 2. С. 61-62.
5. Бугаев А.С., Буслаев А.П., Козлов В.В., Яшина М.В. Некоторые математические и информационные аспекты моделирования трафика // Т-Comm: Телекоммуникации и транспорт. 2011. Т. 5. № 4. С. 29-31.

# НЕКОТОРЫЕ ОСОБЕННОСТИ ИЗМЕРЕНИЯ, ПОЛУЧЕНИЯ И ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ ТРАФИКОВЫХ МАТРИЦ

Денисов И.А., Пименов С.А.,  
МТУСИ  
Шелухин О.И.,  
МТУСИ, д.т.н., профессор

Одним из перспективных направлений исследований последних лет являются разработки в области сетевой томографии. Сетевая томография – это изучение внутренних характеристик сети, используя информацию, полученную из данных конечной точки. Несмотря на то, что этот термин впервые был введен Yehuda Vardi еще в 1996 году [1], исследования в данной области не прекращаются и по сей день: в будущем это позволит разрабатывать более эффективные компьютерные сети и повысить их качество обслуживания (Quality of Service) ограничением потери пакетов в физических соединениях (links), а также оптимизацией маршрутов передачи данных. Одной из важнейших задач в рамках сетевой томографии является получение трафиковых матриц.

В современных сетях связи трафик, проходящий через сеть, имеет источник и пункт назначения. Информацию об уровне трафика в сети позволяет оценить трафиковая матрица (англ. *traffic matrix*). Элементы, или ячейки данной матрицы отражают уровень трафика в каждой паре источник/получатель (англ. *origin-destination pairs*, или *OD pairs*) [2].

Основная сложность при работе с ТМ и их восстановлении заключается в некорректно поставленной проблеме их получения.

Рассмотрим данные нагрузки соединений  $y$ , которые удовлетворяют системе линейных уравнений  $y = Ax$ , где вектор  $y$  размерностью  $n \times 1$  содержит известные данные, а вектор  $x$  размерностью  $r \times 1$  содержит неизвестные для оценки. Матрица  $A$  имеет размерность  $n \times r$ . Во многих интересных случаях  $r > n$ , следовательно, не существует единого решения такой системы. Такие проблемы называются некорректно поставленными проблемами.

Пусть  $y$  – это измерения соединений (полученные, например, при помощи протокола SNMP);  $x$  – вектор-столбец ненулевых значений ТМ (подобная форма применяется для вычислительных целей);  $A$  – матрица маршрутизации [3].

На практике при восстановлении ТМ пользуются методиками, комбинирующими особенности двух соответствующих основных подходов: оценка ТМ и получение ТМ. Подход «оценки ТМ» принимает в качестве входных данных некую исходную ТМ, синтетически измеренную вручную в течение короткого периода времени. В результате можно получить ТМ, которая может не отражать реальное количество переданного трафика в расчете на каждую ненулевую OD-пару. Однако в таких ТМ всегда будут иметь место определенные метрики течения трафика в сети, которые позволят с достаточной степенью достоверности делать соответствующие выводы, в том числе и о присутствии аномалий трафика в сети.

Подход «получения ТМ» принимает на вход данные реальной сети, которые можно снять с доступного сетевого оборудования (например, маршрутизаторов трафика). Зная это, а также топологию сети и действующие в ней маршруты, можно также получить ТМ, на основании которых можно описать поведение трафика в сети.

В качестве конкретных методик восстановления ТМ из исходных данных были рассмотрены три метода оптимизации: метод Хука-Дживса, один из квазиньютоновских методов, использующего для основной цели метод линейного поиска [4], а также метод Левенберга-Марквардта [5].

Существует множество различных метрик и критериев оценки получаемого результата (в нашем случае результата оптимизации), основными из которых являются: точность найденного решения по сравнению с известным нам истинным и скорость работы, выражаемая количеством произведенных итераций.

Определив методологию исследования как оценку того, как влияет на критерии относительной погрешности и скорости выполнения зависимость выбора исходной точки в некоторой окрестности истинного решения с вариативным радиусом, были получены следующие результаты, представленные в таблице 1. В качестве среды для исследования был выбран математический пакет MatLAB R2014a.

Таблица 1

### Сравнительная характеристика исследуемых методов

Метод	Хука-Дживса	Квазиньютоновский	Левенберга-Марквардта
Максимальная погрешность	> 10%	~ 10%	~ 0.02%
Максимальное количество необходимых итераций	> 600	~ 35	~ 20

Можно видеть, что метод Левенберга-Марквардта демонстрирует лучшие значения критериев относительной погрешности и скорости результаты по сравнению с другими методами.

Вместе с тем подобные «прямые» решения задачи получения ТМ (то есть такие, которые не требуют внесения дополнений и изменений в исходные данные) имеют ряд недостатков, которые не позволяют использовать их непосредственно. Среди них можно выделить необходимость значительной априорной информации о примерном местонахождении истинного решения в известном диапазоне, а также неоправданно высокие затрачиваемые ресурсы и время работы соответствующих реализаций.

Проведенные исследования показали, что данный тип решений может быть использован как инструмент нахождения ТМ, учитывающий как все имеющиеся особенности заданных исходных данных, так и некую дополнительную информацию, которая позволит снизить вычислительную сложность.

### Литература

1. Vardi, Y. Network Tomography: estimating source-destination traffic intensities from link data // Journal of the American Statistical Association (American Statistical Association), 1996
2. I. Juva. Traffic Matrix Estimation // Helsinki University Of Technology, Department of Electrical and Communications Engineering Networking Laboratory, 2005
3. P. Tune, M. Roughan. Internet Traffic Matrices: A Primer // Recent Advances in Networking, 2013
4. Трифонов А.Г. Постановка задачи оптимизации и численные методы ее решения // Web: [http://matlab.exponenta.ru/optimiz/book\\_2/index.php](http://matlab.exponenta.ru/optimiz/book_2/index.php)
5. А.В. Пантеев, Т.А. Летова. Методы оптимизации в примерах и задачах. // М.: Высшая школа 2005.
6. Костин Д.В., Шелухин О.И. Разработка защищенной системы обмена мгновенными сообщениями и медиа-файлами // Телекоммуникации и информационные технологии. 2015. Т. 2. № 2. С. 40-44.
7. Шелухин О.И., Олейникова Т.В. Оценка эффективности стеганографического скрывания цифровых водяных знаков в видеопоследовательностях за счет дифференциальной разности энергий областей изображения // Научные технологии в космических исследованиях Земли. 2016. Т. 8. № 2. С. 70-75.
8. Шелухин О.И., Панкрушин А.П. Оценка достоверности обнаружения аномалий сетевого трафика методами дискретного вейвлет-анализа // Т-Comm: Телекоммуникации и транспорт. 2013. Т. 7. № 10. С. 110-115.
9. Шелухин О.И., Антонян А.А. Анализ изменений фрактальных свойств телекоммуникационного трафика вызванных аномальными вторжениями // Т-Comm: Телекоммуникации и транспорт. 2014. Т. 8. № 6. С. 61-64.
10. Шелухин О.И., Савелов А.В. Имитационное моделирование аномалий трафика в локальной компьютерной сети // Т-Comm: Телекоммуникации и транспорт. 2013. Т. 7. № 10. С. 103-109.

# ОЦЕНКА РАССТОЯНИЯ МЕЖДУ ДВУМЯ УСТРОЙСТВАМИ, СОЕДИНЁННЫМИ ЧЕРЕЗ Wi-Fi

Новиков А.М.,  
МТУСИ, студент группы БВТ1301  
Волков М.М.,  
МТУСИ, ассистент

В настоящее время сети *Wi-Fi* охватывают все больше сфер жизни и позволяют реализовать полный спектр информационных услуг: передачу сообщений, обмен данными, подключение к интернету, получение и передача видеоизображений и т.д. *Wi-Fi* становится популярной не только как сеть домашнего пользования, но также она устанавливается в общественных местах, и даже транспортных средствах.

Достоинствами беспроводных сетей *Wi-Fi* являются: гибкость в построении, высокая скорость передачи информации, быстрота и простота построения локальной сети, дешевизна содержания. Цель работы - определения расстояния до устройства с включенным *Wi-Fi* передатчиком.

Одной из самых популярных категорий среди мобильных приложений является категория, использующая сервисы определения местоположения. Огромное количество людей использует навигационные системы. Но главным недостатком *Global Positioning System (GPS)* является невозможность предоставления данных о местоположении внутри зданий из-за сильных помех, создаваемых стенами и перекрытиями.

На сегодняшний день задачу позиционирования внутри здания можно решить с помощью сетей *Wi-Fi*. Основные реально действующие системы позиционирования в сетях *Wi-Fi* используют *Received Signal Strength Indicator (RSSI)* – показатель принимаемого уровня сигнала. Он измеряется приемником от так называемых «радиомаяков».

Позиционирование в беспроводных сетях можно реализовать несколькими способами:

Метод распознавания шаблона. Метод заключается в том, что устройство сканирует сети на наличие точек доступа и их *RSSI*, сравнивает полученный результат в виде схемы радиосигналов со списком заранее заданных шаблонов и согласно наиболее подходящему шаблону определяет координату устройства. Для создания набора шаблонов необходимо проводить длительный процесс сканирования эфира помещений и повторять этот процесс регулярно для калибровки созданных шаблонов. Точность позиционирования при использовании этого метода будет низкой.

По точке доступа, к которой присоединен клиент. Этот метод является простым, но также не отличается высокой точностью. С его помощью можно определить наличие устройства в определенном диаметре, но не положение его относительно источника в этом диаметре. Потому метод скорее подойдет для определения количества присутствующих клиентов, нежели для их позиционирования.

Триангуляция. Метод Триангуляции заключается в расположении 3-4 точек доступа *Wi-Fi* так, чтобы диаметры распространения их сигналов пересекались, определении мощности сигнала от клиента каждой из этих точек и позиционировании клиента в зоне пересечения сигналов относительно каждой из них. При правильном расположении точек доступа позиционирование данным методом может достигать точности 5-7 метров, что является вполне хорошим результатом по сравнению с методами, описанными ранее. Препятствия между точками доступа могут быть статическими и движущимися. Статические препятствия – это стены и перегородки. Так как они в основном неподвижны, их влияние можно учитывать при позиционировании. Движущиеся препятствия – это люди, их влияние хуже поддается учету, так как можно лишь приблизительно оценивать количество людей в те или иные часы.

При увеличении количества точек доступа увеличится также и точность позиционирования. Это связано с экспоненциальностью зависимости уровня сигнала от расстояния до точки доступа, то есть чем дальше устройство от точки доступа – тем менее точно определяется расстояние до него, соответственно и менее точно определяется конечная координата.

Ангуляция или позиционирование с определением угла входящего сигнала. Данный метод является разработкой компании *Cisco* и основывается он на показаниях *Cisco Aironet* – внешнем модуле точечного позиционирования. На данном модуле установлена антенна, на которой расположены ещё 32 антенны, каждая из которых принимает сигнал по-разному. С помощью анализа мощностей сигнала, принимаемого этими антеннами, и удается выяснить угол, под которым пришел сигнал на модуль.

При совмещении методов триангуляции и ангуляции, т.е. установки нескольких модулей *Cisco Aironet*, диаметры распространения сигналов которых пересекаются, можно добиться точности позиционирования в 1 метр.

Позиционирование устройства в сети *Wi-Fi* можно использовать для нескольких видов приложений:

- А) обнаружение активов, помеченных метками *Wi-Fi*;
- Б) подключение к сети *Wi-Fi* с учетом местонахождения клиента;
- В) навигация по помещению;
- Г) отправка высокоэффективных предложений с учетом местонахождения клиента;
- Д) сбор данных для анализа поведения клиентов.

Частота обновления координаты клиента играет немаловажную роль для приложений В)-Д).

В то время как клиентское устройство не подключено к сети *Wi-Fi*, оно отправляет пробы (*probe request*). Пробы являются широковещательными сигналами, то есть они передаются по всем частотным сигналам и видны всем точкам доступа в доступном радиусе, позволяя позиционировать себя методами триангуляции и ангуляции.

Однако периодичность отправления проб зависит от производителя устройства, и она может быть 15, 30, а то и 60 секунд, что является большим недостатком для позиционирования, ведь за 60 секунд человек может уйти в другую часть помещения, а его движения не будут отслеживаться в течение этих 60-ти секунд. Еще хуже на позиционировании по пробам сказывается тот факт, что производители для экономии энергии устройств стремятся снизить частоту отправки проб.

Чтобы увеличить частоту обновления координат клиента, разработчики *Cisco* использовали трафик данных для позиционирования устройства. Это позволило собирать данные с частотой до 10-ти раз в минуту – *FastLocate*.

*FastLocate* может быть реализован на отдельном модуле для модульных точек доступа *Cisco Aironet* серий 3600 и 3700. Модуль *WSM* будет сканировать эфир и собирать информацию для расчета координат устройств примерно каждые 8 секунд.

Есть и другой вариант использования *FastLocate*, подходящий для любых точек доступа *Cisco Aironet*, и не требующий дополнительного модуля. На короткий промежуток времени точка доступа выходит из режима обслуживания клиентов и переключается в режим сканирования эфира (*Enhanced Local Mode* — *ELM*). В *ELM* точка, как и в предыдущем случае, собирает информацию для расчета координат устройств и переключается обратно в режим обслуживания клиентов. Такой метод имеет недостаток: снижается производительность сети *Wi-Fi* примерно на 15%, так как точки доступа не могут обслуживать клиентов 100% времени.

## Литература

1. Д.Е. Намиот, Использование оценки приближенности к узлам *Wi-Fi*-сети для доставки гиперлокального контента, 2012
2. Салех Х.М., Фролов И., Матафонов Д., Мобильная система позиционирования внутри помещений посредством *wi-fi* сетей, -С. 99-101.
3. Rainer Mautz, Indoor Positioning Technologies, Habilitation Thesis submitted to ETH Zurich, 2012

# РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ КОНТЕНТ-АНАЛИЗА ИНТЕРНЕТ-ПУБЛИКАЦИЙ

Толмачев Р.В.,  
МТУСИ, студент группы БВТ1201  
Воронова Л.И.,  
МТУСИ, д. ф-м.н, профессор

Исследование блогов, как источников интеллектуальной и социологической информации, имеет важное значение при *анализе различных социальных явлений*. Блогосфера это важный источник, который позволяет узнать реакцию людей на ту или иную проблему. В этой связи целесообразно разработать интернет-приложение, отражающее реакцию людей на то или иное событие, а также работающую, как поисковая система, позволяя найти в блогах записи и упоминания о конкретном событии или человеке.

В 1989 году, на первом семинаре посвященном поиску знаний в данных, Григорием Пятецким-Шапиро был введен термин Data Mining.[1]

Data mining – это технология включающая в себя методы обработки информации для получения полезных знаний из различных источников. С помощью методов Data Mining можно обработать текстовую информацию, изображения, видео звук.

Web Mining – это анализ интернет-ресурсов методами Data Mining. В настоящее время Интернет является одним из самых больших источников информации, поэтому важно уметь получать знания из Интернет-источников. Одним из способов анализа информации в технологии Web Mining– это контент анализ.

Контент-анализ (от англ.: англ. contents — содержание, содержимое) или анализ содержания — стандартная методика исследования, предметом анализа которой является содержание текстовых массивов[2]. В большинстве случаев объектами исследования являются тексты.

Контент анализ может быть *количественным*, а может *качественным*.

*Количественный контент-анализ* основывается на изучении содержания сообщения. Для анализа необходимых элементов, нужно учитывать их смысл и определять любой возможный результат анализа в соответствии с ожиданиями исследователя.

В качестве источника интернет-публикаций мы будем использовать Живой Журнал ([www.livejournal.com](http://www.livejournal.com)). Живой Журнал — блог-платформа для ведения онлайн-дневников (блогов), а также отдельный персональный блог, размещённый на этой платформе.[2]

На главной странице мобильной версии сайта находится "топ" публикаций - это наиболее обсуждаемые и просматриваемые публикации *за последние 4 часа*. Эти публикации будет использовать приложение в качестве источника публикаций.

На рисунке 1 изображена архитектура интернет-ресурса «Мнение».

Интернет-ресурс мнение состоит из четырех взаимодействующих модулей.

Первый модуль – это загрузчик. Он загружает HTML-код страницы с публикацией и передает ее следующему модулю.

Второй модуль – это обработчик. Модуль обрабатывает HTML-код страницы и выделяет основную информацию о публикации.

Модуль обработчик принимает значение переменной от модуля-загрузчика, производит обработку HTML-кода и получает данные необходимые для заполнения сведений об авторе в базе данных и текст публикации. Текст публикации подвергается обработке модулем-обработчиком методами контент-анализа:

1. составление частотного словаря

При добавлении слова в словарь учитывается только основа слова (слова без суффиксов и окончаний). Стеммер Портера — алгоритм, который применяя последовательно ряд правил, отсекает окончания и суффиксы, основываясь на особенностях языка, в связи с чем работает быстро.

При составление словаря из текста исключаются стоп-слова. Стоп-слова - это слова-связки, без которых не возможно построение полноценного смыслового текста.

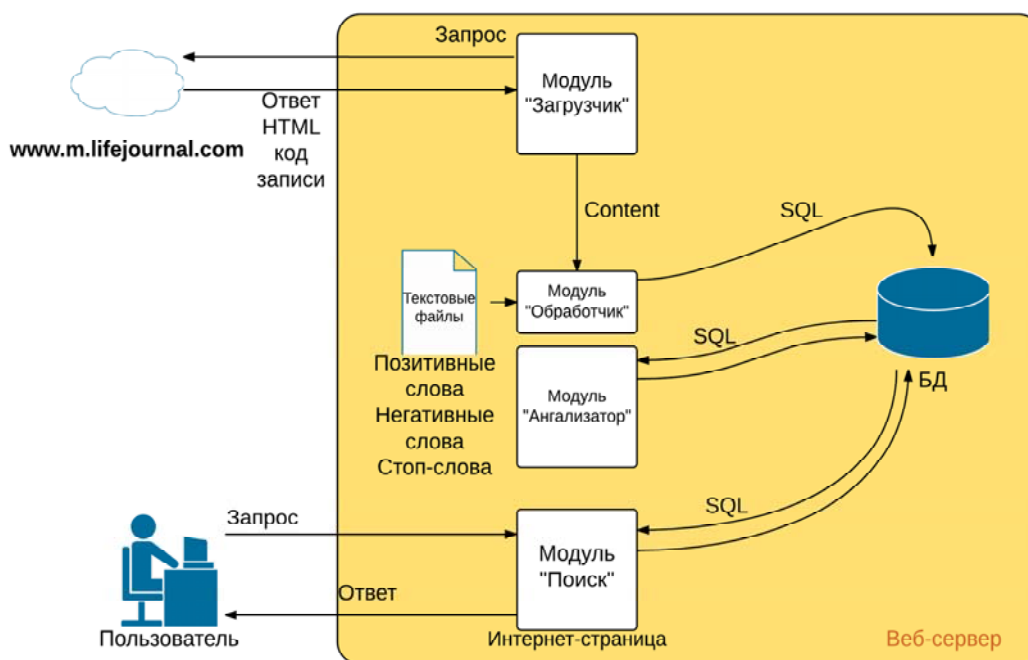


Рис. 1. Архитектура интернет-ресурса «Мнение»

## 2. реферирование

Реферирование — это вариант структурирования информации, упрощающий восприятие информации посредством ее сжатия и структурирования.

В приложении реферирование проводится на основе словаря термов, при помощи метода Луна. Метод Луна (Luhn). Он предположил, что часто встречающиеся в тексте слова являются наиболее значимыми и чем больше часто встречающихся слов оказывается рядом, тем более существенную информацию содержит предложение.

## 3. сентиментный анализ

В памяти ресурса содержатся два файла, в которых хранятся списки слов имеющих позитивную и негативную окраску. Высчитывается частота вхождения слов из исходного текста в файле с положительными и отрицательными словами. На основе частоты определяется настроение статьи: положительная (частота положительных слов больше), отрицательная (частота отрицательных слов больше), нейтральная (частота одинакова).

Третий модуль – это анализатор. Объединяет публикации со схожей тематикой путем латентно семантического анализа.

Латентно семантический анализ отображает документы и отдельные слова в так называемое «семантическое пространство», в котором и производятся все дальнейшие сравнения.

Модулем производится сингулярное разложение матрицы для получения координат статей в семантическом пространстве. Координаты публикаций записываются в базу данных.

После этого идет время ожидания загрузки новых статей модулем-загрузчиком.

Четвертый модуль – это поисковик. Позволяет пользователю взаимодействовать с базой данных, которая заполняется во время работы предыдущих модулей.

Модуль-поиск работает независимо от других модулей. Пользователь на своем компьютере заходит на сайт интернет-ресурса и видит данные записанные в базу данных, в виде графиков и таблиц. Также пользователь имеет возможность найти интересующие его публикации, используя ключевые слова в строке поиска.

База данных содержит таблицы в которой хранятся данные об авторе блога и о записи в блоге.

Модули взаимодействуют с базой данных, которая также находится на WEB-сервере, с помощью SQL запросов.

Для хранения данных, полученных после обработки была создана база данных «Content», состоящая из 5 таблиц:

1. Публикация: содержит информацию о публикациях;



2. Словарь: содержит словарь всех слов
3. Автор: содержит информацию об авторах
4. Слово в публикации: содержит информацию о содержании слова в публикации
5. Схожесть: содержит координаты публикаций в семантическом пространстве.

Для реализации интернет-ресурса была написана программа на языке C#. Пользовательский интерфейс разработан в виде интернет-страницы. База данных реализована с помощью MySQL.

Модуль-загрузчик посылает запрос к HTML-странице публикации и возвращает HTML-код страницы, для обработки его следующим модулем. Модуль загружает статьи автоматически, через определенный промежуток времени.

Модуль обработчик получает на вход HTML-код страницы и производит его обработку. Получая тем самым Необходимые данные для заполнения базы данных «CONTENT». Для обработки HTML-кода страницы используются регулярные выражения. Из регулярных выражений была создана маска.

Модуль анализатор реализован на языке C#. Модуль состоит из одной процедуры. Данная процедура формирует матрицу, строки которой – слова из словаря а столбцы – публикации. Матрица заполняется числовыми значениями которые равны количеству вхождений каждого слова в каждую публикацию.

С помощью функции SVD(), дополнительной библиотеки для C# MathNet.Numerics.LinearAlgebra.Double, производится сингулярное разложение матрицы, результатом которой также является матрица, которая содержит координаты каждой статьи в семантическом пространстве. Первые 2 строки этой матрицы это координата X и Y соответственно.

Модуль-поиск реализован на языке PHP. Он необходим для того чтобы дать возможность пользователю искать необходимую ему информацию.

Пользователь формирует запрос в строке поиска, после этого модуль разбивает строку на слова и ищет в таблице базы данных какой статье принадлежат все или часть из этих слов. Это достигается путем составления запроса на языке SQL к таблице «ARTICDC» с использованием объединения UNION. На рисунке 4 изображена реализация поисковика.

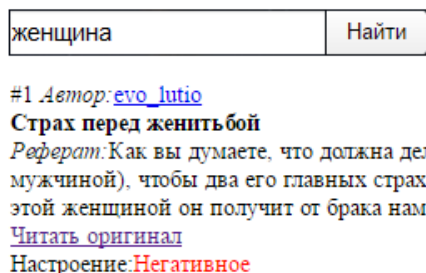


Рис. 4. Результат поиска

Пользовательский интерфейс реализован в виде интернет-страницы на языке HTML и PHP. На рисунке 5 изображена главная страница интернет-ресурса «Мнение»

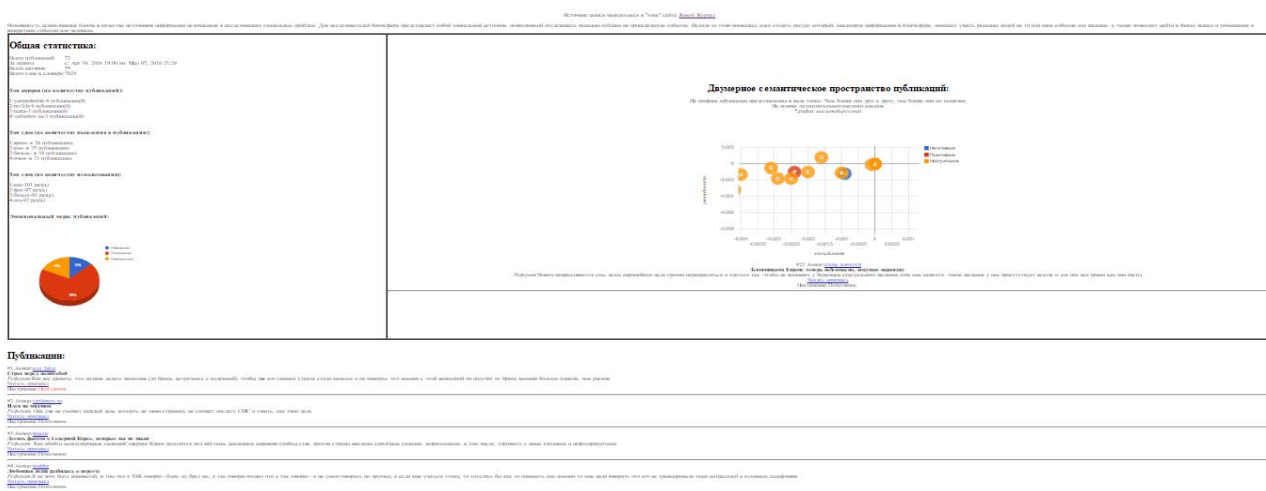


Рис. 5. Главная страница интернет-ресурса «Мнение»

Пользовательский интерфейс разделен на 4 части:

- Поисковая строка (сверху страницы)
- Общая статистика (слева страницы)

Содержит общую статистику по всем публикациям. Например, какое слово чаще всего встречается в публикациях или самое часто используемое слово.

- График семантического пространства (справа страницы)

График является визуальным отображением семантического пространства. Его можно масштабировать. Точки которые изображены на графике это публикации. Чем ближе точки тем более схожие темы у публикаций. При клике по точке снизу выводится информация о публикации.

- Полный список публикаций (внизу страницы).

## **Литература**

1. *Барсегян А.* Технологии анализа данных: Data Mining, Text Mining, Visual Mining, OLAP. 2 изд.
2. <https://ru.wikipedia.org/wiki/Контент-анализ>.

# СОЗДАНИЕ ПРИЛОЖЕНИЕ ДЛЯ WINDOWS, УПРОЩАЮЩЕГО ВЗАИМОДЕЙСТВИЕ ПОЛЬЗОВАТЕЛЯ С СИСТЕМОЙ ПРИ КОПИРОВАНИИ ФАЙЛОВ

*Курдюмов А.А.,  
МТУСИ, студент группы БИК1403  
Загвоздкина А.В.,  
МТУСИ, доцент*

В современном мире, где практически каждый день выпускаются новые электронные устройства, обмен информацией между ними очень актуален. Ежегодно, большая часть из нас создает тысячи документов, переносит фотографии и видео с фотоаппаратов, видеокамер, разных форматов файлы со смартфонов, планшетов и других переносных компьютеров. Нередки случаи, когда при поломке устройства пользователи лишались всего того контента, что был на них. Возможно, это основная причина, почему многие хранят все данные на жестком диске обычного компьютера. Другая причина переноса данных – возможность редактирования файлов на другом устройстве, которое может повысить качество созданных документов, либо же повысить комфорт при работе. С развитием технологий появилось несколько решений для переноса данных с устройства на компьютер: «по воздуху» (при помощи Wi-Fi, Bluetooth или других беспроводных сетей поддерживаемых устройством) или подключением устройства к компьютеру по кабелю. Из-за этого на рынке электроники существует много устройств, позволяющих копировать данные со всех, даже с переносных компьютеров пользователя на свой собственный отдельный носитель «по воздуху», чтобы избежать потери информации в случае их поломки. Процесс в данном виде техники идет автоматически, и человек даже не замечает того, что создается копия данных на резервном носителе. Как ни печально, но передача файлов с большими размерами «по воздуху» может занять долгое время, а передача через некоторые сети требует немалых денег из-за тарифов операторов этих сетей. Проблема с безопасностью при такой передаче данных тоже актуальна. Поэтому большинство пользователей подключают носители по кабелю.

С одной из проблем, подтолкнувшей меня к созданию данной проектной работы, я столкнулся, когда у меня сломался компьютер, и возникла необходимость перенести данные с жесткого диска сломанного компьютера на жесткий диск рабочего. Конечно, нет ничего сложного в переносе файлов общим объемом 500 гигабайт, но так получилось, что файлы были вперемешку. Не знаю как большинство, но мои знакомые и я для удобства сортируем по папкам данные на компьютере (так легче найти что-то либо на его носителе). Если файлов мало, то перенести их в нужную папку не составляет большого труда, но если информации (как в моем случае) много, то весьма затруднительно отсортировать ее. Во время переноса файлов у меня возникла идея попробовать найти какое-нибудь приложение, которое поможет мне в их сортировке. Но, как ни печально, я так и не смог найти никаких отдельных приложений позволяющих копировать файлы с сортировкой по их формату. Такая программа смогла бы облегчить перенос информации между устройствами не только мне, но и другим пользователям. Тогда в мою голову пришла идея попробовать свои силы в программировании и написать свое собственное приложение.

На данный момент существует сотни языков, но в большинстве своем, они являются усовершенствованными вариациями нескольких популярных языков программирования, таких как: **Assembler, Basic, Pascal, C, Java и других** (приложение №1). Каждый из языков имеет свои преимущества, но у каждого плюсов должны быть и минусы. Простые языки не могут выполнять сложные задачи, а для простых задач сложные языки необоснованно усложняют создание проектов. Основным языком, как я понял, является C, или усовершенствованный его аналог C++ (на котором и написаны 70% программ, в том числе и Windows). Из всех языков я имел дело только с C++ и Pascal, но сейчас в институте изучаю Visual Basic.

Чтобы создать сами файлы, которые мы называем приложениями, существуют специальные среды программирования, в которых код компилируется в конечный файл – компиляторы. Самыми популярными компиляторами являются: Microsoft Visual Studio и Embarcadero RAD Studio. Для себя, в качестве утилиты для создания программы, я выбрал Delphi в комплекте Embarcadero RAD Studio, которая позволяет создавать качественные проекты как для обычных компьютеров так и (в последних версиях Delphi) для смартфонов с планшетами.

Если говорить об истории среды разработки ПО Delphi, то ранее она носила название Object Pascal и первая ее версия (Delphi 1) была создана фирмой Borland в 1995 году. На данный момент среда принадлежит и разрабатывается Embarcadero Technologies и, последняя ее версия RAD Studio XE5 (на которой я и писал свою программу), появилась 11 сентября 2013 года (Приложение № 2). Embarcadero RAD Studio поставляется в четырех редакциях: Starter, Professional, Enterprise и Architect, каждая из которых является мощным решением для разработки и редактирования приложений, в зависимости от поставленных задач. Среда программирования предназначена для разработки прикладного ПО для операционных систем, таких как: Windows, Mac OS X, Linux, а также IOS и Android. Благодаря уникальной совокупности простоты языка и генерации машинного кода, среда позволяет достаточно низкоуровнево взаимодействовать с операционной системой, а также с библиотеками, написанными на C/C++. Созданные программы не зависят от стороннего ПО, а выделение и освобождение памяти контролируется в основном пользовательским кодом, что, с одной стороны, ужесточает требования к качеству кода, а с другой - делает возможным создание сложных приложений с высокими требованиями к отзывчивости (таких, как работа в реальном времени). В кросс-компиляторах для мобильных платформ предусмотрен автоматический подсчет ссылок на объекты, облегчающий задачу управления их временем жизни.

Следующий шаг – выбор дизайна и функций моего будущего приложения. Так как программы в большинстве своем пишутся для обычных пользователей, было решено сделать наиболее простое и понятное его исполнение – три кнопки. Первая кнопка будет выводить пользователю программы проводник, по которому он сможет выбрать папку с файлами для копирования, в то время как вторая – папку, куда нужно копировать. Третья кнопка – начать копирование. Также, рядом с первой и второй кнопками будут указаны адреса, откуда и куда копируются файлы. Даже при первом запуске программы пользователь не должен столкнуться с трудностями в использовании приложения.

Как я представлял себе алгоритм программы. Первая и вторая кнопки будут выводить пользователю проводник для выбора папок (откуда и куда копировать). Самая трудная часть разработки приложения будет при работе с третьей кнопкой. При нажатии на нее, программа, для начала, создаст в папке, выбранной второй кнопкой подпапки с названиями: картинки, музыка, видео, документы, другое. Далее, программа будет «сканировать» папку, выбранную первой кнопкой, и, если в папке есть не только файлы, но и другие папки (подпапки), то их тоже. Все это делается для того, чтобы приложение нашло все файлы в указанной первой кнопкой папке. После этого, программа будет подставлять формат файла в написанной мною перечень форматов допустимых для каждого из файлов типа: фотографии, музыка, видео, документы, для последующего копирования в соответствующую папку. Если формат файла не подходит ни одному из мною перечисленных, то файл копируется в папку «другое». Весь цикл копирования будет длиться до тех пор, пока все файлы не будут скопированы. Обдумав это, можно было приступить к написанию программы.

Вначале, я создал кнопки (Приложение № 3). Добавил TLabel, TLabel 2 для указания адресов первой, второй кнопок и добавил OpenFileDialog. Далее, переименовал кнопки и TLabel 1, TLabel 2 (Приложение № 4). Компонент TLabel служит для получения коротких надписей в форме. Обычно его используют как заголовки к другим компонентам, а также как табло для отображения информации, что я и использовал для отображения адресов файлов. OpenFileDialog же не визуальный компонент. Он предназначен для поддержки операции открытия файлов и способен работать с любыми их типами. При обращении к этому компоненту выводится стандартное окно открытия файла.

Затем, я решил заняться сортировкой файлов по их формату (то есть расширениями файлов). **Расширение файла** - это часть имени файла, которое отделяется от основного имени точкой. Прежде всего, по расширению можно понять какое содержимое находится в файле. Форматов существуют тысячи, но лишь с некоторыми мы сталкиваемся, поэтому я перечислю лишь основные форматы музыки, видео, документов и картинок.

Картинки: jpg, jpeg, gif, psd, bmp, tiff, png, ico, mop, nef, mrw, crw, cr2, tga, pnm, rpf, pcx, ico, raw.

Музыка: mt9, cda, mp3, wma, odd, wav, flac, ape, wv, midi.

Видео: mp4, avi, mpg, mkv, mov, dv, vhs, avchd, wmv.

Документы: STD, pub, accdb, jnt, xlsx, xls, pdf, doc, odt, txt, docx, lit, pages, text, pptx, docm.

Данные файлы записываю в массив.

Используя функцию CreateDir, создаю папки, куда будут копироваться отсортированные файлы и придаю им названия. Функция CreateDir создаёт папку в текущей директории. С помощью функции SearchFolder запускаю поиск всех файлов в указанной кнопкой 1 папке. Цикл поиска будет продолжаться до тех пор, пока не будут найдены все файлы в папке. Затем, используя функцию CopyFile, копирую файлы в соответствующую папку, при этом присваиваю номера для файлов каждого типа. Если файлы не подходят ни одному из форматов, то им присваивается номер 0, и они попадают в папку «другое». Если файлы удовлетворяют формату картинок, то им присваивается номер 1, и они попадают в папку «картинки». Если файлы удовлетворяют формату музыки, то им присваивается номер 2, и они попадают в папку «музыка». Если файлы удовлетворяют формату видео, то им присваивается номер 3, и они попадают в папку «видео». Если файлы удовлетворяют формату документов, то им присваивается номер 4, и они попадают в папку «документы».

И, наконец, для первой и второй кнопок, используя SelectDirectory, позволяю открывать окно для выбора папок. Функция SelectDirectory выводит диалоговое окно, чтобы позволить пользователю выбрать директорию.

Код написан (Приложение № 5). При моих тестах приложение доказало свою работоспособность и удобство. В качестве примера, я взял жесткий диск объемом 1 терабайт и решил перенести туда файлы с диска моего компьютера (в размере ~ 450 гигабайт), содержащего в нескольких десятках папок разного формата файлы. Моя программа не ускоряет копирование, но зато сразу сортирует все по типу файла, что и было успешно мною доказано. Единственные изменения, в программе после проверки – это добавление новых форматов файлов, которые я внес в свой перечень (в проектной работе указывается окончательный вариант).

Часть отзывов о программе: «Очень удобно переносить файлы с флешки, на которой у меня много файлов», «Использую программу для переноса фотографий и видео на компьютер. На фотоаппарате видео не отделяется в отдельную папку от фотографий. Ранее, мне было неудобно копировать их, но сейчас этот процесс я почти не замечаю». Мое приложение подойдет тем, кто сталкивается с переносом большого количества файлов. Например, при переносе файлов с фотокамер и видекамер на компьютер; при переносе данных с жестких дисков на другие жесткие диски через подключение к компьютеру; при копировании на них информации с устройств на ОС Android, часть данных iOS и многое другое. Приложение могут использовать в офисах для облегченного переноса большого объема файлов, профессионалы, сталкивающиеся с большим количеством информации, требующей переноса. Однако, тем, кто не часто копирует данные или не имеет необходимости в копировании многих файлов, приложение может показаться бесполезным.

Что касается проекта в целом, то полученные мною знания в дальнейшем помогут при создании других приложений уже на более профессиональной основе. Помимо этого, проектная работа помогла определить мои интересы и выбрать направление в жизни.

## Литература

1. Ресурсы интернета (блоги по программированию, сайты по программированию)
2. Книги: Pascal для программистов, Delphi.

# ИССЛЕДОВАНИЕ ПРАКТИЧЕСКОГО ИСПОЛЬЗОВАНИЯ МЕТОДОВ ПАРСИНГА ДАННЫХ ДЛЯ РАЗРАБОТКИ ВЕБ-СЛУЖБ НА ПРИМЕРЕ ПАРСИНГА МАТЕМАТИЧЕСКИХ ВЫРАЖЕНИЙ

*Акимов П.М., Губанкова Е.В.,  
МТУСИ, студенты группы БИН1403  
Гузев А.В.,  
МТУСИ, к.т.н., доцент  
Сосновиков Г.К.,  
МТУСИ, доцент*

В современном мире все более или менее сложные системы обмениваются сообщениями между собой. К примеру, после принятия запроса от клиента, сервер начинает разбор заданного запроса и, если требуется, распределяет обработку данных по дополнительным машинам. Здесь запросом и ответом могут быть различные данные от обычной текстовой строки до более сложных структур. Основной проблемой для сервера является быстрое и безошибочное выполнение указанных инструкций или обработка потока данных. Мы построили веб-сервис, который удаленно разбирает и вычисляет математические выражения, отправленные пользователем, и на этом примере мы хотим показать возможное решение проблемы обработки данных сервером.

## 1. Постановка задачи

Для примера рассмотрим обработку данных в текстовом виде, а именно математического выражения. Человек довольно легко сможет его посчитать, поскольку имеет понятия о приоритетах операций еще со школы.

Машина же воспринимает это выражение как поток символов, не имеющий никакого смысла. Для вычисления значения этого выражения ей нужно воспринять каждый символ или группу символов как число или знак операции. Также, если исходное математическое выражение содержит скобки, то нужно, чтобы компьютер выполнял операции в верном порядке, в соответствии с их приоритетами. Все эти действия образуют довольно сложный процесс, который проходит в несколько этапов.

Перед нами стоит задача написания программы, которая сможет быстро обрабатывать входные данные, а именно математические выражения различной сложности и дальнейшей реализации нашей программы в качестве веб-сервиса.

Для этого нужно вникнуть в суть процессов, описанных выше.

Разберем этапы обработки программой выражения, которое посылается на вход.

## 2. Этапы обработки входной последовательности символов

Анализ машиной любого математического выражения осуществляется в 2 этапа.

1) Первый этап — лексический анализ.

**Лексический анализ** — процесс разбиения введенной пользователем входной последовательности символов на так называемые **токены**. Токеном может быть цифра, оператор или любое другое отдельное математическое обозначение (знак), как функция или переменная. Этот этап обработки позволяет машине выделять числа и операции из входного потока для дальнейшей обработки. Здесь определяется тип токена (число, операция) и его смысловое значение. Токен с типом и соответственно подготовленным значением передается на вход синтаксического анализатора, где начинается второй этап обработки входной последовательности символов.

Пример: выражение  $9+24/4$  в процессе лексического анализа будет поделено на токены:  $9,+ ,24, /,4$  (запятые использованы для разделения токенов).

2) Второй этап — синтаксический анализ (парсинг).

**Синтаксический анализ** – это процесс, который определяет, принадлежит ли некоторая последовательность токенов (слов) какому-либо языку (напр., естественному или формальному), описанному правилами некоторой грамматики.

Задачей этого этапа является сконструировать эффективный анализатор, который определяет, является ли корректной входная строка.

Синтаксический анализ выполняет **парсер**, он же **синтаксический анализатор** — часть программы, преобразующая входные данные в структурированный формат. В нашем случае таким форматом является обратная польская нотация, о которой мы поговорим чуть позже. Отметим, что фазы лексического и синтаксического анализа тесно связаны и взаимодействуют друг с другом.

### 3. Форматы записи выражения

Многие программы (скриптовые языки, компиляторы и калькуляторы) требуют парсинга и вычисления выражений на ходу. Соответственно нам необходимо, чтобы выражение было записано в определенном формате, удобном для подсчета.

#### 3.1 Инфиксная запись

Итак, мы будем вводить в программу некое математическое выражение, например,  $9+24/4$ . Такое представление выражения называется **инфиксным**. Это обычный способ записи математических и логических выражений, при котором знак бинарной операции находится между операндами.

Мы все очень хорошо знакомы с инфиксной записью. Конечно, каждый из нас сможет довольно быстро вычислить значение представленного выражения и получить ответ - 15.

Хотя наши действия кажутся вполне естественными, то, что мы сделали, не так легко, как кажется на первый взгляд. Решение этого примера потребовало от нас сканирования выражения и расположения всех операторов в порядке приоритета, то есть, парсинга.

В отличие от нас, компьютер не может сразу посчитать выражение в инфиксной записи, поскольку не имеет понятия о том, как разделять числа и операции, а также о приоритетах операций. Обработка инфиксной записи для него неудобна.

#### Постановка проблемы выбора промежуточной формы записи выражения.

Как видим, основную проблему для нас представляют скобки и приоритеты. Именно они повышают сложность алгоритма. Существенно снизить сложность разбора выражения, а также затрат памяти может помочь некоторая **промежуточная форма** записи выражения, решающая проблему скобок и приоритетов. Разделение чисел и операций на синтаксические единицы решается выполнением лексического анализа выражения - разбиения его на токены. Приоритет операций же проще всего задать, сразу расположив операции в том порядке, в котором они будут выполняться. При таком порядке операций отпадает необходимость в скобках. Такими свойствами обладает польская запись, или польская нотация.

#### 3.2. Польская нотация

**Польская запись (нотация)** - это альтернативный способ записи арифметических выражений, преимущество которого состоит в отсутствии скобок. Существует два типа польской записи: **прямая** и **обратная**, также известные как **префиксная** и **постфиксная**. Отличие их от классического, инфиксного способа заключается в том, что знаки операций пишутся не между аргументами, а до в случае префиксной и после в случае постфиксной записи.

Пример: инфиксное выражение  $9+24/4$  выглядит следующим образом в польских нотациях.

Прямая польская нотация                             $+; /; 24; 4; 9$

Обратная польская нотация     $9; 24; 4; /; +$

### 3. Алгоритм Сортировочной станции

Из двух вариантов записи выражения, удобных для дальнейшего парсинга, для программной реализации будем использовать обратную польскую запись, так как ее получение из инфиксного выражения можно осуществить посредством простого алгоритма сортировочной станции.

Этот алгоритм изобрел нидерландский учёный Эдсгер Дейкстра. Алгоритм получил свое название за сходство его операций с происходящими на железнодорожных сортировочных станциях перестановками. Как и алгоритм вычисления ОПН, алгоритм сортировочной станции основан на стеке. Это достаточно простой алгоритм, вместе с тем элегантный, мощный и быстродействующий.

Для преобразования в обратную польскую нотацию используется 2 строки: входная и выходная, а также:

- 1 стек для хранения операторов, еще не добавленных в выходную очередь

- 1 очередь для вывода
- 1 массив (или другой список токенов)

**Стек** - структура организации данных, основанная на принципе *LIFO*.

Для перевода между нотациями используется стек и выходная очередь. Входная строка разбивается на токены. Считывание входной строки происходит по одному токену и в зависимости от того, к какой категории он относится, производится одно из возможных действий.

В конце выполнения этого алгоритма очередь вывода содержит токены в порядке обратной польской записи. Поскольку это очередь вывода, она должна быть прочитана с первого токена, который был в нее помещен, до последнего. (по принципу *FIFO*).

При использовании этого алгоритма следует обратить внимание, что скобки не являются операторами.

Алгоритм также может работать с функциями и параметрами, разделенными запятой. Он не ограничивается работой с числовыми выражениями, удобен для строковых данных, алгебры логики и операций с классами.

#### **Алгоритм сортировочной станции**

Пока имеются токены, которые нужно прочесть:

Читаем токен.

Если это число, добавляем его в очередь.

Если это оператор:

Пока на вершине стека есть оператор с более высоким приоритетом:

Выталкиваем операторы из стека и помещаем в очередь вывода.

Помещаем текущий оператор в стек.

Если токен – левосторонняя скобка «(», помещаем его в стек.

Если токен – правосторонняя скобка «)»:

Пока на вершине стека нет левосторонней скобки:

Выталкиваем операторы из стека и помещаем в очередь вывода.

Выталкиваем левостороннюю скобку из стека и просто отбрасываем ее.

Пока имеются операторы на вершине стека, выталкиваем их из стека и помещаем в очередь вывода.

#### **4. Алгоритм автоматизации вычисления выражений в ОПН для стековой машины**

Автоматизация вычисления выражений в обратной польской нотации основана на использовании стека. **Стековой машиной** называется алгоритм, проводящий вычисления по обратной польской записи. Реализация стековой машины, как программная, так и аппаратная, чрезвычайно проста и очень эффективна.

##### **Алгоритм вычисления для стековой машины**

###### **1) Обработка входного символа**

Если на вход подан операнд, он помещается на вершину стека.

Если на вход подан знак операции, то соответствующая операция выполняется над требуемым количеством значений, извлечённых из стека. Результат операции кладётся на вершину стека.

###### **2) Если входной набор символов обработан не полностью, перейти к шагу 1.**

**3) После полной обработки входного набора токенов результат вычисления выражения лежит на вершине стека.**

Следуя этому алгоритму, компьютер с легкостью вычислит значение выражения, представленного в обратной польской записи.

На основе описанных алгоритмов мы реализовали вариант калькулятора в виде веб-сервиса, выполняющего обработку математических выражений, поступивших с клиентской стороны.



<pre> ***** Calculator (host) ***** The service is ready. Press the Enter key to terminate service. Calculator awaits your expression... Expression recieved: 2+2*2 Result sent: 6  Calculator awaits your expression... Expression recieved: SIN(PI/4)-3 Result sent: -2,29289 </pre>	<pre> ***** Calculator (client) ***** Your expression: 2+2*2 Response: 6  Another one? Y N Y  Your expression: SIN(PI/4)-3 Response: -2,29289  Another one? Y N </pre>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Для того чтобы убедиться в оптимальности выбранных нами алгоритмов мы выполнили следующее исследование. Мы посылали на вход разные по длине математические выражения и с помощью таймера замеряли время подсчета программой каждого выражения. Для уточнения полученных результатов такой эксперимент был проведен дважды. При усреднении полученных графиков мы получили линейную зависимость времени подсчета выражения от количества токенов в этом выражении. Это говорит о том, что выбранные алгоритмы оптимальны и программа имеет хорошую производительность.

### 5. Реализации обработки структурированных данных

Сейчас широкое применение получила технология *DOM*, позволяющая программам или скриптам получать доступ к содержимому структурированных данных. Однако эта система обладает своим недостатком - ей требуется сначала получить весь документ и только потом будет начат разбор данных. Так же есть альтернативная технология *SAX*, которая читает данные сразу во время приема и может обработать некоторые части документа еще до завершения загрузки, что ускоряет общее время обработки и выполнения принятых данных. В этом ее существенный плюс и преимущество перед технологией *DOM*. Однако *SAX* не получила широкого распространения. Наша реализация парсера очень похожа на *SAX*-обработчик и после некоторой доработки может работать ничуть не хуже.

### Литература

1. [https://ru.wikipedia.org/wiki/Лексический\\_анализ](https://ru.wikipedia.org/wiki/Лексический_анализ)
2. [https://ru.wikipedia.org/wiki/Синтаксический\\_анализ](https://ru.wikipedia.org/wiki/Синтаксический_анализ)
3. [https://ru.wikipedia.org/wiki/Обратная\\_польская\\_запись](https://ru.wikipedia.org/wiki/Обратная_польская_запись)
4. *Гузеев А.В.* Формирование распределения вероятностей появления отдельных сообщений источника при статистическом кодировании // Т-Comm: Телекоммуникации и транспорт. 2010. Т. 4. № 6. С. 12-16.
5. *Щукарёв И.А., Гузеев А.В.* Оптимизация отображения иерархической структуры данных на реляционную модель средствами ORACLE // Телекоммуникации и информационные технологии. 2015. Т. 2. № 2. С. 37-39.
6. *Чернова А.А., Гузеев А.В.* Планирование инфокоммуникационных услуг на базе сервис-ориентированной архитектуры // Телекоммуникации и информационные технологии. 2014. Т. 1. № 2. С. 5-6.

# ПРОГРАММА ДЛЯ БИНАРНОГО ШИФРОВАНИЯ РОТОРНЫМ АЛГОРИТМОМ

Мирошниченко А.В.,  
 МТУСИ, студент группы БИН1401  
 Семенова Т.И.,  
 МТУСИ, к.т.н. доцент

Роторный алгоритм является одним из самых известных алгоритмов шифрования. Однако изначально он является механическим, а не программным. В данной работе, я попробую разобраться в роторном алгоритме и реализовать его в виде программного обеспечения.

Роторный алгоритм возник в 1920е годы. На его основе было создано большое число шифровальных машин.

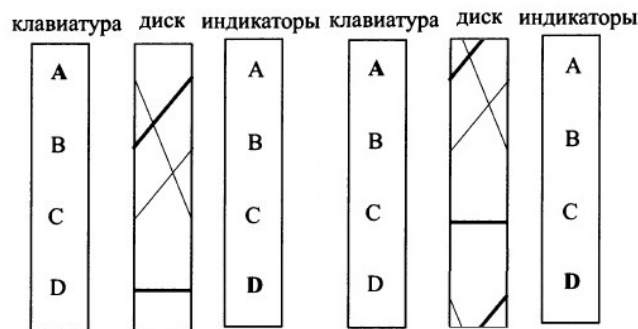


Рис. 1. Роторный алгоритм

Общий алгоритм работы представлен на рис.1. На вход с клавиатуры подается символ, например *C*. После чего он поступает на ротор (диск), где происходит замена этого символа, на другой. Так символ *C* заменяется на *B*. После шифрования символа, ротор меняет свою позицию, и символ *C* будет кодироваться (заменяться) на *C*. Таким образом, система имеет 4 разных шаблона шифрования, которые повторяются каждые 4 символа. То есть текст *AAAAAAA* будет зашифрован как *CDDACDDA*.

В машинах 1940 годов, было 3 ротора, по 30 символов, что давало  $30^3 = 27.000$  шаблонов шифрования. Современные шифровальные машины имеют уже  $40^{10} = 10^{16}$  возможных позиций роторов.

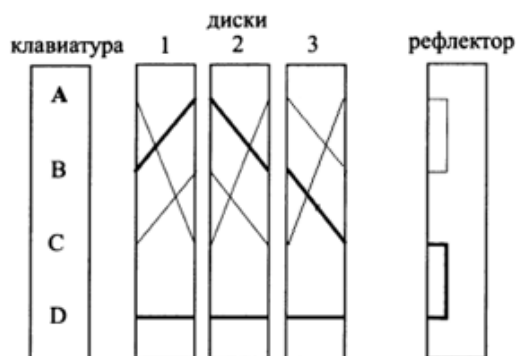


Рис. 2. Использование рефлектора

Важным шагом в развитии шифровальных машин с роторным строением стал рефлектор рис. 2.

Рефлектор выполнял очень важную функцию, он соединял два контакта между собой, то есть на рис. 2, символ *B* будет зашифрована как *D*, а символ *D* как *B*. Если бы рефлектора не было, то символ *B* закодировался в *C*, а символ *C* в *A*. Добавление рефлектора очень сильно облегчает алгоритм расшифровки. Без рефлектора, для расшифровки, надо взять зашифрованный текст и пропустить его через системы задом наперед, то есть, сделать алгоритм, противоположный алгоритму зашифровки. Но с рефлектором мы можем взять исходный текст, пропустить его повторно по тому же алгоритму, по которому он шифровался, и мы получим исходный текст. Пример: *ABC* – исходный текст – пропускаем его через роторы – получаем *CDA* – повторно шифруем уже зашифрованный текст – получаем на выходе наш исходный текст *ABC*. Подобные алгоритмы называются симметричными, так как для зашифровки и расшифровки используется один и тот же алгоритм.

В создаваемой мной программе будет использоваться 5 роторов (данное число связано с ограничением на размер ключа, подробнее об ограничении можно посмотреть в постановление Правительства Российской Федерации от 16 апреля 2012 г. *N* 313), каждый ротор будет иметь 256 контактов (байт представленный в виде числа, от 0 до 255). Таким образом, создаваемая программа будет иметь  $256^5$  комбинаций роторов, и будет абсолютно криптостойкой, при шифровании файла размером до  $256^5$  байт или 1 Тбайт.

Прежде чем приступить к созданию программы, было проведено исследование скорости работ циклов в разных языках программирования. Для каждого языка был запущен цикл от 1 до 1.000.000 и измерено время выполнения программ.

Язык	<i>BASIC – Visual studio 2010</i>	<i>C# - Visual studio 2010</i>	<i>Fortran - silverfrost fn95</i>	<i>Python – Python 3.5</i>	<i>Ada – GHAT</i>
Результат, с	18,632	27,446	6,932	267	2,466

Как можно судить из измерений, *Ada* является наиболее быстрым языком программирования, однако создания приложений на этом языке является очень сложным и трудоемким, поэтому в качестве оптимального языка был выбран *Fortran*. Но на *Fortran* крайне проблематично создать интерфейс, поэтому интерфейс к программе был создан в *Visual studio 2010* на языке *C#*, а все вычислительное ядро было создано в среде *silverfrost fn95* на языке *Fortran*.

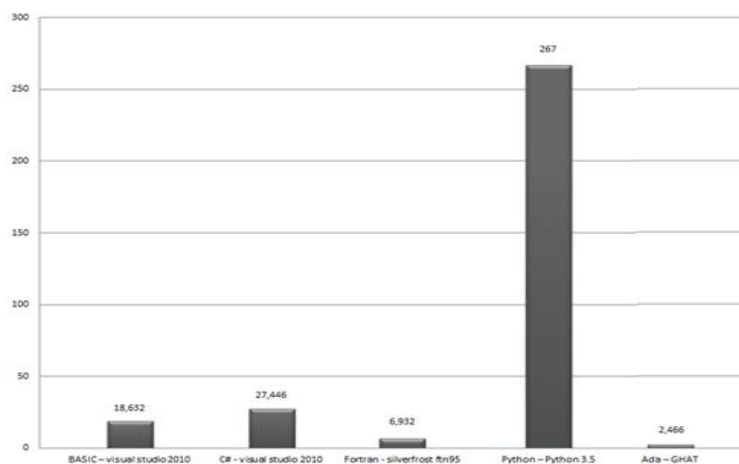
Пользовательский интерфейс, созданный на *C#*, не представляет особого интереса, в нем задаются пути к файлам, пароль, запускается вычислительное ядро на *Fortran*.

На рис. 4 представлен укрупненный алгоритм работы ядра.

Таким образом, программа берет исходный файл:

- 1) считывает из него байт (число от 0 до 255)
- 2) задает позиции роторов, исходя из номера байта, а так же пароля
- 3) пропускает байт через систему роторов
- 4) записывает зашифрованный байт в конец нового файла
- 5) проверяет, весь ли был считан исходный файл, если нет – то считывает следующий байт

Для наглядности результаты представлены в виде диаграммы рис. 3.



**Рис. 3.** Время выполнения цикла

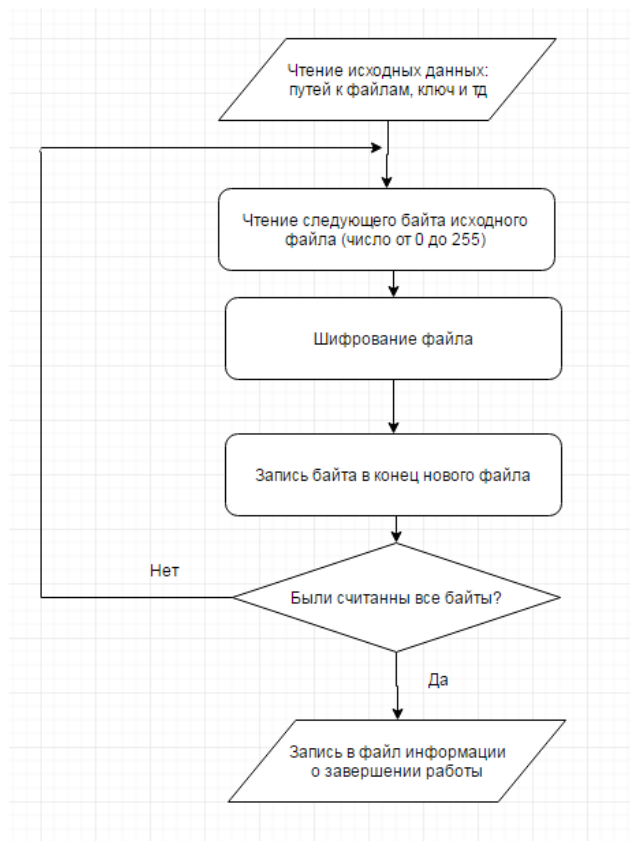


Рис. 4. Алгоритм работы

Наибольший интерес представляет система эмуляции роторов. Если в механическом виде, это был диск с контактами, то в программном виде существует несколько способов его реализации. Так как алгоритм замены в роторах должен быть случайным, поэтому использовать какую-либо формулу мы не можем, и надо идти в лоб. Первый способ это большое число *if (case)*. Но, от идеи использовать в программе 2.500 *if*, решено было отказаться. Другой способ эмуляции ротора – это массивы. Создается массив, в котором индексы это номер входного контакта, а значение в ячейках – номера выходных контактов. Пример: на вход поступает число 112, программа берет первый массив, моделирующий первый ротор, и в ячейки с номером 113 (так как нумерация массивов в Fortran идет с 1, а не 0, то к индексу надо добавить 1), а в ячейке задано число 10. Число 113 шифруется в 10, и в следующий ротор подается на 11 контакт (элемент массива).

Следует обратить внимание, что описанный алгоритм работает, при прохождении сигнала до рефлектора. На обратном пути, от рефлектора, уже значения в массиве имеют значения входного контакта, а их индексы – выходного. Говоря простым языком, если после рефлектора у нас получилось число 10, то мы должны пройти по всем элементам массива, пока не найдем число 10, а потом считать индекс ячейки, в которой был номер 10 и отнять 1, тогда мы получим нужное нам (зашифрованный) выходной байт.

Так как поиск нужного значения в массивах занимает много времени, мною был оптимизирован алгоритм. Было созданы дополнительные 5 массивов, которые эмулировали обратное прохождение роторов. Они отличались от исходных тем, что значения индексов исходных массивов находилось в ячейках нового, под индексами, равными содержимому исходного массива. Это значит, что если  $X1(10) = 50$ , то в новом массиве  $X2(50) = 10$ . То есть, индекс, и содержание ячеек поменяются местами. Это позволит нам не искать нужное значение на каждом шаге, а сразу обращаться к индексам.

Пример шифрования программой приведен на рис. 5. Верхний текст – исходный, в середине – зашифрованный, а нижний – расшифрованный. Очень хорошо видно, как в процессе шифрования полностью меняется вся структура файла.

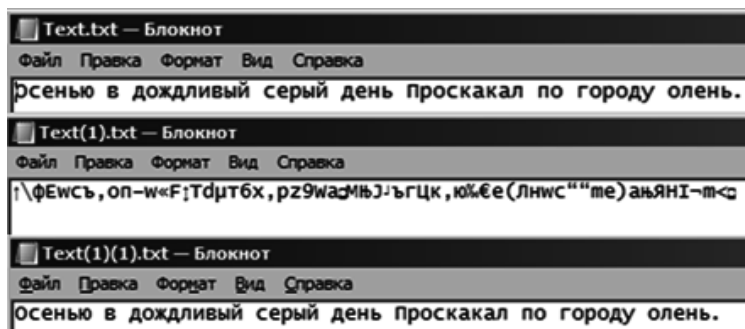


Рис. 5. Результат работы

Аналогичным образом изменяется структура файлов любого типа (картинок, видеозаписей...)

### Литература

1. Рижменантс, Дирк. Технические детали Энигмы. Шифровальные машины и криптология, 2004.
2. Ulbricht, Heinz Die Chiffriermaschine Enigma — Trügerische Sicherheit: Ein Beitrag zur Geschichte der Nachrichtendienste, 2005.

# РАЗРАБОТКА ЭЛЕКТРОННОГО МОДУЛЯ ПОСЕЩАЕМОСТИ ДЛЯ ДЕКАНАТА ВУЗА

*Морозов И.С.,  
МТУСИ, студент группы БСТ1503  
Гуриков С.Р.,  
МТУСИ, к.п.н.*

В настоящее время для контроля посещаемости занятий студентов Московского технического университета связи и информатики используются данные, которые предоставляют старосты групп по итогам каждой недели в деканаты соответствующих факультетов. Однако, человеческий фактор, бумажная форма отчетности, последующая обработка информации, занимающая много времени, отсутствие возможности оперативного контроля посещаемости привели к необходимости создания программы (модуля), которая решала бы задачи представления информации о посещаемости студентов в электронном виде.

В частности, в задачи будущего проекта входили:

- разработка страницы старосты группы, на которой имеется;
- возможность внесения информации о количестве пропущенных часов;
- реализация интерфейса для просмотра статистики по каждому студенту, для каждой отдельной группы студентов за определенный промежуток времени;
- разработка системы поиска по группам и студентам;
- обеспечение гибкой системы управления отчетными периодами.

Таким образом, целью создания электронного модуля посещаемости для деканата ВУЗа являлась автоматизация процесса учета посещаемости студентов деканатами факультетов, удобство ведения учета и анализа полученных сведений от старост групп, повышение надежности хранения информации.

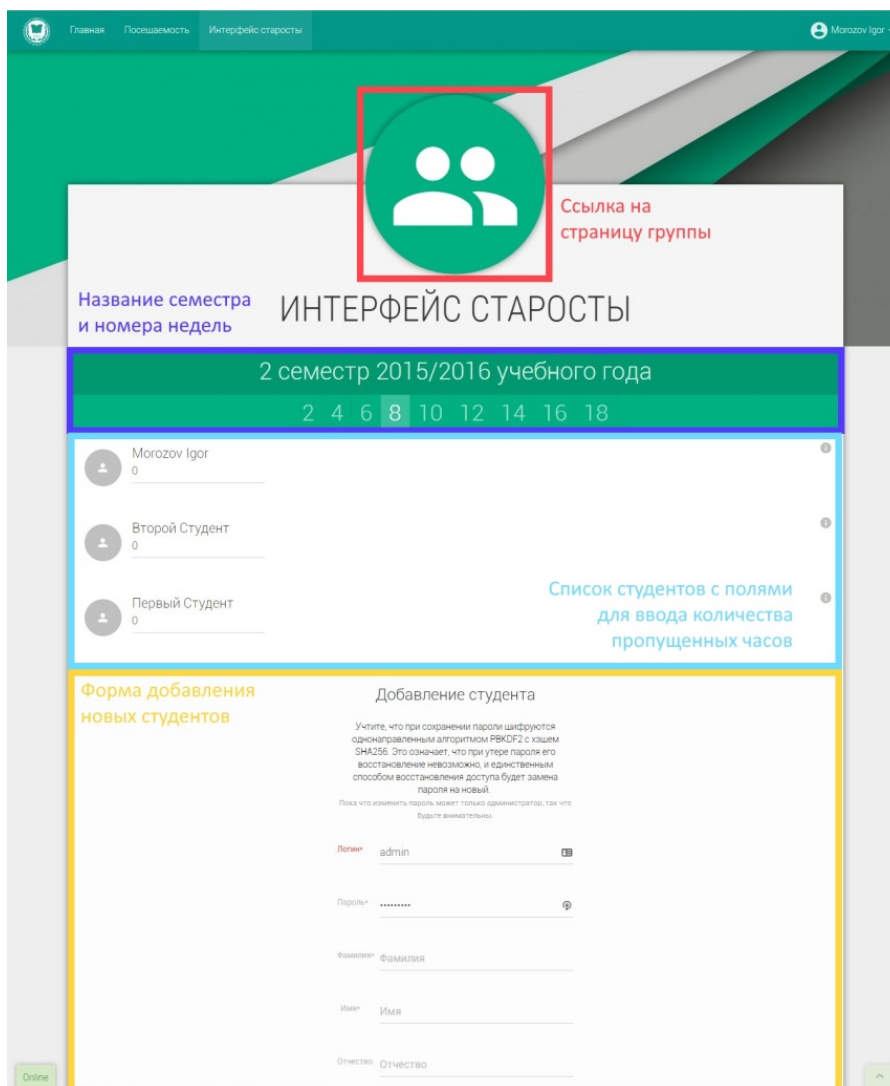
Наличие сведений об истории посещаемости занятий студентом, позволяет сделать выводы о влиянии пропусков занятий на успеваемость, оперативно принимать меры в отношении лиц, пропускающих занятия без уважительной причины.

При выборе средств разработки учитывалось, что программа будет использоваться небольшим количеством людей, но при этом ей будет необходимо производить множество относительно сложных расчетов, использовать ресурсы сервера. В результате выбор был остановлен на простой и надежной платформе Django, основанной на языке программирования Python.

Перейдем к характеристике основных составляющих разработанного нами электронного модуля посещаемости. Так, при разработке Кабинета старосты (рис. 1) решались следующие задачи:

- реализация проверки доступа, в частности, только авторизованные пользователи с правами старосты могут зайти на соответствующую страницу;
- возможность отображения текущего семестра в зависимости от даты;
- возможность добавления студентов в группу;
- возможность быстрого просмотра статистики для каждого студента.

Доступ к странице старосты могут получить лишь пользователи с необходимыми правами, для всех остальных будет отображена страница с кодом ошибки. Неавторизованные пользователи перенаправляются на страницу входа. При первом входе в программу, староста получает предложение пройти обучающий курс, в процессе которого он познакомится с ее возможностями.



**Рис. 1.** Кабинет старосты

Доступ к странице старосты могут получить лишь пользователи с необходимыми правами, для всех остальных будет отображена страница с кодом ошибки. Неавторизованные пользователи перенаправляются на страницу входа. При первом входе в программу, староста получает предложение пройти обучающий курс, в процессе которого он познакомится с ее возможностями.

Для того, чтобы работники деканата могли осуществлять администрирование электронного модуля, создана Панель управления (рис. 2), на главной странице которой, имеется доступ к карточке с графиком, отображающим тенденцию общего количества пропущенных часов студентами за определенный период и карточке, служащей для генерации отчетности. В последней можно указать конкретную группу или сразу для всех групп, при этом будет сформирован отчет в формате Microsoft Excel.

Среди возможностей Панели управления можно выделить:

- создание новых отчетных периодов несколькими способами: по начальной и конечной неделе с заданным шагом и по номерам конкретных недель;
- удаление существующих отчетных периодов;
- поиск по пользователям и группам;
- возможность автозаполнения;
- быстрый просмотр информации о найденном пользователе.

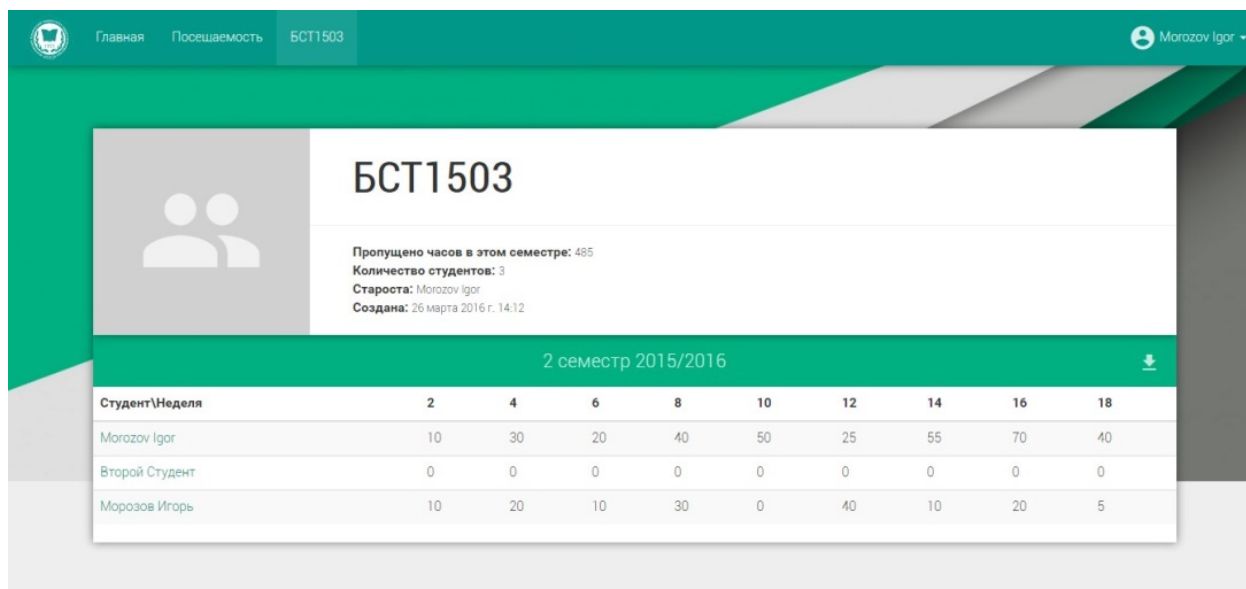


*Рис. 2. Интерфейс Панели управления*

Разработанная нами Страница группы (рис. 3) представляет собой сводку всей актуальной информации: количество студентов, имя старосты, когда группа была создана, а также общее количество пропущенных часов студентами в текущем семестре.

Кроме того, имеется возможность просмотреть сведения, отражающие общее количество часов для всех студентов по неделям семестра.





*Рис. 3. Страница группы*

Другие страницы разработанного электронного модуля посещаемости позволяют старостам просматривать статистику пропусков занятий группы, а студентам свою собственную статистику. Также в целях мотивации студентов формируется список из пяти лучших студентов и пяти лучших групп, с точки зрения наименьшего количества пропусков занятий.

Таким образом, разработанный электронный модуль посещаемости для деканата ВУЗа, позволяет облегчить обязанности работников деканата и старост, предоставляя при этом множество дополнительных возможностей, например, генерация готовой к печати отчетности, обновление статистики в реальном времени, визуализация данных, создание списка лучших студентов, групп и др.

Следует отметить, что большое значение имеет разработанный адаптивный дизайн приложения, что позволяет использовать программу на устройствах с любыми размерами экрана. Благодаря реализации такой возможности, старосты групп могут заносить данные непосредственно в стенах учебного заведения с любого мобильного устройства. Активное использование технологии асинхронных запросов (AJAX) существенно снижает количество трафика и увеличивает скорость работы с приложением.

В настоящее время, разработанный электронный модуль посещаемости проходит период тестирования в деканате общетехнического факультета (ОТФ-1) Московского технического университета связи и информатики.

## Литература

1. *Аркаева Р.П.* Квалиметрический подход к реализации балльно-рейтинговой системы контроля и оценки знаний студентов вузов // автореф. дисс.... канд. пед. наук. Владикавказ, 2011. – 22 с.
2. *Гуриков С.Р.* Использование электронного модуля успеваемости и посещаемости студентов в техническом университете // Политематический журнал научных публикаций «Дискуссия». №1, 2015. С. 96-100.
3. *Миселимян Т.Л.* Мониторинг посещаемости учебных занятий как средство управления образовательным процессом: Дис. ... канд. пед. наук : 13.00.01 : Екатеринбург, 2002. – 220 с.

# ИСПОЛЬЗОВАНИЕ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ АВТОМАТИЧЕСКОЙ ИДЕНТИФИКАЦИИ ЛИЧНОСТИ ПО ИЗОБРАЖЕНИЯМ ЛИЦ В ВИДЕОПОТОКЕ

Пугачев Д.А.,  
МТУСИ, студент группы М111501(73)  
Шелухин О.И.,  
д.т.н., профессор

Системы автоматического распознавания лиц имеют множество возможностей применения и охватывают все больше сфер использования. Например, они могут использоваться для улучшения интерфейса человек-компьютер, для разграничения доступа к помещениям или оборудованию, поиска человека и для других целей. Стремительное развитие информационных технологий и других областей науки и техники создает как новые способы использования систем распознавания, так и возможности для их реализации.

В общем виде задача идентификации личности по изображению может быть разделена на обнаружение и локализацию лица, при необходимости предобработку изображения лица, идентификацию личности. Идентификация может проводиться на открытом или закрытом множестве записей в базе.

Идентификация может проводиться различными методами, которые можно разделить на эмпирические и методы моделирования лица. Эмпирические методы основываются на опыте человека в распознавании образов, методы моделирования лица являются частным случаем общей задачи распознавания образов. Эмпирические методы имеют более высокую скорость работы, в то время как методы моделирования лица имеют существенно более высокую точность и способны работать в более сложных условиях.

В методах моделирования лица в качестве характеристик изображения возможно использовать как исходные значения цветов пикселей, так и результаты их обработки. Например, вейвлеты Хаара, локальные бинарные шаблоны, двумерное дискретное косинусное преобразование. В дальнейшем на основе вычисленных характеристик изображения производится распознавание. Одними из классификаторов, используемых и при обнаружении лица на изображении, и при распознавании личности, являются сверточные нейронные сети.

В разработанной мной программе реализована простая сверточная нейронная сеть [1], состоящая из 14 слоев. Обобщенная схема архитектуры простой сверточной нейронной сети представлена на рисунке 1. В использованной мной архитектуре СНС среди скрытых слоев помимо свертки и подвыборки, также используется выпрямленный линейный модуль, являющийся одним из вариантов функции активации. Используемая мной архитектура нейронной сети представлена на рисунке 2. Для ее реализации я использовал библиотеку функций СНС MatConvNet [2]. Сеть производит распознавание личности по закрытому множеству записей в базе, то есть исходя из предположения, что идентифицируемый индивид точно присутствует в базе идентификаторов. При этом в качестве базы выступает сама нейронная сеть.

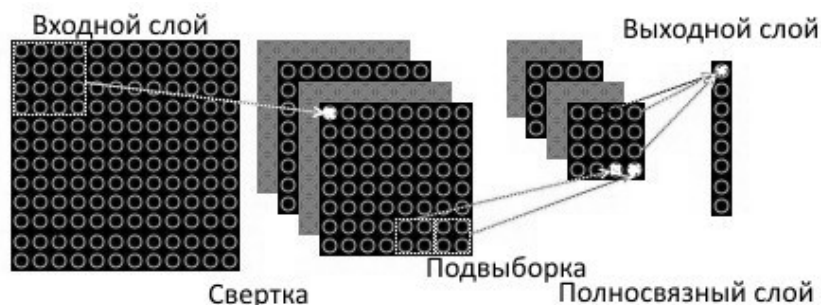


Рис. 1. Обобщенная схема архитектуры простой нейронной сети

Для экспериментов использовались различные базы изображений: The AT&T Database of faces; Face Recognition Data, University of Essex, UK [3]; Georgia Tech Face Database. Названные базы изображений находятся в свободном доступе и могут быть использованы для обучения и тестирования программы благодаря наличию в них нескольких изображений для каждого индивида, а также незначительных искажений этих изображений. Также для работы создавались собственные наборы изображений. Пример изображений из собственного набора представлен на рис. 3.

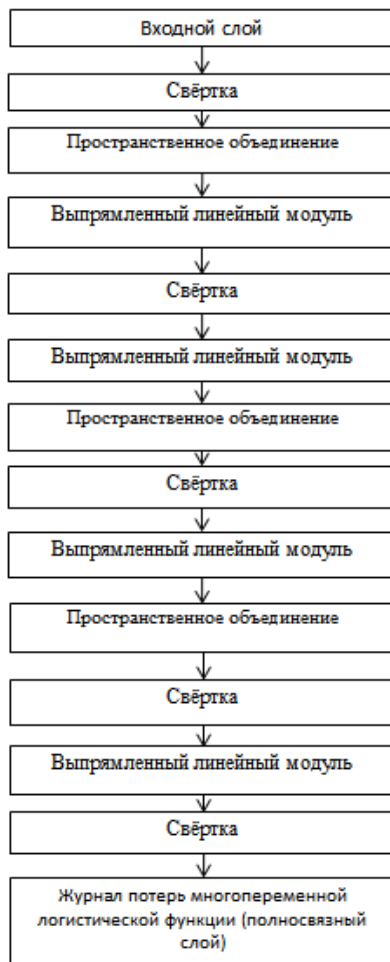


Рис. 2. Схема реализованной сверточной нейронной сети

В одном из экспериментов обучение СНС проходило на 3950 изображениях из базы FRD, по 10 на каждого субъекта из базы данных. При этом 9 изображений из 10 для каждого субъекта использовались для обучения сети, а 1 для проверки во время обучения. В описываемом эксперименте обучение проходило в течение 67 эпох. Вероятность правильного срабатывания на обучающих изображениях в процессе обучения вплотную приблизилась к 1. Графически это показано на рисунке 4 слева. Для проверки сети использовались 3950 изображений, не принимавших участие в обучении сети. Также проводились эксперименты с наборами искаженных изображений.

За основу были взяты изображения из базы FRD, не участвовавшие в обучении СНС.

В различных экспериментах к ним применялись следующие искажения: поворот изображения (на случайный угол до 1°, до 2°, до 3°, до 4°, до 5°, в случайном направлении по часовой или против часовой стрелки), изменение яркости изображения, искажение изображения гауссовским шумом. Вероятность правильного распознавания, полученная в этих экспериментах отображена на рисунке 4 справа.



Рис. 3. Примеры изображений из собственного набора

Для каждого из результатов распознавания проводилась проверка достоверности. Для этого максимальное значение в массиве выходных характеристик, умноженное на пороговый коэффициент, сравнивалось со следующим по величине. На рисунке 5 показаны ROC кривые, отражающие эффективность данной оценки в зависимости от порогового значения для четырех тестовых наборов изображений: исходные неизмененные изображения, повернутые на  $3^\circ$ , изображения с изменением яркости, изображения с гауссовским шумом. Площадь под кривой AUC при тестировании обычных изображений составила 0.8552, для изображений с поворотом 0.7811, для изображений с изменением яркости 0.7881, для изображений с гауссовским шумом 0.8145.

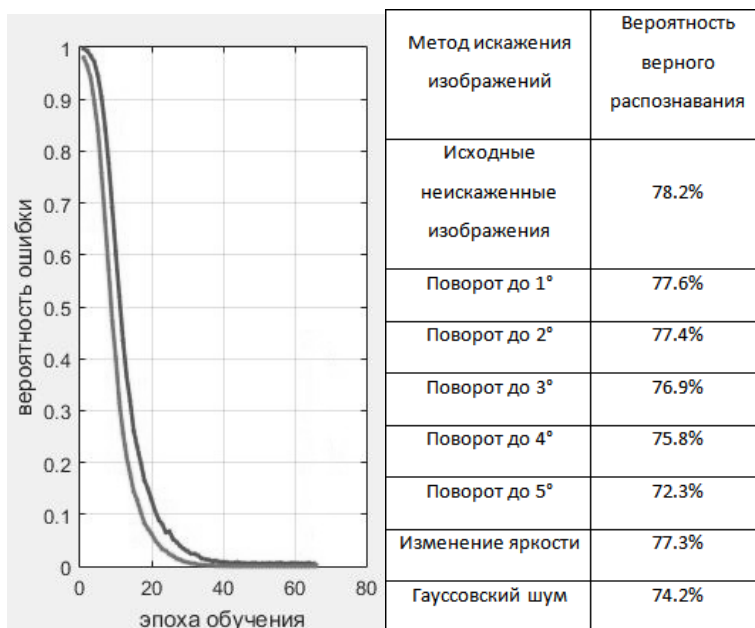


Рис. 4. Результаты обучения и тестирования СНС в одном из экспериментов с базой FRD

При распознавании личности по изображениям из видеопотока возможно получение нескольких разных кадров одного человека, что при полученной точности распознавания и оценки достоверности результата распознавания позволяет выстраивать из классификаторов по различным кадрам один общий классификатор с большей эффективностью, чем у отдельных классификаторов по отдельным кадрам.

Выявлено, что увеличение количества слоев приводит к повышению качества распознавания при снижении скорости работы. Установлено, что сверточная нейронная сеть в экспериментах показала значительную устойчивость к изменениям освещения, к искажению изображений и небольшую устойчивость к изменению масштаба изображений.

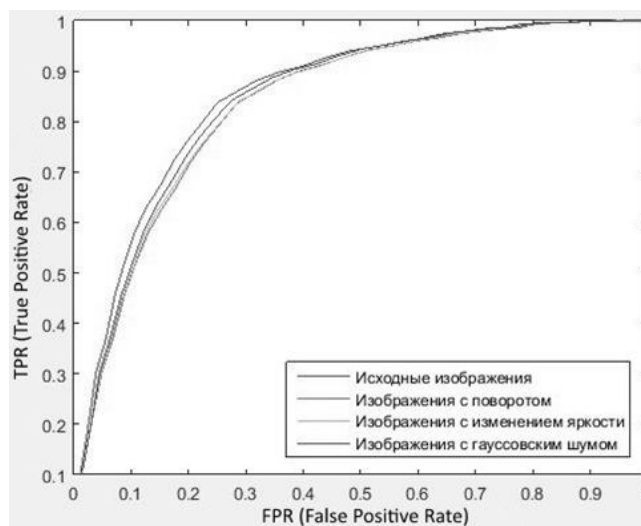


Рис. 5. ROC кривые оценки достоверности результатов распознавания

В результате тестирования разработанной программы выявлены следующие свойства реализованной СНС:

- Универсальность
- Достаточное быстродействие при распознавании
- Небольшая устойчивость к искажениям
- Распознавание только на закрытом множестве

Выявленные свойства реализованной СНС частично ограничивают возможности использования ее для распознавания по одиночным изображениям, оставляя возможность использования для распознавания по изображениям из видеопотока.

### Литература

1. Документация библиотеки MatConvNet <http://www.vlfeat.org/matconvnet/matconvnet-manual.pdf> (2016).
2. Библиотека функций СНС MatConvNet MATLAB toolbox implementing *Convolutional Neural Networks* (CNNs) for computer vision applications. <http://www.vlfeat.org/matconvnet/> (2016).
3. База изображений лиц Face image database Face Recognition Data, University of Essex, UK <http://cswww.essex.ac.uk/mv/allfaces/index.html> (2016).
4. Sheluhin O.I., Sirukhi J.W., Pankrushin A.V. Wavelet type selection in the problem of anomaly intrusions detection in computer networks using multifractal analysis methods // Т-Comm: Телекоммуникации и транспорт. 2015. Т. 9. № 4. С. 88-92.
5. Шелухин О.И., Иванов Ю.А., Арсеньев А.В. Оценка качества передачи видео в системах видеонаблюдения с помощью программно-аппаратного комплекса // Т-Comm: Телекоммуникации и транспорт. 2010. Т. 4. № 10. С. 81-84.

# РАЗРАБОТКА МУЛЬТИМЕДИА-СЕРВЕРА ПОД УПРАВЛЕНИЕМ RTLINUX

Креймер А.В., Михайлова А.В., Новиков А.С.,  
МТУСИ, студенты группы БСТ1201,  
Беленькая М.Н.,  
МТУСИ, доцент

С каждым годом объём трафика реального времени увеличивается во многом благодаря широкому использованию мультимедийного контента. К этой категории можно отнести не только просмотр потокового видео обычными пользователями, но и массовое применение систем видеонаблюдения. Цель работы – разработка в образовательных целях мультимедиа-сервера с учётом возможности его мониторинга и архивирования конфигурации.

Мультимедиа – это использование двух или более средств звуковой или визуальной информации в одном контенте, проигрываемом в течение определённого интервала времени [2].

Мультимедийная система – это система реального времени, в которой гарантировано время реакции на конкретное событие. Такое время называется дедлайном (от англ. *deadline*). Существует системы жёсткого реального времени (*hard real-time*) и мягкого реального времени (*soft real-time*). В первой получение корректного результата за дедлайн обязательно. Во второй можно превысить время выполнения, то есть получить корректный результат согласно некоторым приоритетам. Такие системы обеспечивают лишь хорошее среднее время реакции (примером может послужить платформа *Win32*) и, строго говоря, не являются реальным временем вообще [1].

Дедлайн для мультимедиа системы – это требования к передаче и обработке мультимедийных данных. Они определяются способностями человека распознать аудио и видео информацию. В мультимедиа основной проблемой является синхронизация изображения на экране со звуком. Последний обычно генерируется внешним аппаратным устройством с собственным таймером, и изображение синхронизируется с ним. Человек способен заметить довольно малые временные неоднородности в звуковом потоке, а пропуск кадров в визуальном потоке не так заметен. Расхождение звука и изображения фиксируется человеком уже при задержках около 30 мс. Поэтому системы высококачественного мультимедиа должны обеспечивать синхронизацию с такой же или более высокой точностью [1]. Дедлайн же для видео – это 24-30 кадров в секунду (*frames per second, fps*), не больше и не меньше. Основой здесь является мировой стандарт частоты киносъёмки и проекции – 24 *fps*.

Для передачи мультимедийных данных по сети используется специальная технология – *stream* (поток). Её применение обусловлено тем, что трафик реального времени чувствителен к среде передачи и задержкам. Есть два вида потоковой передачи (*Real-Time Streaming*):

- *Live-streaming* – основан на непосредственном получении потока данных от источника, например, от IP-камеры.
- *Video-on-demand (VoD)* – запрос на поток данных, когда-либо созданный и сохранённый во внешней памяти некоторого устройства, например, сервера.

Существуют специальные протоколы, позволяющие передавать потоковые данные:

- *RTP (Real-time Transport Protocol)* – протокол для передачи данных реального времени, работает поверх транспортного протокола *UDP*. В своём заголовке *RTP* передаёт данные, необходимые для корректного восстановления аудио и видео информации на приёмной стороне. Например, поле *timestamp* (временная метка) позволяет приёмнику не только буферизовать небольшое количество данных и проигрывать их через определённые промежутки времени, но и синхронизировать несколько потоков данных между собой. Буферизация на стороне получателя необходима для устранения возможной задержки при передаче данных, а также для возможности восстановления потерянных пакетов. Ведь в протоколе *RTP* не предусмотрены механизмы повторной передачи и подтверждения доставки – это слишком медленно для трафика реального времени. Количество потерянных пакетов можно определить по значению поля *sequence number* (порядковый номер), которое инкрементируется в каждом пакете [2].

- *RTCP (Real-time Transport Control Protocol)* – управляющий транспортный протокол реального времени. Он определён в *RFC 3550* вместе с *RTP*. На *RTCP* возложены функции поддержки обратной связи, синхронизации и именованя источников потоков. Важно отметить, что протокол не передаёт какие-либо медиаданные. В силу того, что сообщения *RTCP* рассылаются всем участникам сессии, периодичность рассылки таких сообщений динамически настраивается каждым участником.

- *RTSP (Real-time Transport Streaming Protocol)* – протокол, позволяющий удалённо управлять потоком мультимедийных данных. По синтаксису и операциям он схож с протоколом *HTTP*. Список некоторых команд:

- *DESCRIBE* – описание параметров мультимедийных данных;
- *SETUP* – установка логического соединения между принимающей и передающей сторонами;
- *PLAY* – старт передачи данных клиенту от сервера;
- *PAUSE* – приостановка передачи данных;
- *RECORD* – старт передачи данных от клиента серверу;
- *TEARDOWN* – остановка потока и разрыв логического соединения.

Для того чтобы корректно обрабатывать данные реального времени операционная система (ОС) мультимедиа-сервера должна отвечать определённым требованиям.

1. Ядро ОС должно поддерживать *Quality of Service (QoS)* – для текущей задачи (*task*) при любых условиях выделяется определённое количество ресурсов.

2. Расписание (*schedule*) задач должно переключать задачи реального времени за дедлайн. Если процесс по каким-либо причинам не укладывается во временные рамки – он не должен быть запущен. Планирование задач должно осуществляться согласно их приоритетам, которые являются статическими.

3. Серверный процесс должен поддерживать вышеуказанные протоколы потоковой передачи.

4. Если сервер должен осуществлять передачу мультимедийных данных в режиме *VoD*, необходимо, чтобы его подсистема ввода-вывода работала согласно одному из алгоритмов сканирования диска: *SCAN*, *LOOK*, *C-SCAN*, *C-LOOK*. Их суть заключается в том, что запросы на обработку данных исполняются не в том порядке, в котором они поступили, а в порядке близлежащего запроса к положению считывающей головки. Возможно использование и другого алгоритма, называемого алгоритмом «планирования по ближайшему сроку завершения» (от англ. *Early Deadline First, EDF*). Он может применяться не только к запросам ввода-вывода, но и к процессам в системах реального времени.

5. Поскольку ОС лично не занимается вопросами сжатия (компрессии) потока данных, то необходимо наличие специальной утилиты, выполняющей данную задачу. Все методы сжатия данных можно разделить на две группы: с потерей данных и без потерь. Первые применяются при сжатии видео- и аудиоданных и, следовательно, в мультимедийных системах. Вторые – при сжатии текстов, программ и т.п.

Ещё одно важное требование, предъявляемое к ОС маршрутизаторов – это поддержка *multicast* и *unicast* вещаний. Действительно, если роутер не поддерживает групповую рассылку (режим *unicast*), то мультимедиа-серверу необходимо создавать соединение для каждого из  $N$  хостов, запросивших некоторый поток данных, что требует от его сетевого интерфейса пропускной способности в  $N$  раз больше, чем необходимо для данного потока. Если же роутер поддерживает режим *multicast*, то мультимедиа-серверу достаточно создать одно соединение, которое будет общим для  $N$  хостов.

Благодаря современным методам сжатия данных требование к среде передачи ушло на второй план. Согласно американской компании *Netflix*, являющейся крупным поставщиком фильмов и сериалов в режиме потокового мультимедиа, для видео формата *4K* требуется пропускная способность в *25* Мбит/с, в то время как видео с разрешением в *1080p* требуют всего *3* Мбит/с. Тем не менее, было бы неправильно совсем не учитывать такое требование.

Существуют следующие метрики для мультимедийных систем:

- Пропускная способность канала.
- Латенция или задержка передачи данных – время между отправленными и принятыми данными.

- Джиттер (*jitter*) – это непредсказуемая задержка или «дрожание» между соседними кадрами видео потока.

- Битрейт (*bitrate*) – это скорость потока, то количество бит, которое выделяется для хранения одной секунды аудио или видео данных. Эту величину принято использовать при измерении эффективности передачи потока данных по каналу. Существует три типа битрейта: постоянный, переменный и усреднённый.

После краткого рассмотрения теоретических аспектов можно перейти непосредственно к разработке мультимедиа-сервера. Ключевым вопросом здесь является выбор ОС, под управлением которой будет функционировать серверный процесс. Этой ОС будет *RTLlinux*.

*RTLlinux* – это микроядерная ОС жёсткого реального времени, которая выполняет *Linux* как полностью вытесняемый процесс. Что же такое процесс *Linux*? Дело в том, что *RTLlinux* – это плагин, который можно подключить к любому дистрибутиву *Linux* и превратить его в систему жёсткого реального времени с исходным ядром в качестве процесса с наименьшим приоритетом, который может быть прерван по требованию в любой момент.

*RTLlinux* выполняет две важные функции: обработка прерываний и планирование задач реального времени. Если нет надобности в запуске задачи реального времени, то прерывание, обслуживаемое ядром *RTLlinux*, передаётся стандартному ядру. Фактически, *RTLlinux* выступает в качестве аппаратного обеспечения (*hardware*) для исходного ядра.

Одним из способов обмена данными между ядром реального времени и пользовательскими задачами *Linux* является применение разделяемой памяти (*shared memory*), доступной через системный вызов *mmap*. *RTLlinux* использует *Linux* для загрузки, доступа к большинству устройств, работы с сетью и файловой системой. Любой процесс реального времени для *RTLlinux* состоит минимум из двух нитей (*thread*) – первая обрабатывается ядром реального времени, вторая представляет собой обыкновенную нить, исполняемую исходным ядром *Linux*.

Достоинства ОС *RTLlinux*:

- Свободно распространяемая операционная система;
- Быстрый переход от обычного ядра *Linux* к *RT*-ядру;
- Требования к аппаратной части такие же, как и у исходного дистрибутива *Linux*.

Недостатки:

- Ядро *Linux* является процессом с наименьшим приоритетом, поэтому задачи *Linux* могут испытывать больше задержки.

Основная функция процесса, запускаемого на мультимедиа-сервере, состоит в обработке запросов от хостов сети (клиентов) на предоставление потока мультимедийных данных. При этом сервер обращается к источнику данных, например, *IP*-камере с поддержкой протоколов потоковой передачи, и с помощью команды *PLAY* протокола *RTSP* получает необходимый поток. Далее ретранслирует его клиенту с требуемыми параметрами.

На стороне клиента необходим процесс, который бы поддерживал вышеуказанные протоколы и осуществлял декомпрессию принимаемых данных. К тому же, буферизация принятых пакетов позволит убрать джиттер. Всем этим требованиям в полной мере отвечает, например, *VLC Media Player* – бесплатный кроссплатформенный медиа-плеер.

Для обеспечения стабильной работы мультимедийной среды необходим постоянный мониторинг параметров мультимедиа-сервера и *IP*-камер. После подробного анализа систем управления сетью (*NMS*) был сделан выбор в пользу программного продукта *Nagios XI*, менеджер которого установлен на отдельном сервере. Преимущества данной системы:

- Свободно распространяемая *NMS*;
- Легко настраиваемый и понятный интерфейс;
- Информативность;
- Открытый исходный код;
- Возможность лёгкого создания собственных плагинов на любом языке программирования;
- Оперативное информирование с помощью уведомлений по электронной почте и *SMS*.

Сбор статистики осуществляется с помощью множества протоколов, в числе которых *SNMP*, *NetFlow v5/7/9*, *RMON*, и возможен по таким параметрам как: загрузка *CPU*, загрузка *RAM*, температура *CPU*, состояние и пропускная способность сетевого интерфейса, свободное место на жёстком диске [3]. В список контролируемых параметров мультимедиа-сервера можно добавить значение битрейта для конкретного потока данных и состояние *IP*-камер, так как конфигурация последних осуществляется через сервер.

Для взаимодействия с системой *Nagios* существуют специальные приложения:



- *NSClient++* – это программное обеспечение, устанавливаемое на удалённых хостах под управлением *Windows* и позволяющее запускать плагины *Nagios* для мониторинга доступных параметров;

- *NRPE (Nagios Remote Plugin Executor)* – дополнение к *Nagios*, выполняющее те же функции, что и *NSClient++*, но для *UNIX*-подобных систем.

Для того чтобы в критических ситуациях иметь возможность быстрого восстановления работоспособности мультимедийной среды, необходим своевременный сбор и архивирование конфигурации как мультимедиа-сервера, так и *IP*-камер. Одним из способов решения этой задачи является сбор данных по протоколу *TFTP*.

*TFTP (Trivial File Transfer Protocol)* – это простой протокол передачи файлов, работающий поверх транспортного протокола *UDP*. Главным преимуществом протокола является скорость и простота передачи данных от хоста к серверу архивирования конфигураций и наоборот. Недостатком протокола является отсутствие аутентификации, хосты идентифицируются только по *IP*-адресу.

Для реализации архивирования требуется специальный процесс, работающий на отдельной машине. Этим процессом может быть, например, программный продукт *ftpd32* с открытым исходным кодом.

Процесс передачи данных происходит следующим образом. *TFTP* для инициализации соединения использует порт с номером 69, а для передачи данных выбирает не зарезервированные порты со случайными номерами с обеих сторон. Передача начинается с посылки запроса на чтение или запись файла от хоста к серверу, далее файл пересылается блоками по 512 байт. При получении очередного блока принимающая сторона посылает подтверждение о приёме. Посылка блока длиной менее 512 байт сигнализирует о конце файла. Если передающая сторона не получает подтверждения о приёме очередного блока данных, то происходит повтор передачи. В случае ошибки, например, отсутствия файла или прав доступа, потери доступа к ресурсу, любая сторона посылает сообщение об ошибке и завершает работу, не дожидаясь подтверждения. Проверка правильности передачи данных возлагается на подсчёт контрольных сумм протокола *UDP*.

В заключение работы стоит сказать о том, что описанная в статье мультимедийная среда организована на кафедре «Мультимедийных сетей и услуг связи» Московского технического университета связи и информатики и успешно выполняет возложенные на неё функции.

## Литература

1. *Иртегов Д.В.* Введение в операционные системы. 2-е изд., перераб. и доп. СПб.: БВХ-Петербург, 2008. 1040 с.
2. *Таненбаум Э., Уэзеролл Д.*, Компьютерные сети. 5-е изд. СПб.: Питер, 2012. 960 с.
3. [www.nagios.org](http://www.nagios.org). Система мониторинга *Nagios XI*.
4. *Хуторов В.С., Беленькая М.Н.* Основные проблемы и цели мониторинга базы данных средствами СУБД ORACLE // Т-Comm: Телекоммуникации и транспорт. 2013. Т. 7. № 7. С. 133-134.
5. *Малиновский С.Т., Беленькая М.Н., Спиридонов А.А.* Метод повышения производительности транспортного протокола TCP в глобальных корпоративных сетях передачи данных // Т-Comm: Телекоммуникации и транспорт. 2010. Т. 4. № 7. С. 39-42.

# АНАЛИЗ СЕТЕВОГО ТРАФИКА НА ОСНОВЕ КОЛИЧЕСТВЕННЫХ ХАРАКТЕРИСТИК ПОТОКА ТРАНСПОРТНОГО УРОВНЯ

Прохоров Д.О., Трофлянина Н.В., Фомин С.А.,  
МТУСИ, студенты группы БСТ1201,  
Беленькая М.Н.,  
МТУСИ, доцент

С каждым годом всё больше растут объёмы сетевого трафика, передаваемого по сети Интернет. Для решения многих прикладных задач требуется проводить анализ этого трафика – например, для организации классов обслуживания, для обнаружения нежелательного и запрещённого трафика и т.д.

Одним из средств, позволяющих проводить анализ передаваемых данных, является глубокий анализ пакетов (*DPI*). К достоинствам этого способа можно отнести высокую точность анализа, к недостаткам — низкую скорость, что критично при современных объёмах передаваемого трафика.

В данной работе предлагается быстрый способ анализа трафика, основанный на количественных характеристиках потока транспортного уровня. Этот способ намного быстрее, чем *DPI*, поскольку при его осуществлении не требуется смотреть и записывать содержимое *IP*-пакетов, достаточно запоминать лишь некоторые характеристики потока[1].

Целью работы ставится разработка способа разделения сетевого трафика на классы обслуживания с целью повышения качества обслуживания интернет-провайдера.

При разработке классов обслуживания за основу было взято разделение, предложенное телекоммуникационным стандартизирующим сектором Международного Союза Электросвязи (*ITU-T*), имеющее название *Y.1541*[2]. Это разделение содержит 6 классов трафика, каждый из которых в разной степени требователен к одной из четырёх характеристик:

- Задержка (верхняя граница для среднего значения задержки).
- Джиттер (верхняя граница задержки для квантиля 99.9%).
- Потери (верхняя граница на вероятность потери пакета).
- Ошибки (верхняя граница на вероятность ошибки в пакете).

Так как требования к потерям и ошибкам одинаковы почти у всех классов (они отсутствуют лишь у шестого, «неопределённого» класса), на основе предложенных *ITU* характеристик были выделены следующие 3 типа сетевых приложений:

- Тип 0: не чувствительные к задержкам и джиттеру. Примеры: браузеры при просмотре веб-страниц (без мультимедиа), *BitTorrent*.
- Тип 1: чувствителен только к задержкам, но не к джиттеру. Пример: браузеры при просмотре потокового мультимедиа.
- Тип 2: чувствительные и к задержкам, и к джиттеру. Примеры: *Skype*, *Viber* при осуществлении аудио- и видеосвязи реального времени.

В данной классификации оказывается, что трафик, генерируемый одним и тем же приложением ОС, может относиться к разным классам обслуживания. В этой связи необходимо уточнить понятие «приложение». В рамках данной работы под приложением понимается владелец открытого сокета. Например, приложение ОС *chrome.exe* может одновременно держать открытым сокет на некотором *TCP*-порту и принимать через этот порт гипертекстовую разметку веб-страницы (*HTML*), а также держать открытым сокет на некотором *UDP*-порту и принимать через этот сокет видеоданные (с такого сайта, как *YouTube*). В этой ситуации будем считать, что работают два отдельных приложения: одно типа 0 (гипертекст), другое типа 1 (мультимедиа).

Для проведения анализа отобраны следующие виды трафика:

- Тип 0: *DNS*, *BitTorrent*, *HTML* поверх *HTTP(S)*.
- Тип 1: видео и аудио поверх *HTTP(S)* и *QUIC*.

- Тип 2: видео и аудио в ПО *Skype*.

Были выбраны именно эти виды трафика по той причине, что именно такой трафик генерируют большинство пользователей сети Интернет.

Было собрано более 3 Гб трафика приведённых выше видов. Для выделения этих видов среди прочего трафика использована библиотека *nDPI*, осуществляющая глубокий анализ пакетов. Однако *nDPI* определяет лишь протокол прикладного уровня, а этого оказывается недостаточно для разделения трафика на необходимые классы, так что поверх *DPI* был проведён дополнительный анализ трафика каждого протокола. Итоговое распределение потоков по классам представлено в таблице 1.

Следующий шаг – выделение количественных характеристик потока и применение алгоритмов машинного обучения для создания классификатора.

При выделении количественных характеристик потока следует пояснить некоторые используемые понятия:

- Порция данных – объём полезной нагрузки, переданный от одной стороны к другой, не прерываемый полезной нагрузкой с противоположной стороны;
- Инициатор – инициатор соединения;
- Адресат – адресат соединения.

Таблица 1

### Критерии отбора трафика для обучающей выборки

Тип трафика	Тип приложения	Критерий разделения
<i>DNS</i>	0	-
<i>BitTorrent</i>	0	-
<i>HTTP.text</i>	0	Анализ <i>HTTP</i> -заголовка, разделение по <i>MIME</i> -типу
<i>HTTP.image</i>	0	
<i>HTTP.audio</i>	1	
<i>HTTP.video</i>	1	
<i>SSL.multimedia</i>	1	Эвристическое разделение: мультимедиа > 1 Мб полезной нагрузки
<i>SSL.other</i>	0	
<i>QUIC.multimedia</i>	1	
<i>QUIC.other</i>	0	
<i>Skype.realtime</i>	2	Эвристическое разделение: <i>realtime</i> > 50 пакетов в секунду и > 3 секунд длительности
<i>Skype.other</i>	0	

Полный список выделенных характеристик представлен в таблице 2.

Таблица 2

### Полный список выделяемых характеристик потока

Характеристика	Пояснение
Средний размер сегмента со стороны инициатора	Среднее значение – статистическая оценка математического ожидания
Средний размер сегмента со стороны адресата	
Средний размер порции данных со стороны адресата	
Средний размер порции данных со стороны инициатора	
СКО размера порции данных со стороны инициатора	Среднеквадратическое отклонение – статистическая оценка дисперсии
СКО размера порции данных со стороны адресата	
СКО размера сегмента со стороны инициатора	
СКО размера сегмента со стороны адресата	
Среднее число сегментов на порцию данных со стороны инициатора	Общее число пакетов делённое на общее число порций данных
Среднее число сегментов на порцию данных со стороны адресата	
КПД инициатора	Общее количество полезной нагрузки делённое на общее количество байт
КПД адресата	
Соотношение переданных байт	Во сколько раз адресат передаёт больше, чем инициатор
Соотношение переданной полезной нагрузки	
Соотношение переданных сегментов	
Размеры первых 4 порций данных	2 со стороны инициатора, 2 со стороны адресата
Размеры первых 4 пакетов данных	Не учитывая <i>TCP handshake</i>

Значения характеристик определялись по первой 1000 пакетов каждого потока.

Было испробовано несколько алгоритмов машинного обучения, однако лучший результат по Ф-мере показал алгоритм «Случайный Лес». Для обучения классификатора было использовано две трети выборки, для проверки его производительности — оставшаяся треть. Результаты работы алгоритма представлены в таблице 3.

Таблица 3

### Результат обучения классификатора

Реальные классы	Предсказанные классы			
	0	1	2	
0	1624	1	0	
1	0	36	0	
2	0	0	3	

Кроме того, была предпринята попытка обучить классификатор определять не только тип приложения, но и само приложение, генерирующее трафик. Результаты представлены в таблице 4.

Таблица 4

### Результат расширенного обучения классификатора

Предсказанные классы	Предсказанные классы											
	<i>BitTorrent</i>	<i>DNS</i>	<i>HTTP.audio</i>	<i>HTTP.image</i>	<i>HTTP.text</i>	<i>HTTP.video</i>	<i>QUIC.multimedia</i>	<i>QUIC.other</i>	<i>SSL.multimedia</i>	<i>SSL.other</i>	<i>Skype.other</i>	<i>Skype.realtime</i>
<i>BitTorrent</i>	20											
<i>DNS</i>		1019										
<i>HTTP.audio</i>			6			2						
<i>HTTP.image</i>				139	1							
<i>HTTP.text</i>				4	74					3		
<i>HTTP.video</i>						2						
<i>QUIC.multimedia</i>							1					
<i>QUIC.other</i>								38				
<i>SSL.multimedia</i>			3			1			23	1		
<i>SSL.other</i>				1						239		
<i>Skype.other</i>											88	
<i>Skype.realtime</i>												3

В результате эксперимента была доказана принципиальная возможность определения класса сетевого приложения на основе количественных характеристик потока транспортного уровня. Более того, было показано, что с некоторыми погрешностями возможно даже определять протокол 7-го уровня без анализа содержимого пакетов.

Дальнейшее развитие предложенных идей возможно в нескольких направлениях. Во-первых, следует усовершенствовать методы отделения целевого трафика, чтобы устранить эвристические критерии. Во-вторых, представленные идеи требуется проверить на большем объеме трафика, ведь собранные 3 Гб не могут претендовать на полную репрезентативность.

### Литература

1. *Yeon-sup Lim, Hyun-chul Kim, Jiwoong Jeong, Chong-kwon Kim, Ted "Taekyoung" Kwon, Yanghee Choi*, Internet Traffic Classification Demystified: On the Sources of the Discriminative Power. New York: ACM, 2010. 12 с.
2. Telecommunication standardization sector of ITU, Network performance objectives for IP-based services. [Электронное издание]: ITU, 2012. 66 с.
3. *Малиновский С.Т., Беленькая М.Н., Спиридонов А.А.* Метод повышения производительности транспортного протокола TCP в глобальных корпоративных сетях передачи данных // Т-Comm: Телекоммуникации и транспорт. 2010. Т. 4. № 7. С. 39-42.

# ИССЛЕДОВАНИЕ СРЕДСТВ МАРШРУТИЗАЦИИ ПРИ ИНТЕГРАЦИИ ИНФОРМАЦИОННЫХ СИСТЕМ

Чельшиков С.А.,  
МТУСИ, студент группы М091501(72)  
Гузев А.В.,  
МТУСИ, к.т.н., доцент

Информационная система компании может насчитывать несколько десятков приложений, автоматизирующих различные бизнес-процессы и выполняющихся под управлением различных операционных систем. При этом выполнение одного бизнес-процесса может быть распределено между несколькими такими приложениями. В этом случае возникает необходимость в их интеграции между собой. Интеграция между двумя приложениями является простой задачей. Проблемы начинаются, когда в интеграции участвуют три или более приложений. В таком случае приложения должны координировать свои действия сами и точно знать, какую процедуру в рамках бизнес-процесса необходимо выполнить. В таком решении приложения сильно связаны друг с другом, что усложнит изменение самого бизнес-процесса. Кроме того, замена одного приложения на другое, сходное по функционалу, потребует внесения изменений в остальные приложения, участвующие в интеграции.

Одним из решений, призванных уменьшить связность между приложениями, является использование системы обмена сообщениями (СОС), как посредника между участниками интеграции. Она берет на себя роль по координированию участников интеграции, следя за последовательностью выполнения бизнес-процесса.

Интеграция с использованием системы обмена сообщениями основана на методе обмена сообщениями, при котором участники интеграции обмениваются между собой небольшими порциями данных (сообщениями). Сообщения состоят из заголовка, используемого СОС, и тела, содержащего полезные данные. Приложения не посылают сообщения непосредственно друг другу, а помещают его в специальные логические маршруты (каналы), которые используются СОС для определения получателя сообщения. Иногда бизнес-логика интеграции может быть настолько сложной, что использование одних только каналов становится недостаточным. В таких случаях используются промежуточные модули, которые будут управлять потоком сообщений, пересылая их нужным получателям.

Такое управление можно осуществить несколькими способами:

1. С использованием канала публикация-подписка и фильтров сообщений
2. С использованием маршрутизатора

В первом случае перед каждым приложением ставится фильтр, который отбрасывает ненужные сообщения (рис. 1).

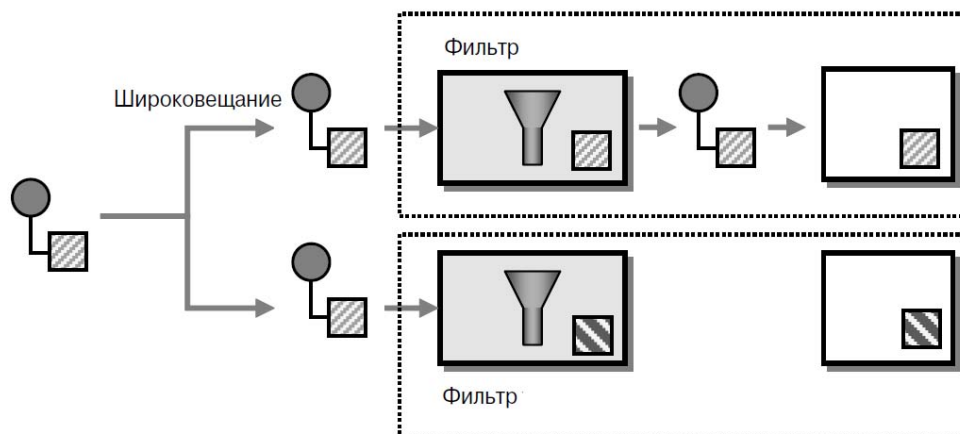
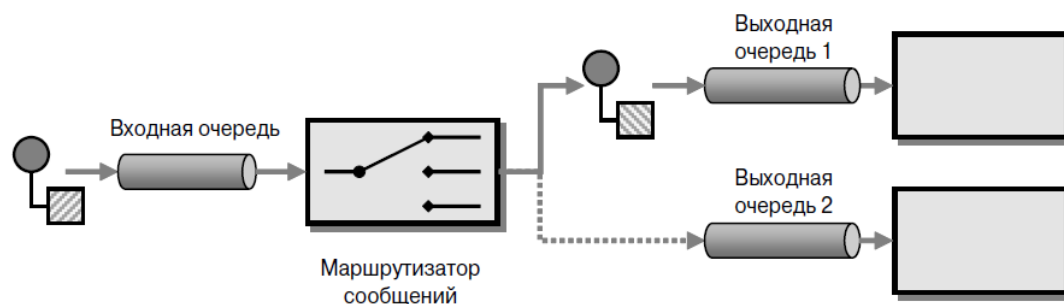


Рис. 1. Использование каналов и фильтров для управления потоком сообщений

При такой реализации возможно использование нескольких каналов публикация-подписка, что позволяет распределить нагрузку между всеми участниками интеграции. В то же время распределенная архитектура такой интеграции затрудняет контроль, администрирование и тестирование.

Во втором случае используется специальный компонент – маршрутизатор на основе содержимого, который анализирует сообщения и помещает их в соответствующий канал (рис. 2).



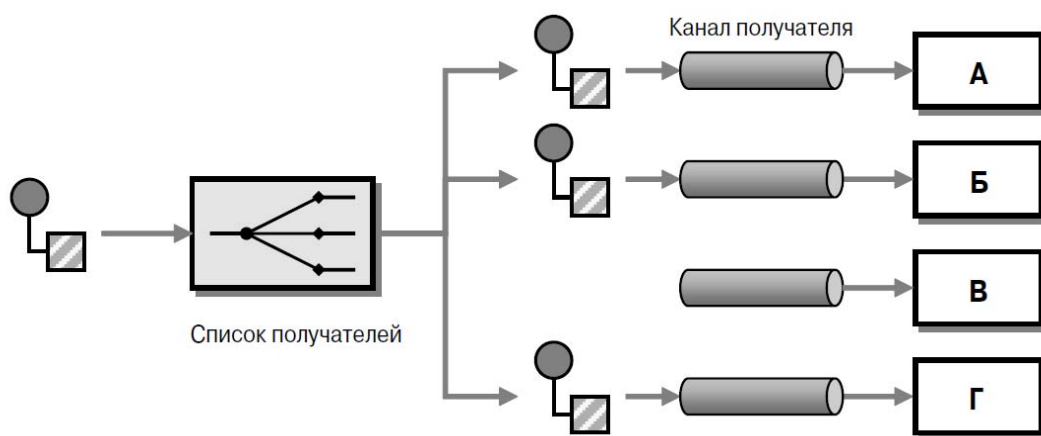
**Рис. 2.** Использование маршрутизатора на основе содержимого для управления потоком сообщений

Маршрутизатор на основе содержимого реализует предиктивную маршрутизацию, т.е. обладает сведениями о возможностях всех остальных систем, таким образом, усиливая связь между собой и интегрируемыми приложениями. Также на каждого получателя сообщения необходимо создать отдельный канал, что влечет за собой дополнительные расходы.

В отличие от использования канала публикация-подписка и фильтров маршрутизатор не позволяет приложениям решать, какие сообщения им необходимы и не позволяет рассылать сообщения нескольким получателям.

Архитектурные шаблоны «Список получателей» и «Динамический маршрутизатор» решают эти проблемы.

Список получателей – шаблон интеграции на основе маршрутизатора, позволяющий устраивать рассылку (Рисунок 3). Список получателей может быть указан в заголовке сообщения, но чаще всего он установлен в соответствии с правилами в маршрутизаторе. Как и канал, публикация-подписка список получателей посылает сообщения нескольким потребителям.



**Рис. 3.** Шаблон интеграции «Список получателей»

Динамический маршрутизатор использует дополнительный управляющий канал, с помощью которого определяются правила маршрутизации (рис. 4).

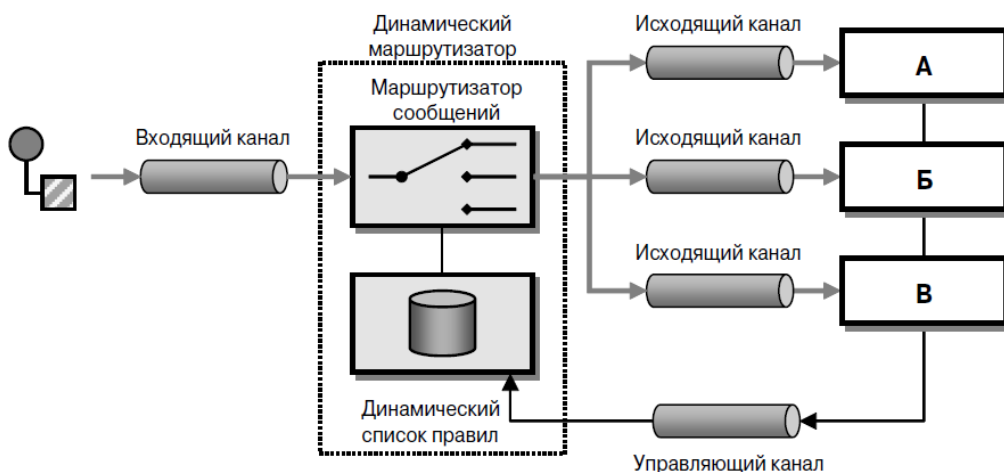


Рис. 4. Шаблон интеграции «Динамический маршрутизатор»

Во время запуска системы каждый потенциальный получатель отправляет динамическому маршрутизатору по этому каналу специальное сообщение, в котором объявляет о своем присутствии и перечисляет условия, при которых может обрабатывать сообщения. Динамический маршрутизатор сохраняет настройки каждого получателя в базе правил. При поступлении нового сообщения динамический маршрутизатор просматривает базу правил и отправляет сообщение тому получателю, правила которого оно удовлетворяет. Это позволяет эффективно реализовать предиктивную маршрутизацию без необходимости ставить динамический маршрутизатор в зависимость от каждого потенциального получателя.

Главными недостатками динамического маршрутизатора являются сложная архитектура решения, а также трудности, связанные с отладкой.

Таким образом, были рассмотрены несколько интеграционных шаблонов для осуществления маршрутизации сообщений. Использование фильтров и каналов является самым простым способом управления потоком сообщений, но усложняет контроль, тестирование и отладку за счет распределенной архитектуры. Использование маршрутизатора на основе содержимого наоборот обеспечивает хороший контроль и тестирование, но увеличивает связность между маршрутизатором и интегрируемыми приложениями. Архитектура на основе шаблона «Динамический маршрутизатор» решает проблемы связности при использовании маршрутизации, предоставляя интерфейс для интегрируемых приложений, через которые они могут влиять на бизнес-процесс, сохраняя при этом централизованный контроль над его прохождением.

## Литература

1. Шаблоны интеграции корпоративных приложений. М. Фаулер, 2007.
2. SOA patterns / Arnon Rotem-Gal-Oz, 2012.

# ИССЛЕДОВАНИЕ МЕТОДОВ И СРЕДСТВ СИСТЕМ ПЕРЕДАЧИ МУЛЬТИМЕДИЙНЫХ ДАННЫХ

Трофлянин В.В.,  
МТУСИ, студент группы М091501(72)  
Беленькая М.Н.,  
МТУСИ, доцент

Согласно докладу *Cisco Global Cloud Index* от октября 2015 года мировой интернет-трафик увеличился в 1,5 раза по сравнению с 2014 годом. А основной вклад в такое резкое увеличение внесли мультимедийные данные (данные, включающие в себя более одного средства коммуникации), которые в 2015 году составили 67% мирового интернет-трафика. Для того, чтобы обеспечивать возросшие потребности пользователей, операторы связи вынуждены постоянно наращивать и расширять имеющиеся сети передачи данных. Однако, постоянное наращивание мощностей привело к тому, что возникла проблема централизованного управления и поддержки сетевого оборудования, которая не решается в традиционных сетях передачи данных. Также в больших сетях становится трудно разрешимой проблема балансировки нагрузки между сетевыми устройствами.

Еще одной важной деталью является то, что канал передачи данных – это услуга, продаваемая операторами связи, и если общая пропускная способность каналов оператора может быть высокой (в современных сетях – до нескольких сотен гигабит в секунду), то каналы, продаваемые клиентам, имеют куда меньшую пропускную способность. Здесь необходимо четкое понимание, что в сфере телекоммуникаций товаром является именно пропускная способность, или же «скорость». Оператор связи должен гарантированно предоставлять своим клиентам именно ту пропускную способность, которая указана в *SLA* (соглашении об уровне предоставления услуги).

Однако, увеличение объема мультимедийного трафика, передаваемого по сетям, создает для оператора связи определенные трудности, которые объясняются следующими особенностями таких данных:

- С каждым годом файлы мультимедиа становятся всё больше (1 минута *UHD*-видео, сжатого с помощью кодека *MPEG-4*, занимает 375 МБ);
- Необходимы высоко эффективные алгоритмы сжатия данных;
- Мультимедийные данные чувствительны к временным задержкам.

Исходя из выше перечисленных особенностей можно выделить следующие параметры сети, которые необходимо контролировать оператору связи:

- 1) Пропускная способность (англ. *bandwidth*) – мера, позволяющая определить способность каждого элемента сети передавать информацию. Она ограничена наименьшей пропускной способностью, которую поддерживает каждый линк между двумя оконечными точками.
- 2) Задержка – это время, затрачиваемое на передачу пакета из одной точки в другую.
- 3) Джиттер – дисперсия задержки или степень ее разброса при передаче потока пакетов.
- 4) Потери (англ. *loss*) – представляет собой процент пакетов, которые не были доставлены. Причиной потерь могут стать ошибки в процессе передачи пакетов или их отбрасывание при возникновении перегрузок на канале связи.

Для поддержания данных параметров и эффективного использования ресурсов сети операторы связи применяют различные средства. Наиболее известным средством является качество обслуживания (*Quality of Service*) – технология, позволяющая сети обеспечить необходимый сервис заданному трафику в определенных технологических рамках. Суть *QoS* заключается в классификации трафика, в основе которого лежит анализ различных параметров данных, формирований очередей и правил обработки согласно произведенной классификации.

Существует три модели реализации *QoS*:

- 1) *Best Effort* - сервис с максимальными усилиями. Реализуется в сети, когда делается все возможное для доставки пакета, но при этом ничего не гарантируется. Между пакетами отдельных



приложений и пользователей не делается никаких различий и обслуживаются они по принципу *FIFO* (англ. *First In First Out*).

2) *Soft QoS* – негарантированная доставка. При использовании этого вида *QoS* происходит некая приоритезация трафика, на основании которой часть трафика получает преимущество при обработке над другой его частью. При этом более приоритетному трафику предоставляется большая пропускная способность и происходит меньшее количество потерь данных. Этот вид не гарантирует обеспечения каких-либо определенных значений параметров *QoS* и представляет собой лишь статическое предпочтение.

3) *Hard QoS* – гарантированная доставка. Этот вид обеспечивает численные гарантии значений различных параметров *QoS* при передаче различного трафика. Значения этих параметров остаются неизменными ни при каких обстоятельствах даже при возникновении сильных перегрузок в сети.

Вышеперечисленные виды *QoS* могут использоваться как отдельно, так и в дополнение друг к другу, что позволяет учесть разнообразные требования приложений и различные условия работы сети.

Основываясь на рекомендациях компании *Cisco*, *RFC 3246*, а также стандартах *ITU*, мной были сформированы требования к *QoS* для различных типов мультимедийных данных, которые представлены в таблице 1.

Таблица 1

#### Требования к *QoS* для различных мультимедийных данных

Характеристика	<i>VoIP</i>	<i>Live streaming</i>	<i>On-demand streaming</i>
Класс <i>DSCP</i>	<i>EF</i>	<i>AF41</i>	<i>CS4</i>
Задержка, мс	≤ 150	≤ 150	2000-4000
Потери, %	≤ 1	≤ 1	≤ 2
Джиттер, мс	≤ 10	≤ 30	-
Минимальная пропускная способность, Мбит/с	0,106-0,150	2	1,5

Другой возможностью оператора связи по увеличению эффективности использования ресурсов сети в процессе передачи мультимедийных данных является переход к программно-конфигурируемой сети (*SDN* от англ. *Software-defined Networking*) – сеть передачи данных, в которой уровень управления сетью отделён от устройств передачи данных и реализуется программно. Данная архитектура позволяет реализовать автоматическое конфигурирование сетей под определенные приложения, или набор приложений, дает возможность приложениям запрашивать сетевые услуги и манипулировать ими, а с другой стороны, предоставляет приложениям топологию и состояние сети. Цели *SDN* включают способность быстро внедрять сетевые инновации, быстрее и радикально упростить и автоматизировать процесс управления крупными сетями.

Архитектура *SDN* представлена на рисунке 1. Кратко рассмотрим её основные компоненты:

1) Уровень приложений – реализует интерфейсы оптимизации сети под конкретное бизнес приложение (к примеру, *Microsoft Lync*) и его основная роль – изменение сети в реальном времени под текущие нужды обслуживаемой программы.

2) Уровень управления – реализуется центральным устройством (контроллер *SDN*), который выполняет функции управления устройствами сети и централизованного управления трафиком. Управление устройствами сети (коммутаторы *OpenFlow*) производится по протоколу *OpenFlow*. Контроллер *SDN* управляет таблицами потоков коммутаторов, на основании которых принимается решение о передаче принятого пакета на конкретный порт коммутатора.

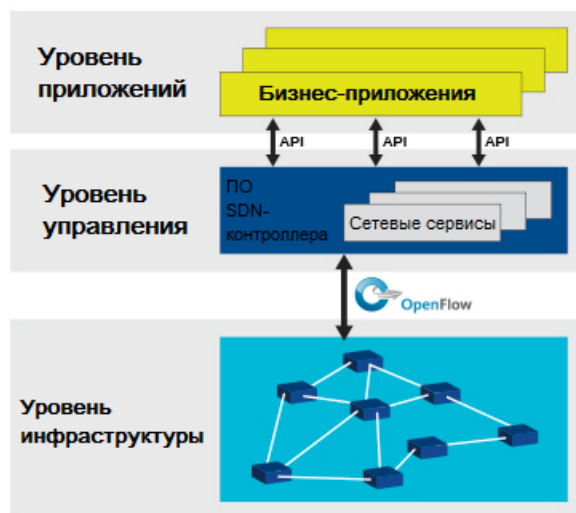


Рис 1. Архитектура SDN

3) Уровень инфраструктуры – это специализированные *OpenFlow*-коммутаторы, которые не имеют собственных алгоритмов маршрутизации и реализуют только функцию передачи данных по таблицам потоков, которыми управляет контроллер *SDN*.

Преимущества *SDN* по сравнению с классическими сетями передачи данных:

- централизованное управление качеством передачи данных, маршрутизацией;
- возможность быстрого реконфигурирования сети;
- удобный механизм абстракции и виртуализации сетевых ресурсов и сервисов;
- повышение эффективности распределения ресурсов и размещения сервисов в сети.

Следующая технология, увеличивающая эффективность использования сетевых ресурсов и предоставления услуг - виртуализация сетевых функций *NFV* (англ. *Network Functions Virtualization*). Суть данной технологии заключается в виртуализации физических сетевых элементов телекоммуникационной сети, когда сетевые функции исполняются программными модулями, работающие на стандартных серверах и виртуальных машинах. Эти программные модули могут взаимодействовать между собой для предоставления услуг связи, чем ранее занимались аппаратные платформы.

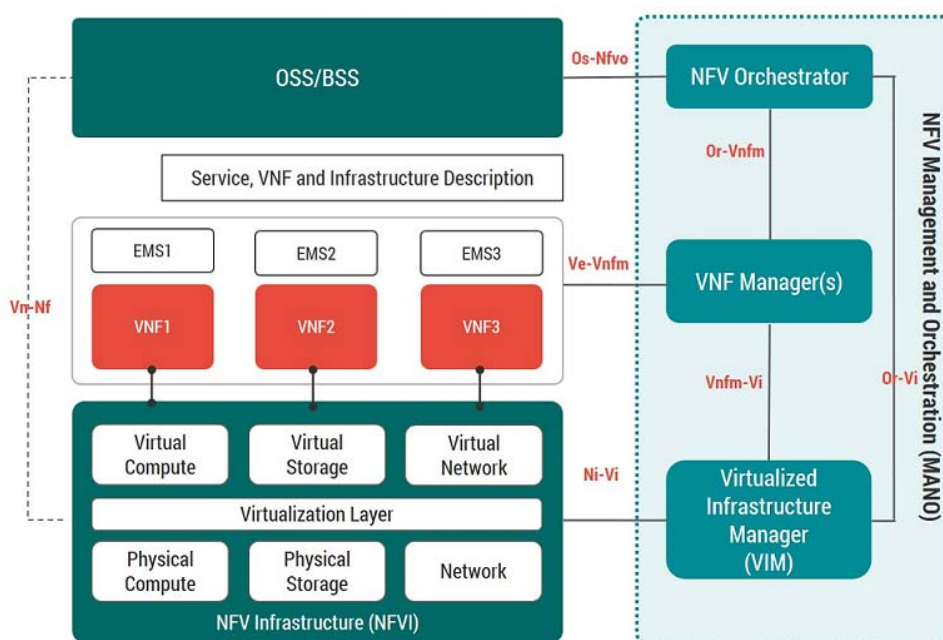


Рис. 2. Архитектура NFV

На данный момент, технология *NFV* только начинает свой путь развития. Компанией *ETSI* предложена первая архитектура *NFV* - Management and Orchestration (*MANO*). Она включает концептуальные блоки, при этом их реализация остается на откуп поставщикам решений. Интерфейсы взаимодействия между блоками пока не стандартизированы. Архитектура виртуализации сетевых функций (*NFV*), разработанная *ETSI* показана на рис. 2.

Различные телекоммуникационные компании уже преступили к внедрению технологий *SDN/NFV* в своих сетях. Ниже представлены основные из них: *Vodafone*, *AT&T*, *Telefonica*, *Deutsche Telekom*, *NTT*, Ростелеком.

Исходя из списка, указанного выше можно сделать вывод, что основные игроки мирового телекоммуникационного рынка оценили преимущества новых технологий и приступили к их внедрению. Переход операторов связи к технологии *SDN/NFV*, а также оптимизация алгоритмов *QoS* позволят оперативно выводить новые мультимедийные услуги на рынок, обеспечить качество предоставления таких услуг, а также сократить капитальные затраты и повысить эффективность использования ресурсов.

## Литература

1. Беленькая М.Н., Малиновский С.Т., Яковенко Н.В., Администрирование в Информационных системах. Учебное пособие для вузов. – М.: Горячая линия – Телеком, 2011. 400 с.
2. Silberschatz, Galvin, Operating System Concepts. Yale University. John Wiley & Sons, Inc. 2014.
3. ONF White Paper, Software-Defined Networking: The New Norm for Networks, April 13, 2012.

# СИНТЕЗ ОПТИМАЛЬНОГО ФИЛЬТРОВОГО ДЕМОДУЛЯТОРА ДЛЯ ЦИФРОВОЙ СИСТЕМЫ СВЯЗИ С ПОСИМВОЛЬНОЙ ПЕРЕДАЧЕЙ

Арбузов С.В., Стогова К.В.,  
 МТУСИ, студенты группы БЗС1301  
 Волчков В.П.,  
 МТУСИ, д.т.н., профессор

**Постановка задачи и ее актуальность.** В цифровых системах связи с посимвольной передачей данных передаваемый в канал низкочастотный эквивалент сигнала  $\dot{s}(t)$  (комплексная огибающая) представляет собой последовательность из  $n$  одинаковых по форме, но разнесенных по времени модулированных импульсных посылок

$$\dot{s}(t) = \sum_{k=0}^{n-1} \dot{u}_k g(t - kT), \quad t \in \mathbb{R} = (-\infty, \infty), \quad (1)$$

где  $g(t)$  – вещественная базовая импульсная посылка (базисный импульс);  $\dot{u}_k$  – комплексный модулирующий символ из алфавита QAM или PSK [1,2].  $T$  – временная задержка между соседними импульсами, называемая символьным интервалом. При этом важной задачей является выбор наиболее подходящего сигнального базиса  $\mathcal{B} = \{g(t - kT)\}_{k=0}^{n-1}$ , который должен быть ортогональным, обладать хорошей частотно-временной локализацией и создавать предпосылки для относительно простой реализации оптимального алгоритма приема. Известно [2], что этим условиям удовлетворяет базис  $\mathcal{B}$ , состоящий из  $n$  сдвинутых по времени импульсов  $g(t)$  с корреляционной функцией (рис. 1)

$$B_g(\tau) = \int_{-\infty}^{\infty} g(t)g(t+\tau)dt = \begin{cases} \frac{\pi}{4} \operatorname{sinc}(1/2\beta), & \tau = \pm 1/2\beta \\ \operatorname{sinc}(\tau/T) \frac{\cos(\pi\beta\tau/T)}{1 - (2\pi\beta\tau/T)^2}, & \text{иначе} \end{cases} \quad (2)$$

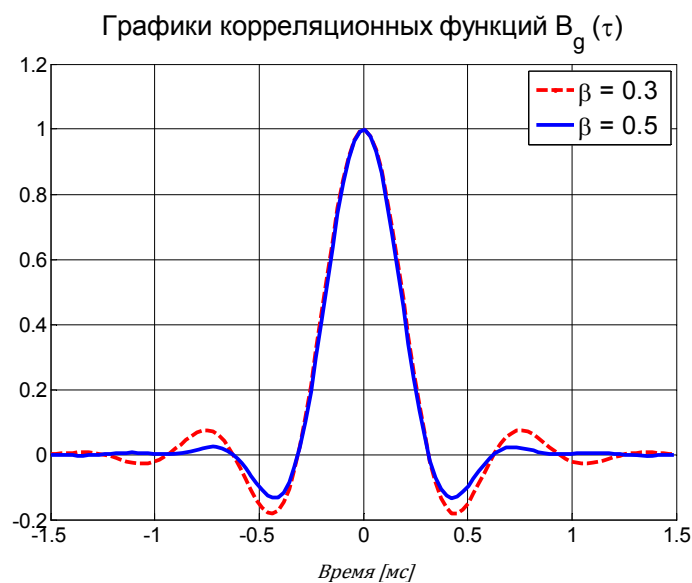


Рис. 1. Графики корреляционных функций для разных  $\beta$  удовлетворяющей критерию Найквиста

$$B_g(iT) = \int_{-\infty}^{\infty} g(t)g(t+iT)dt = \begin{cases} 1, & i=0 \\ 0, & i \neq 0 \end{cases} \quad (3)$$

и спектральной плотностью мощности (СПМ)

$$G_g(f) = \mathcal{F}(B_g(\tau)) = \begin{cases} T, & |f| \leq \frac{1-\beta}{2T}, \\ \frac{T}{2} \left[ 1 + \cos \left( \frac{\pi T}{\beta} \left( |f| - \frac{1-\beta}{2T} \right) \right) \right], & \frac{1-\beta}{2T} \leq |f| \leq \frac{1+\beta}{2T}, \\ 0, & |f| \geq \frac{1+\beta}{2T}. \end{cases} \quad (4)$$

принадлежащей семейству функций  $\{G_g(f; \beta), \beta \in [0, 1]\}$  приподнятого косинуса (raised cosine function), зависящему от параметра  $\beta \in [0, 1]$  – коэффициента сглаживания (rolloff factor) [1,2] (здесь и далее  $\mathcal{F}(\cdot)$ ,  $\mathcal{F}^{-1}(\cdot)$  – прямое и обратное преобразования Фурье). Нетрудно убедиться, что такие базисные импульсы описываются выражением  $g(t) = \mathcal{F}^{-1}(G_g(f)^{1/2})$ , а для их формирования на передающей стороне может быть использован БИХ фильтр с коэффициентом передачи  $H_g(f) = \sqrt{G_g(f)}e^{-j2\pi f t_0}$  и импульсной характеристикой (ИХ)  $h(t) = g(t - t_0)$ , где  $t_0$  – некоторая номинальная задержка, которая должна обеспечивать физическую реализуемость фильтра. Пусть на приемной стороне (выходе канала) стоит согласованный БИХ фильтр с комплексно-сопряженным коэффициентом передачи  $H_g^*(f) = \sqrt{G_g(f)}e^{j2\pi f t_0}$ , которому соответствует ИХ  $h_{c\phi}(t) = g(t_0 - t)$ . Тогда, сквозная частотная характеристика приемно-передающего тракта (без учета шумов и искажений в канале)  $H_o(f) = H_g(f)H_g^*(f) = G_g(f)$  совпадает с (4) и расщепляется поровну между формирующим фильтром передатчика  $H_g(f)$  и согласованным фильтром (СФ) приемника  $H_g^*(f)$ . Покажем, что в этом случае на выходе  $y(t)$  СФ приемника в моменты времени кратные  $T$  происходит восстановление передаваемой информации  $\dot{u}_k$ . Действительно, спектр выходного сигнала равен  $Y(f) = \mathcal{F}(y(t)) = H_g^*(f)S(f) = H_g^*(f) \sum_{k=0}^{n-1} \dot{u}_k H_g(f) e^{-j2\pi f T k} = \sum_{k=0}^{n-1} \dot{u}_k G_g(f) e^{-j2\pi f T k}$ , поэтому  $y(t) = \mathcal{F}^{-1}(Y(f)) = \sum_{k=0}^{n-1} \dot{u}_k \mathcal{F}^{-1}(G_g(f) e^{-j2\pi f T k}) = \sum_{k=0}^{n-1} \dot{u}_k B_g(t - Tk)$ , а значит, согласно (3),  $y(lT) = \sum_{k=0}^{n-1} \dot{u}_k B_g(T(l-k)) = \begin{cases} \dot{u}_k, & l=k \\ 0, & l \neq k \end{cases}$ .

Таким образом, БИХ фильтр с частотной характеристикой  $H_g^*(f)$  и ключом, замыкающимся в моменты времени  $t = Tk$ , является оптимальным фильтровым демодулятором (ФД), а задача его синтеза сводится к построению фильтра, у которого корреляционная функция импульсной характеристики (ИХ)  $B_h(\tau) = \mathcal{F}^{-1}(H_o(f)) = B_g(\tau)$  совпадает с выражением (2). Структурная схема соответствующего приемно-передающего тракта для гауссовского канала связи с аддитивным белым гауссовским шумом  $w(t)$  представлена на рис.2, где  $D(\cdot)$  – функция Дирака. Строгое решение данной задачи невозможно, поскольку системная функция такого фильтра не представляется в виде дробно-рациональной функции. Поэтому возможно лишь приближенное решение, основанное на синтезе цифрового БИХ фильтра, наилучшим образом аппроксимирующим дискретный вариант ИХ  $h[i] = h(i\Delta)$ ,  $i \in \mathbb{Z} \triangleq \{0, \pm 1, \pm 2, \dots\}$ , или его спектр  $P(f) = \mathcal{F}_\Delta(h[i])$ , где  $\Delta$  – интервал дискретизации  $\mathcal{F}_\Delta(\cdot)$  – дискретное преобразование Фурье. Известно [3] множество методов субоптимального синтеза, основанных на указанных критериях качества, однако они не дают однозначного ответа какой из них является наилучшим с точки зрения выполнения критерия Найквиста (3) и простоты реализации фильтра. Ведь именно выполнение критерия Найквиста обеспечивает ортогональность сигнального базиса, а значит и отсутствие межсимвольной интерференции (МСИ) на приемной стороне.

Чем точнее он выполняется, тем меньше остаточная МСИ, обусловленная практической реализацией элементов приемно-передающего тракта цифровой системы. Поэтому задача синтеза оптимального фильтрового демодулятора для цифрового сигнала (1), при которой обеспечивается минимальная остаточная МСИ, является актуальной. В докладе для решения этой задачи предлагаются два новых критерия, ориентированных на минимизацию остаточной МСИ, описывается метод синтеза цифрового БИХ фильтра, входящего в состав оптимального фильтрового демодулятора, приводятся результаты соответствующего моделирования

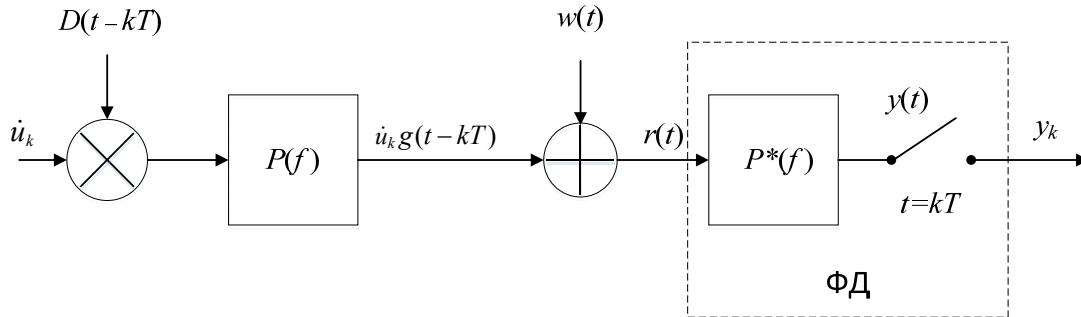


Рис. 2. Схема приёмно-передающего тракта для гауссовского канала связи

**Метод решения.** Для разработки цифрового БИХ фильтра примем за основу наиболее простую структуру, описываемую разностным уравнением  $N$ -го порядка

$$y_i = \sum_{m=1}^N a_m y_{i-m} + x_i, \quad i = 0, 1, 2, \dots \quad (5)$$

где  $x_i$ ,  $y_i$  – отсчеты сигнала на входе и выходе фильтра. Для определения коэффициентов  $a_m$  в (5) воспользуемся критерием минимума среднего квадрата ошибки линейного предсказания

$$\tilde{\epsilon}^2 = \| y_i - \sum_{m=1}^N a_m y_{i-m} \|^2 = B_y[0] - 2 \sum_{m=1}^N a_m B_y[m] + \sum_{m=1}^N \sum_{n=1}^N a_n a_m B_y[m-n] \rightarrow \min_{a_1, \dots, a_N} \quad (6)$$

где  $\| y_i \|^2 \triangleq \left( \sum_{i=-\infty}^{\infty} |y_i|^2 \right)^{1/2}$ ,  $B_y[m] \triangleq \sum_{i=-\infty}^{\infty} y[i]y[i+m]^*$  – соответственно, норма и корреляционная функция последовательности  $\{y_i\}$  на выходе фильтра. Дифференцирование  $\tilde{\epsilon}^2$  по параметрам  $a_m$ ,  $m = 1, \dots, N$  и приравнивание полученных производных нулю приводит к системе уравнений

$$\sum_{m=1}^N a_m B_y[m-k] = B_y[k], \quad k = 1, 2, \dots, N \quad (7)$$

Если на вход фильтра (5) подать единичный импульс Кронекера  $x_i = \delta_i$ , то при нулевых начальных условиях уравнение (5) будет описывать желаемую импульсную характеристику фильтра, которую обозначим  $h[i]$ , а система (7) принимает вид

$$\sum_{m=1}^N a_m B_h[m-k] = B_h[k], \quad k = 1, 2, \dots, N, \quad (8)$$

решение которой при  $B_h[l] \triangleq B_g(l\Delta)$   $l = 0, 1, 2, \dots, N$  дает оптимальные значения коэффициентов  $a_m$ . Однако при этом остается произвол в выборе порядка фильтра  $N$ .

Для разработки оптимального фильтрового демодулятора ниже предлагается конструктивный алгоритм выбора оптимального порядка  $N$  фильтра (5), основанный на достижении максимальной точности выполнения критерия Найквиста (3). Воспользуемся эквивалентной записью критерия Найквиста (3) в частотной области [1,2]  $\sum_{k=-\infty}^{\infty} G_g(f+k/T) = T$ ,  $0 \leq f \leq 1/T$ . Тогда, учитывая финитность и четность функции СПМ (4), данный критерий может быть упрощен и представлен в виде  $G_g(f) + G_g(f-1/T) = T$ ,  $0 \leq f \leq f_n$ , где  $f_n = (1-\beta)/2T = f_0/2 = 1/2\Delta$  – частота Найквиста. С другой стороны, известно [3], что СПМ  $W(f) = \mathcal{F}_o(B_h[i])$  импульсной характеристики  $h[i]$  цифрового фильтра (6) в случае  $B_h[l] \triangleq B_g(l\Delta)$  описывается выражением

$$W(f) = T\sigma_N^2 / |1 - a_1 e^{-j2\pi f\Delta} - a_2 e^{-j4\pi f\Delta} - \dots - a_N e^{-j2N\pi f\Delta}|^2, \quad (9)$$

где  $\sigma_N^2 = B_g(0) - \sum_{m=1}^N B_g(m) a_m$  – средний квадрат ошибки предсказания. Поэтому, критерий Найквиста, обеспечивающий ортогональность сдвинутых по времени дискретных импульсов  $\{h[i\Delta - kT], i \in \mathbb{Z}\}_{k=0}^{n-1}$ , принимает вид  $W(f) + W(f - 1/T) = T$ ,  $0 \leq f \leq f_n$ . Строгое выполнение данного критерия для фильтра (5) возможно лишь при  $B_h[l] \triangleq B_g(l\Delta)$ ,  $l \in \mathbb{Z}$ , т.е. когда порядок фильтра  $N = \infty$ . Однако можно определить меру максимальной точности выполнения критерия Найквиста  $\varepsilon(N_o) = \|(W(f) + G_g(f - 1/T) - T)/T\| = \varepsilon_o$ , при которой достигается разумный компромисс между порядком фильтра  $N = N_o$  и среднеквадратической ошибкой (СКО) рассогласования  $\varepsilon(N_o)$ . А именно, величина  $\varepsilon_o$  выбирается таким образом, чтобы дальнейшее увеличение порядка  $N > N_o$  не приводило к сколь-нибудь значительному уменьшению величины  $\varepsilon(N)$ . Аналогично, в качестве меры максимальной точности выполнения критерия Найквиста можно предложить относительную СКО аппроксимации СПМ (4) выражением (9)  $\tilde{\varepsilon}(N_o) = \|(W(f) - G_g(f))/T\| = \varepsilon_o$ , где  $\varepsilon_o$  выбирается из аналогичных соображений. Ниже в работе экспериментально исследовались оба способа нахождения оптимального порядка фильтра.

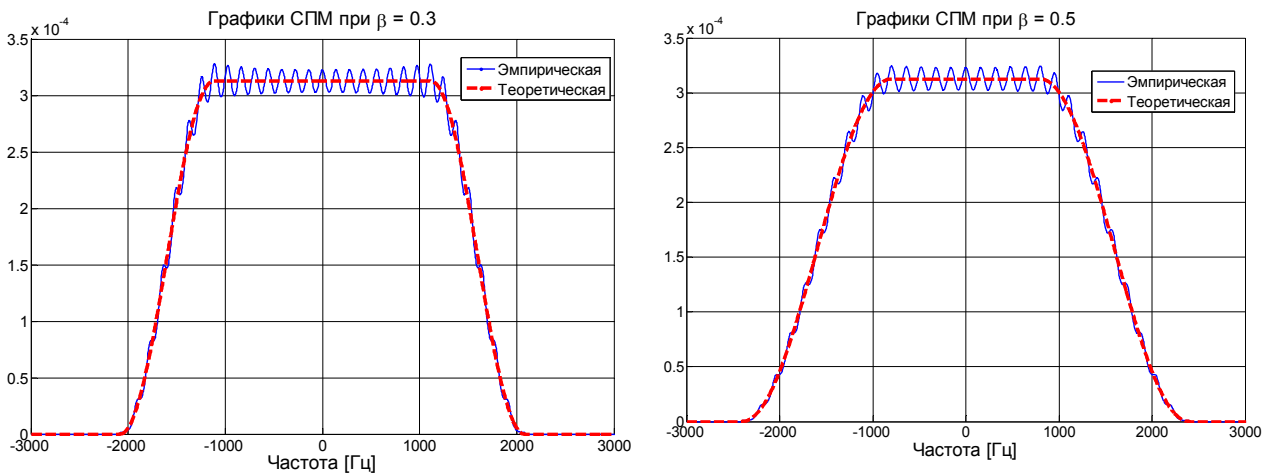


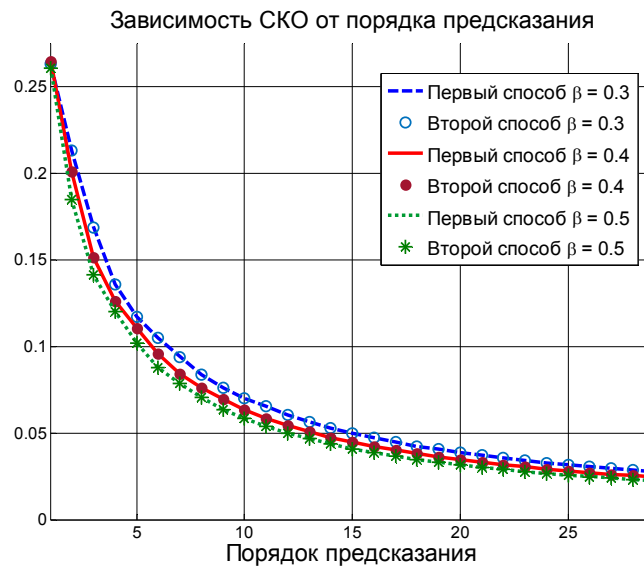
Рис. 3. Графики исходной и эмпирической СПМ для разных  $\beta$

Результаты моделирования. Синтез оптимального цифрового БИХ фильтра осуществлялся с помощью компьютерного моделирования в среде пакета MatLab. Для решения системы уравнений (8) и нахождения параметров фильтра, использовался эффективный вычислительный алгоритм Левинсона-Дарбина, который реализован в виде встроенной MatLab-функции `levinson`. Имея в распоряжении параметры, по формуле (9) для,  $m$ , и разных значений коэффициента и были рассчитаны графики эмпирической СПМ, представленные на рис. 3.

Для сравнения на этом же рисунке приведены аналогичные кривые теоретической СПМ  $G_g(f)$ , построенные по формуле (4). Видно, что спектральная плотность  $W(f)$  хорошо аппроксимирует  $G_g(f)$  при всех  $\beta$ , но наблюдаются небольшие всплески на пологом участке.

Для нахождения оптимального порядка фильтра  $N_o$  используем два предложенных выше способа оценки рассогласования с критерием Найквиста:  $\varepsilon(N_o) = \varepsilon_o$  (первый способ) и  $\tilde{\varepsilon}(N_o) = \varepsilon_o$  (второй способ). С этой целью рассчитывались кривые СКО  $\varepsilon(N)$ ,  $\tilde{\varepsilon}(N)$  для разных порядков фильтра  $N = 1 \div 29$  и разных значений параметра  $\beta$ , которые представлены на рис. 4. Анализ графиков показывает, что полученные двумя способами значения СКО практически совпадают. Это говорит о том, что два метода расчёта СКО эквивалентны, т.е. дают одинаковый результат. Кроме того, нетрудно заметить, что при больших значениях  $N > N_o = 15$  порядка фильтра графики кривых

становятся пологими, а ошибка  $\varepsilon(N)=\tilde{\varepsilon}(N)<\varepsilon_0=0,05$  мала и почти не меняется при увеличении порядка. Следовательно, оптимальный порядок фильтра следует выбрать равным  $N_o = 15$ .



**Рис. 4.** Зависимость СКО от порядка предсказания

### Литература

1. Волчков В.П. Основы построения цифровых систем связи. Учебное пособие. – МГУСИ. М: 2005. – 69 с.
2. Прокис Джон. Цифровая связь. Пер. с англ./Под. ред. Д.Д. Кловского. – М: Радио и связь, 2000. – 800 с.
3. Сергиенко А.Б. Цифровая обработка сигналов: Учебник для вузов. 2-е изд. – СПб: Питер, 2007. – 751.
4. Волчков В.П., Уваров С.С. Аппроксимация узкополосных случайных процессов с помощью комплексной рекуррентной m-модели скользящего окна второго порядка // Т-Сотт: Телекоммуникации и транспорт. 2015. Т. 9. № 3. С. 54-61.



# ИССЛЕДОВАНИЕ ПОМЕХОУСТОЙЧИВОСТИ ЦИФРОВОЙ СИСТЕМЫ СВЯЗИ С КАНАЛЬНЫМ ПРЕКОДЕРОМ И ФИНИТНОЙ ПОСИМВОЛЬНОЙ ПЕРЕДАЧЕЙ

Безруков И.М.,  
 МТУСИ, студент группы БСС1202  
 Волчков В.П.,  
 МТУСИ, д.т.н., профессор

## Введение

В цифровых системах связи оптимальная обработка сигнала на приемной стороне сильно затрудняется из-за наличия межсимвольной интерференции (МСИ). МСИ возникает либо вследствие неидеальных частотных характеристик проводного канала связи, либо из-за частотно-селективных замираний при беспроводном многолучевом распространении. Влияние МСИ ухудшает качество приёма сигнала, что вынуждает снижать скорость передачи данных по каналу. Наиболее известный метод борьбы с МСИ основан на применении эквалайзера, выравнивающего частотную характеристику канала, но его внедрение усложняет схему оптимального приёмника и ограничивает потенциальную помехоустойчивость системы, т.к. эквалайзер может усиливать значительно ослабленные компоненты сигнала. В данной работе рассматривается цифровая система связи с ортогональным временным разделением каналов (OTDM) и QAM модуляцией, для которой предлагается другой вариант борьбы с МСИ – использование на передающей стороне канального прекодирования (предыскажения). Прекодер на основе текущих оценок импульсной характеристики (ИХ) канала, полученных от приемной стороны по обратному каналу, формирует предыскаженные формирующие каналные импульсы, согласованные с параметрами канала. Таким образом, предлагаемый дискретный прекодер является оптимальным по критерию согласования параметров канала связи и используемой системы сигналов. Использование оптимального прекодирования позволяет отказаться от эквалайзера в приёмнике и минимизировать МСИ на выходе канала, не ограничивая потенциальную помехоустойчивость системы связи, за счёт формирования двух ортогональных финитных систем сигналов, согласованных с ИХ канала.

Первая система сигналов  $\mathbf{U}$  используется в квадратурном модуляторе и состоит из сдвинутых по времени предыскаженных финитных формирующих импульсов  $\vec{g}_m$ . Вторая система сигналов  $\mathbf{U}'$  используется в оптимальном приемнике и состоит из сдвинутых по времени финитных опорных импульсов  $\vec{g}'_m$ , хорошо локализованных и симметричных относительно центра символьного интервала. Такие системы сигналов связаны соотношением  $\mathbf{U}' = \mathbf{C}\mathbf{U}$ , где  $\mathbf{C}$  – циркулянтная матрица импульсной характеристики канала.

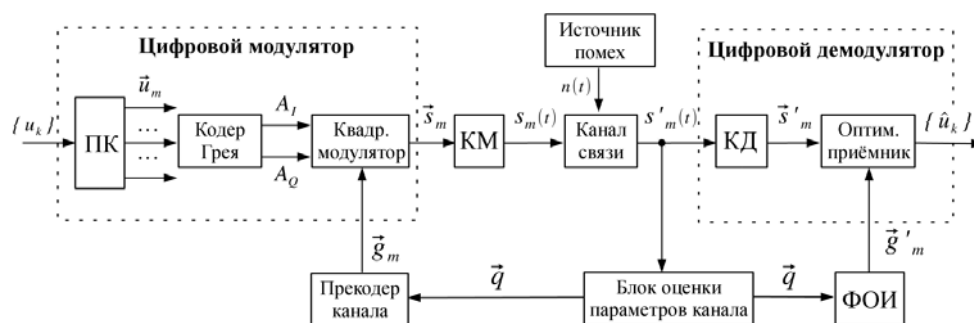


Рис. 1. Структурная схема системы связи с QAM модуляцией и дискретным канальным прекодером

Система связи с OTDM и QAM, представленная на рис. 1, содержит следующие элементы: ПК-преобразователь кода, формирующий бинарные пакеты  $\bar{u}_m$  из входной информационной последовательности  $\{u_k\}$ ; КМ – канальный модулятор; КД – канальный детектор. К схеме связи предлагается добавить следующие элементы:

1. Блок оценки параметров канала, генерирующий вектор оценок импульсной характеристики канала связи  $\bar{q}$ . Оценка импульсной характеристики производится с помощью зондирования канала пилот-сигналом и аппроксимации полученных результатов по методу Прони.

2. Формирователь опорных импульсов (ФОИ), подающий опорные импульсы  $\bar{g}'_m$  на оптимальный приёмник.

3. Прекодер канала, синтезирующий канальные импульсы  $\bar{g}_m$ . Импульсы  $\bar{g}_m$  определяют форму и спектр передаваемых сигналов.

4. Канал обратной связи (служебной связи) между блоком оценки и прекодером, необходимый для передачи вектора оценок импульсной характеристики канала связи  $\bar{q}$ .

### 1. Алгоритм работы канального прекодера

Известно [1], что наилучшую частотно-временную локализацию обеспечивают гауссовские импульсы, однако сформированная на их основе сигнальная система (система Габора) нефинитна и неортогональна. В [2] показано, что можно сформировать финитную сигнальную систему с хорошей частотно-временной локализацией на основе системы Габора посредством канального прекодирования (предыскажения). Алгоритм работы канального прекодера задаётся матричным критерием

$$\min_{\mathbf{U}} \left\{ \|\mathbf{G}_o - \mathbf{C}_o \mathbf{U}\|_E^2 : \mathbf{U} \in M_{m,n} \mid \mathbf{U}^* \mathbf{U} = \mathbf{I}, \mathbf{I} \in M_{n,n} \right\} \quad (1)$$

где  $\|\mathbf{A}\|_E^2 = \langle \mathbf{A}, \mathbf{A} \rangle = \text{tr} \mathbf{A} \mathbf{A}^*$  – евклидова матричная норма (норма Фробениуса), определяемая из скалярного произведения матриц  $\langle \mathbf{A}, \mathbf{B} \rangle = \text{tr} \mathbf{A} \mathbf{B}^*$ ,  $\text{tr} \mathbf{A}$  – след матрицы  $\mathbf{A}$ , символ  $*$  – эрмитово сопряжение;  $M_{m,n}$  – множество всех комплексных матриц размерности  $m \times n$ ;  $\mathbf{G} \in M_{m,n}$  – система Габора, состоящая из сдвинутых по времени гауссовых импульсов;  $\mathbf{U} \in M_{m,n}$  – искомая финитная система сигналов;  $\mathbf{C} \in M_{m,m}$  – циркулянтная матрица, получаемая путём циклического сдвига по столбцам вектора импульсной характеристики  $\bar{q}$  канала связи;  $\mathbf{G}_o = \begin{pmatrix} \mathbf{G} \\ \mathbf{0} \end{pmatrix} \in M_{2m,n}$  – расширенная система Габора;  $\mathbf{C}_o = \begin{pmatrix} \mathbf{C} \\ \mathbf{P} \end{pmatrix} \in M_{2m,m}$  – расширенная матрица импульсных характеристик;  $\mathbf{P} \in M_{m,m}$  – матрица ограничений.

Доказано [2], что критерий (1) будет выполняться, если искомая система сигналов определяется выражением

$$\mathbf{U} = \mathbf{V} \mathbf{I}_o \mathbf{W}^* \quad (2)$$

где  $\mathbf{V} \in M_{m,m}$ ,  $\mathbf{W} \in M_{n,n}$  – унитарные матрицы, получаемые из сингулярного разложения

$$\mathbf{C}_o^* \mathbf{G}_o = \mathbf{V} \mathbf{D} \mathbf{W}^*, \mathbf{I}_o = \begin{pmatrix} \mathbf{I}_{n,n} \\ \mathbf{0}_{m-n,n} \end{pmatrix} \in M_{m,n}, \mathbf{I}_{n,n} - \text{единичная матрица}, \mathbf{0}_{m-n,n} - \text{нулевая матрица}.$$

Таким образом, алгоритм работы прекодера в пошаговой записи принимает вид:

1. Формируется система сигналов Габора  $\mathbf{G}$  из сдвинутых по времени гауссовых импульсов.
2. Формируется матрица импульсных характеристик  $\mathbf{C}$  из вектора оценок импульсной характеристики  $\bar{q}$ . Вектор  $\bar{q}$  является результатом работы блока оценки параметров канала.
3. Формируются расширенные матрицы  $\mathbf{G}_o$  и  $\mathbf{C}_o$ .
4. Вычисляется сингулярное разложение матрицы  $\mathbf{C}_o^* \mathbf{G}_o$ :  $\mathbf{C}_o^* \mathbf{G}_o = \mathbf{V} \mathbf{D} \mathbf{W}^*$
5. Вычисляется финитная сигнальная система  $\mathbf{U}$  по формуле (2).

## 2. Модель системы связи с канальным прекодером

В системе связи, показанной на рис. 1, используется QAM модуляция. Ансамбль сигналов с QAM модуляцией задаётся формулой

$$s_m(t) = g_m(t)A_{kI} \cos(2\pi f_s t) - g_m(t)A_{kQ} \sin(2\pi f_s t), \quad k, l = 1, 2, \dots, \sqrt{M}, \quad (3)$$

где  $M$  – порядок модуляции,  $g_m(t)$  – формирующий импульс канала,  $A_{kI}, A_{kQ}$  – информационные амплитуды. Выражение (2) можно использовать для синтеза оптимальных для заданного канала связи формирующих импульсов.

Модель канала связи представляет собой линейный фильтровый канал (ЛФК)

$$s'_m(t) = q(t) * s_m(t) + n(t) \quad (4)$$

где  $q(t) * s_m(t)$  – свёртка ИХ канала связи и сигнала (3);  $n(t)$  – аддитивный белый гауссовский шум (АБГШ). Модель (4) позволяет, зная ИХ канала связи, представить его как некоторый фильтр, благодаря чему можно моделировать как проводные каналы связи, так и беспроводные каналы с частотно-селективными замираниями. В данной работе используется импульсная характеристика ФНЧ Баттерворта 8-го порядка, задаваемая выражением

$$q_p(n) = q_p \sum_{k=1}^{p/2} c_{p,k} e^{-\alpha_k n T_\Delta} \sin(\beta_k n T_\Delta - \psi_{p,k}), \quad n = 0, \dots, N-1 \quad (5)$$

где  $p$  – порядок фильтра,  $q_p = (\omega_\gamma^{0,5p+1}) / (2^{0,5p-1})$ ,  $\omega_\gamma = 2\pi F_\gamma$  – частота среза в килорadianах,

$T_\Delta$  – период дискретизации,  $N$  – число отсчётов в сигнале;  $\alpha_k = \omega_\gamma \sin \varphi_k$ ,  $\beta_k = \omega_\gamma \cos \varphi_k$ ,

$\varphi_k = (2k-1)\pi/2p$ ,  $\psi_{p,k} = (0,5p-1)(0,5\pi + \varphi_k)$ ,  $c_{p,k} = 1/\beta_k a_{p,k}$ ,  $a_{p,k} = \prod_{i=1}^{p/2} (\alpha_i - \alpha_k) |_{k \neq i}$ .

ИХ (5) хорошо аппроксимирует реальные ИХ проводных каналов связи, а для беспроводных каналов показывает некоторый частный случай частотно-селективных замираний.

## 3. Результаты работы канального прекодера

В результате работы алгоритма канального прекодирования (предыскажения) (2) получаем формирующие импульсы на основе гауссового эталона, показанные на рис. 2.

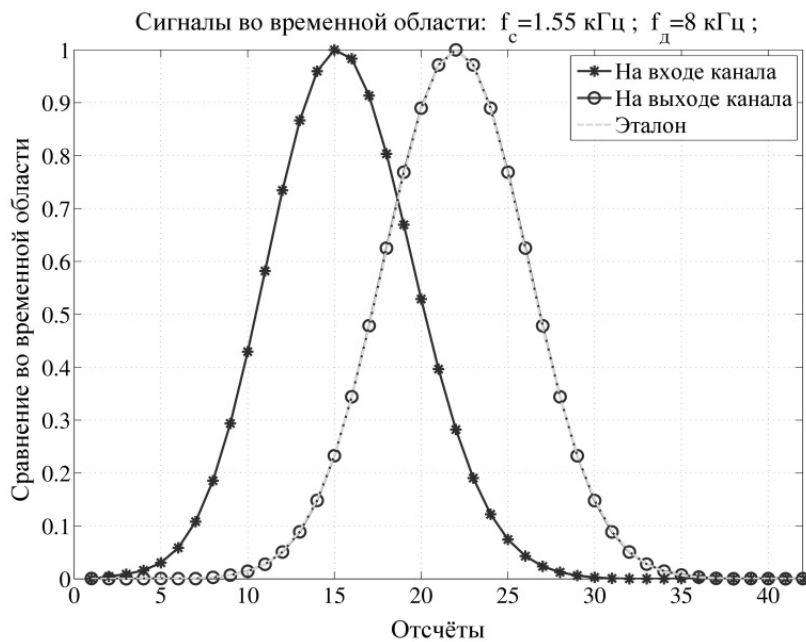
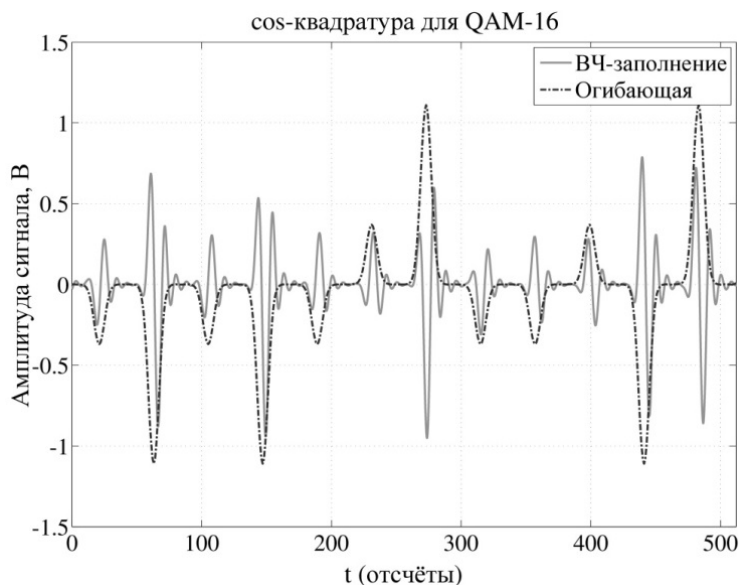


Рис. 2. Синтезированные формирующие импульсы

Как видно из данного рисунка, формирующий импульс на входе канала не обладает центральной симметрией и имеет специфическую форму. При этом после прохождения канала связи формирующий импульс центрально симметричен и практически полностью совпадает с гауссовым эталоном. Таким образом, для синтезированных импульсов канал связи можно рассматривать как фильтр, компенсирующий внесённые в импульс предыскажения.

Теперь рассмотрим применение полученных оптимальных импульсов. Продемонстрируем на рис. 3 одну из квадратур QAM-сигнала, сформированного по формуле (3). В данной квадратуре вместо стандартных прямоугольных формирующих импульсов  $g_m(t)$  используется финитная сигнальная система, вычисленная по правилу (2).

Оценим помехоустойчивость полученного QAM-сигнала с помощью построения графика BER (Bit Error Rate). На рис. 4 приведены результаты статического моделирования BER.



**Рис. 3.** Квадратура QAM-сигнала с гауссовыми формирующими импульсами

График BER демонстрирует, что QAM модуляция с использованием синтезированных прекодером оптимальных формирующих импульсов достигает теоретической помехоустойчивости, рассчитанной в предположении, что МСИ отсутствует. Тогда как QAM сигнал с прямоугольными формирующими импульсами показывает значительно худшие результаты, т.к. помимо белого шума он подвергается существенной межсимвольной интерференции. Таким образом, исследуемый алгоритм канального прекодирования (предыскажения) выполняет свою задачу, практически полностью подавляя МСИ. За счёт этого не только повышается помехоустойчивость, но и упрощается детектирование, т.к. его можно вести в предположении, что МСИ отсутствует.

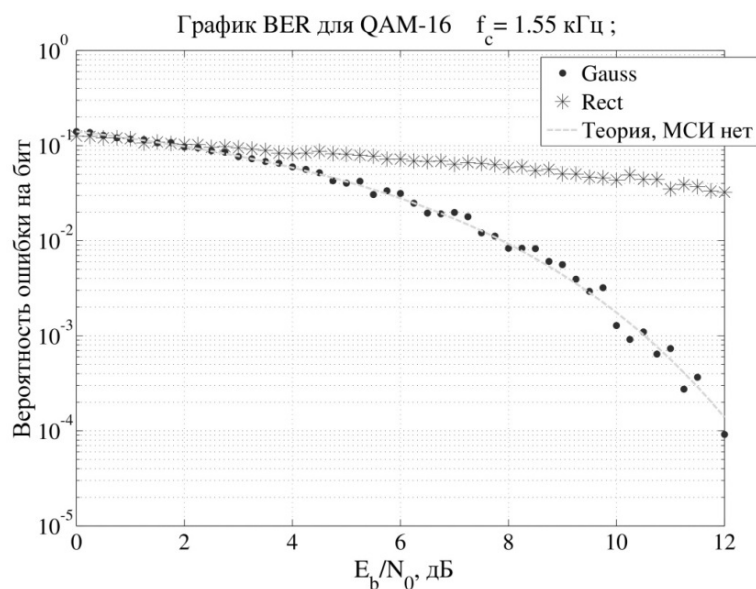
#### 4. Выводы

1. Предложен алгоритм работы дискретного канального прекодера, формирующего финитные канальные импульсы, оптимальные для текущих параметров канала связи и имеющие хорошую частотно-временную локализацию.

2. Формируемые прекодером канальные и опорные базисные импульсы практически не подвержены МСИ, что позволяет не учитывать её при построении схемы оптимального приёмника.

3. Предложенный алгоритм прекодирования реализован в виде программного кода в среде математического пакета Matlab, проведено математическое моделирование, подтвердившее его свойства.

4. Результаты математического моделирования по оцениванию помехоустойчивости сигналов QAM модуляции с синтезированными алгоритмом прекодирования показывают превосходство над традиционной QAM с прямоугольными формирующими импульсами.



**Рис. 4.** График BER для QAM сигнала с различными формирующими импульсами

### Литература

1. Волчков В.П. Сигнальные базисы с хорошей частотно-временной локализацией // Электросвязь. 2007. №2. С. 21-25.
2. Волчков В.П., Безруков И.М. Синтез дискретного оптимального прекодера с финитной системой сигналов для систем OTDM // Фундаментальные проблемы радиоэлектронного приборостроения. 2015. Т. 15. №5. С. 48-51.
3. Волчков В.П., Уваров С.С. Аппроксимация узкополосных случайных процессов с помощью комплексной рекуррентной  $m$ -модели скользящего окна второго порядка // Т-Comm: Телекоммуникации и транспорт. 2015. Т. 9. № 3. С. 54-61.

# АНАЛИЗ ВОЗМОЖНОСТИ ПРОГРАММНО-АППАРАТНОЙ РЕАЛИЗАЦИИ ЛАБОРАТОРНОЙ РАБОТЫ «ФОРМИРОВАНИЕ СИГНАЛОВ АМПЛИТУДНОЙ МОДУЛЯЦИИ»

Дуплищева Я.В.,  
МТУСИ, студентка группы БЗС1201,  
Калугин Ю.А.,  
МТУСИ, аспирант 1-ого года обучения  
Сухоруков А.С.,  
МТУСИ, к.т.н. доцент  
Терехов А.Н.,  
МТУСИ, к.т.н. доцент

Аппаратные стенды, находящиеся в лаборатории кафедры общей теории связи (ОТС) в настоящее время, претерпевают моральное и материальное старение. Проведение цикла лабораторных работ требует выполнить разработку новых стендов, которые заменят существующие. При этом целесообразно использовать цифровые устройства для использования программных, а не аппаратных средств, позволяющих оперативно создавать новые, или изменять существующие лабораторные работы. Целью данной работы является анализ возможности создания программно-аппаратного комплекса - стенда, для проведения лабораторных работ на кафедре ОТС, в частности, лабораторной работы «Формирование сигналов амплитудной модуляции».

Для начала, рассмотрим требования, предъявляемые к аппаратной части стенда и рассмотрим существующие цифровые устройства. К стенду предъявляются следующие требования:

- формирование АМ сигнала как с внешней, так и встроенной моделью информационного сигнала;
- возможность управления и изменения параметров стенда как с применением компьютера, так и без него;
- высокая эффективность применения используемых компонент;
- простота эксплуатации и обслуживания.

Создание стенда на дискретных элементах не рассматривается так, как при их использовании затруднен ремонт и нет возможности добавлять новые лабораторные работы.

Обзор устройств для создания стенда:

- микроконтроллер (англ. *MicroControllerUnit*, *MCU*) – микросхема, предназначенная для управления электронными устройствами, имеющая на одном кристалле процессор и периферийные устройства, ОЗУ и ПЗУ. Наиболее известные производители микроконтроллеров – фирмы *Atmel*, *Microchip*, *TexasInstruments*.
- цифровой сигнальный процессор (англ. *DigitalSignalProcessor*, *DSP*) – специализированный микропроцессор, предназначенный для цифровой обработки сигналов. Наиболее известные производители – *AnalogDevices*, *TexasInstruments*.
- программируемая пользователем вентильная матрица (англ. *Field-ProgrammableGateArray*, *FPGA*) – специальное полупроводниковое устройство, которое может быть сконфигурировано разработчиком после изготовления при помощи специальных языков проектирования (*VHDL*, *VerilogHDL*). Наиболее известные производители – *Altera*, *Xilinx*.

Для удовлетворения требования простоты реализации и обслуживания, целесообразно использовать устройство с корпусом *DIP (DualIn-LinePackage)* – т.е. с типом корпуса, имеющим прямоугольную форму с двумя рядами выводов по длинным сторонам. Микросхемам с DIP-корпусами не требуется пайка, они вставляются в специальные *DIP*-панели. При ремонте стенда, такая микросхема, в случае ее повреждения, просто заменяется на аналогичную. В случае использования микросхем с другими типами корпусов, для замены потребуется пайка, для чего необходимо иметь или квалифи-

цированного специалиста, или обращаться к сторонним предприятиям. Поэтому, для достижения максимальной оперативности, и простоты ремонта, будут использованы микросхемы с корпусом *DIP*.

Применение микросхем с корпусом *DIP* требует обеспечить выбор устройств с учётом данного ограничения. Этому требованию удовлетворяют только микроконтроллеры.

Затем следует выбор модели микроконтроллера, к которой предъявляются следующие требования:

- более двух 8-ми битных портов ввода-вывода;
- встроенный АЦП;
- наличие аппаратного умножения;
- модуль для обеспечения взаимодействия с ПК (*UART* или *SPI*);
- производительность соизмеримая с решаемой задачей;
- низкая стоимость;
- наличие защиты ПО от несанкционированного копирования.

Требование к наличию встроенного ЦАП не предъявляется, так как его можно изготовить при помощи внешних элементов. Основное преимущество микроконтроллеров фирм *Microchip* и *Texas Instruments* высокая производительность при низком энергопотреблении (до сотен нА). Разрабатываемый стенд будет использовать питание от сети (или *USB*-шины), поэтому требование низкого энергопотребления не является критичным. Исходя из экономической целесообразности, выбираем микроконтроллер *Atmega8* фирмы *Atmel*. Он имеет следующие параметры:

- три 8-ми битных порта ввода-вывода;
- встроенный 6-канальный 10-битный АЦП;
- аппаратный умножитель;
- производительность до 16 MIPS при тактовой частоте 16 МГц;
- низкая стоимость;
- встроенные модули *UART* и *SPI*;
- возможность защиты ПО (биты защиты).

Функциональная схема стенда приведена на рис. 1.



Рис. 1. Функциональная схема

В качестве формирователя управляющих сигналов может использоваться ПК, или кнопочное управление. Микроконтроллер формирует выходной цифровой сигнал, который передается на ЦАП. ЦАП реализуется при помощи матрицы R-2R. Структурная схема стенда представлена на рисунке 2.

Для управления стендом при помощи ПК создано программное обеспечение, которое потребовало выбор языка программирования высокого уровня. На сегодняшний день существует большое разнообразие языков программирования. Нам нужен такой язык, который удовлетворяет требованиям простоты разработки приложений и имеет возможность работать с последовательным портом компьютера. Такие языки как *C#* или *Java* используют виртуальную машину, поэтому работа с последовательным портом через них затруднена. Поэтому для создания программного обеспечения выбран язык *Delphi*. Он сочетает в себе простоту использования и разработки, и имеет необходимые инструменты для разработки приложений, взаимодействующих с последовательным портом. Также, можно использовать сторонние программы, позволяющие управлять устройствами через последовательный порт, например, *MATLAB*.

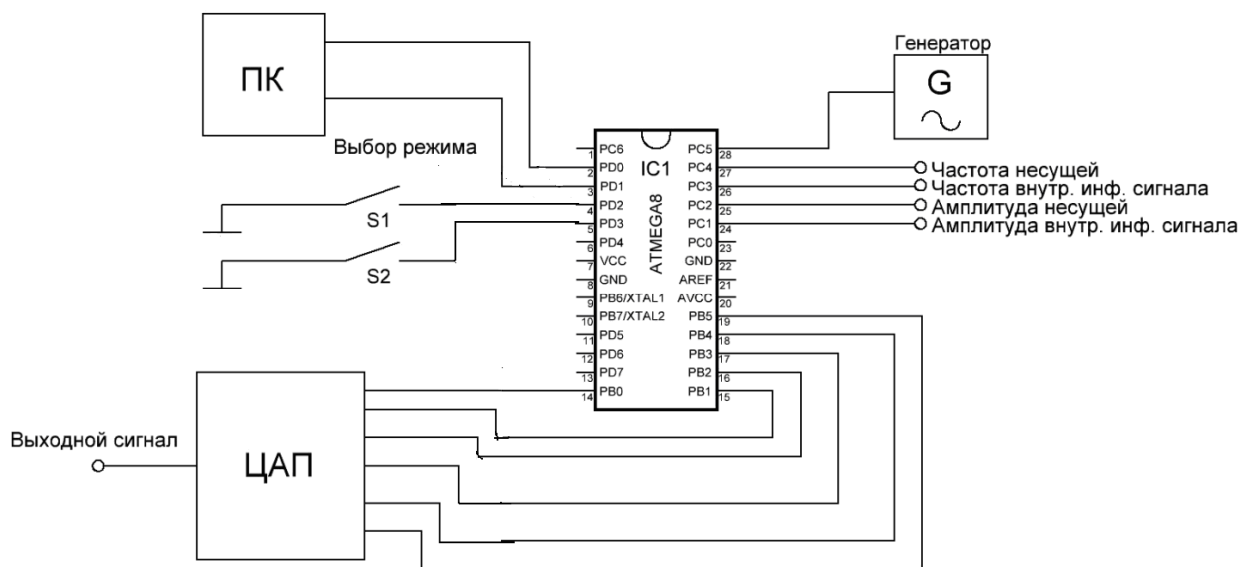


Рис. 2. Структурная схема стенда

#### Выводы:

1. Разработанный стенд на 95% использует ресурс выбранного микроконтроллера, что обеспечивает высокую эффективность решения поставленной задачи.
2. Реализация стенда на базе микроконтроллера, выпуск которого освоен отечественной промышленностью позволяет решать поставленную задачу в условиях импортозамещения.
3. Наличие встроенного модуля *UART* обеспечивает управление с применением ПК.
4. Стенд обеспечивает защиту ПО, что немаловажно в условиях рыночных отношений.

#### Литература

1. ATmega8(L) Complete Datasheet.
2. Архангельский А.Я. Программирование в Derphi–М.: Бином-Пресс, 2008.
3. Венедиктов М.Д., Рысин Ю.С., Терехов А.Н. Программно аппаратный комплекс для оценки параметров сетей, обеспечивающих телефонные услуги связи // Т-Сотт: Телекоммуникации и транспорт. 2013. Т. 7. № 8. С. 39-43.
4. Терехов А.Н. Проект методики расчета оптимального количества аудио информационных устройств, необходимого для информирования абонентов тфоп об изменении телефонной нумерации // Т-Сотт: Телекоммуникации и транспорт. 2012. Т. 6. № 4. С. 79-82.
5. Рысин Ю.С., Терехов А.Н. Алгоритм оценки влияния негативных факторов на качество телефонного общения // Т-Сотт: Телекоммуникации и транспорт. 2012. Т. 6. № 10. С. 96-98.



# ПОКОЛЕНИЯ ПРОГРАММ ИСКУССТВЕННО-СОЦИАЛЬНЫХ СИСТЕМ

*Дюркович И.В., Силантьева А.С.,  
МТУСИ, ОТФ-2, студенты группы: БИН1404  
Корзинкин В.В.,  
МТУСИ*

## **Введение**

Искусственными системами будем называть системы, созданные и развитые человеком - любые материальные продукты и технологии, воплощенные в реальность людьми. То есть, если одна деталь, часть была намеренно подвержена изменению человеком, такая система уже может рассматриваться как искусственная.

Главное отличие искусственных систем от естественных - это их «искусственность», их производство человеком. Это дает несколько очень важных эффектов:

- Огромное упрощение искусственных систем по сравнению с биологическими. (Не производят сами себе подобных особей)
- Не использование в конструкциях лимитирующих развитие органических молекул (белки и т.п.), что позволит использовать в технике предельные нагрузки (механические, температурный, давление и т.д.). Станет возможным применение вредных веществ для биологических видов (сильные кислоты и основания, радиоактивных веществ и т.п.).
- Передача идей, опыта из одних областей в другие, с одних систем на другие
- Возможность сознательного и целенаправленного управления развитием для достижения тех или иных целей

## **Искусственно-социальная система.**

Говоря не посредственно о ИСС, вспомним определение социума :

Социум - это одна огромная группа людей, которых могут объединять общие моральные устои, отношение к миру и самим себе. В подобной группе всегда единая система ценностей и мировоззрений, одинаковые политические и эстетические взгляды.

Следовательно:

ИСС - это такая искусственная система, в которой моделируется виртуальное сообщество интеллектуальных агентов, где ее составляющие взаимодействуют между собой, другими такими же системами и человеком.

## **Интеллектуальные агенты (ИА) и поколения программ ИСС.**

Агентно-ориентированный подход основан на применении интеллектуальных (рациональных) агентах. Согласно этой разработке, интеллект — это вычислительная способность достигать поставленных перед интеллектуальной машиной целей. ИА (интеллектуальный агент) будет взаимодействовать с окружающим его миром с помощью датчиков, при этом обладая возможностью воздействовать на другие объекты с помощью исполнительных механизмов.

Значительное внимание уделяется методам и алгоритмам, благодаря которым интеллектуальный агент адаптирован к окружающей среде при выполнении его задачи. Так, здесь особый акцент делают на изучение алгоритмов поиска пути и принятия решений.

Агенты делятся на:

1. Автономных: функционирует без вмешательства владельца, контролируют свое внутреннее состояние и свои действия
2. Неавтономные

Сильный и слабый искусственные интеллекты — гипотеза в философии искусственного интеллекта, согласно которой некоторые формы искусственного интеллекта могут действительно обособиваться и решать проблемы. Теоретически сильный ИИ (искусственного интеллекта) подразумевает,

что компьютеры смогут мыслить и осознавать себя, но не обязательно, что это будет аналогично мыслительному процессу человека. Теория слабого ИИ опровергает осуществимость такого .

По уровню развития ИИ агенты разделяют на:

1. Реактивных
2. Интеллектуальных

Реактивным или слабым агентам присуще следующие характеристики: автономность, социальное поведение( = дружелюбность), реактивность, про-активность;


Интеллектуальные или сильные агенты обладают характеристиками реактивных, причем с такими дополнениями как: базовые знания, наличие целей/генерация целей, наличие желаний(стоит рассматривать как понятие близкое к потребностям), обязательств и т.д.

В ходе дальнейшей эволюции сильных агентов требуется определить какие из дополнительных характеристик обязательны и какие нет. Это необходимо для предотвращения возможной угрозы техногенной катастрофы.

Разумные Агент - такой интеллектуальный агент, который обладает историчностью (наличием исторического прототипа и близостью к нему), возрастом, сознанием, психологическими чертами личности, способностью общаться, встроенностью в гетерогенный коллектив персонажей v.1 и v.2. Имитационные Агенты - имитируют разумность.

Система, в которой агенты не могут образовать собственный союз, называется многоагентной. То есть каждый элемент (агент) существует самостоятельно в какой-то определенной сфере, допустимой для него. Многоагентная система образуется полуслабыми и слабыми ИА, а сильный ИИ , являясь системой более высокого уровня, обеспечит их взаимодействие и позволит развиваться такой системе как мультиагентная.

Особенности поколений программ:

ИИ	Необязательность безвредности (дружелюбности) к пользователям и другим агентам	Направление увеличения интеллекта агентов и угрозы техногенной катастрофы		
				
ДИИ ДАИС	Обязательность безвредности к пользователям и другим агентам	v.1 Интеллектуально "полуслабые" Реактивность, коммуникативность с пользователем	v.2 Интеллектуально "слабые " Автономны, Про-активны	v.3 Интеллектуально "С наборами элементов "Сильного""

\* v.1, v.2, v.3 (версии)

Мультиагентной называется система, где ИА образуют собственное объединение, союз (систему).

Таким образом, слабые и сильные агенты будут взаимодействовать и смогут выполнять больше всевозможных операций. Например, слабый ИА не имеет возможности сам себя достаточно обеспечить ресурсами и, как только ему что-то становится необходимым, работа прекращается. В этом случае объединение слабых и сильных ИА решает поставленную задачу: сильный ИИ имеет связь не только с одним агентом, но и с другими такими элементами системы, значит есть шанс исправить ситуацию с помощью других частей такого союза.

### Реальные перспективы развития.

При достижении хорошего функционирования искусственно-социальной системы возможно на ее основе создать систему высшего уровня, связав сильные ИА в одно объединение. Тем самым, получить систему, где слабые посредством сильного ИИ взаимодействуют между собой, а сильные взаимодействуют с сильными ИА сами по себе или использования более интеллектуального агента . Если уйти от данного описания, можно представить это проще - как соединение нескольких мультиагентных систем и их слаженной работе.

Но нужно ли обеспечивать их контроль более интеллектуальным агентом? Конечно, для полной автоматизации необходимо поставить во главу всех этих программ другую, которая будет следить за их действиями и ресурсообеспечением (в зависимости от требований ИА). Но с другой стороны мы вновь встаем перед вопросом о том, стоит ли делать более интеллектуальные системы и чем их создание может грозить человеку, к чему приведет недовзвешенность безусловно важных проблем.

Однако, развитие поколений программ ИСС позволит захватить разные области деятельности человека и благодаря автоматизации станет огромным прорывом в науке. Появится 1000 разных возможностей реализации и внедрения таких программ в жизнедеятельность человека, ведь даже сейчас есть теоретически завершённые проекты для улучшения жизни людей, но не воплощенные в реальность по причине недостатка таких программ.

Дальнейшая разработка программ ИСС позволит реализовать множество инновационных идеи. Например: "Интернет Вещей" или "Умный Дом".

(Ниже представлены их краткие описания)

Интернет вещей — концепция вычислительной сети физических объектов («вещей») со встроенными технологиями для взаимодействия друг с другом или с внешней средой.

Причем организацию таких сетей следует рассматривать как явление, которое способно координально поменять процессы экономики и общества, исключая необходимость участия человека в некоторых операциях и части действий.

"Умный дом" — дом современного типа для проживания людей, особенность организации которого заключается в автоматизации и оснащении высокотехнологичными устройствами.

"Умный дом" необходимо понимать как систему, обеспечивающую защиту и сбережение ресурсов (также относящийся к ним комфорт) для всех пользователей.

В простейшем варианте такая система должна распознавать конкретную ситуацию, которая происходит в доме, и определенным образом реагировать на нее: одна из систем управляет другими с помощью заранее установленных алгоритмов. Стоит обратить внимание на то, что автоматизация нескольких подсистем обеспечит синергетический эффект для всего комплекса.

Можно сказать, что более прогрессивной концепцией взаимодействия пользователя с пространством проживания является задание и контроль в автоматизированном режиме в соответствии с внешними и внутренними факторами качества работ всех систем и электроприборов.

Благодаря этому исключается необходимость пользоваться несколькими ТВ-пультами при просмотре передач, немалым количеством выключателей освещения, системами отопления и кондиционирования, сигнализацией и т.п..

### **Заключение.**

Физиологические способности человека ограничены и, как это неудивительно, в современном мире тормозят процесс наращивания темпов и масштабов производств.

Технологиям требуется освоить процессы самоорганизации и самосовершенствования, которые тщательно разрабатывает человек последнее время. Это сможет радикально изменить жизнь людей абсолютно во всех сферах, что необратимо ведет к новым сложным экономическим, социальным и этическим проблемам. При такой постановке вопросов возникает необходимость в дальнейшем осмыслении и методологических обоснованиях.

Добиваясь шаг за шагом новых высот в развитии технологий и автоматизации, меняя мир вокруг и облегчая жизнь человека, не стоит забывать о возникающих вопросах и их решениях. Следует понимать, какая ответственность лежит на каждом решении, нужно определить рамки дозволенного и запретного, нужно понять что мы хотим получить и действительно ли это нам так нужно.

### **Литература**

1. [https://ru.wikipedia.org/wiki/Интернет\\_вещей](https://ru.wikipedia.org/wiki/Интернет_вещей).
2. [www.ideationtriz.com/ZZLab/Resources/Artificial\\_systems.htm](http://www.ideationtriz.com/ZZLab/Resources/Artificial_systems.htm).
3. [amur-dom.ru/about](http://amur-dom.ru/about).
4. *Аджемов А.С.* и др. Детерминационный анализ и интеллектуальные системы (ДАИС). группа соавторов МГУСИ по интеллектуальным системам.

# ОПТИМАЛЬНАЯ ФИЛЬТРАЦИЯ ДИСКРЕТНОГО СИГНАЛА С УЧЕТОМ МЕЖСИМВОЛЬНОЙ ИНТЕРФЕРЕНЦИИ

Походун А.И.,  
МТУСИ, студентка группы БП31301  
Павлюк В.В.,  
МТУСИ, к.т.н., доцент

Дискретные сигналы имеют бесконечный спектр. Полоса же каналов передачи сигналов ограничена. Перед разработчиками стоит задача найти оптимальную для полосы частот, чтобы передать сигнал с максимальной достоверностью и минимальными искажениями.

Прямоугольный видео сигнал с амплитудой  $U$  и длительностью  $T$ , имеет спектральную плотность напряжения:

$$S(f) = UT \frac{\sin \pi f T}{\pi f T} \quad (1)$$

Как следует из графика спектра передаваемого сигнала (Рис.1), он занимает бесконечную полосу частот. Но с увеличением частоты, интенсивность напряжения убывает, поэтому имеет смысл при передаче ограничить его спектр.

Спектр прямоугольного сигнала и полосы пропускания идеального ФНЧ при ПТ=0.5, 1 и 2 изображены на рис.1.

Сигнал на выходе фильтра определяется с помощью обратного преобразования Фурье:

$$U_{\text{вых}}(t) = \frac{U}{\pi} (Si(2\pi\Pi(t + \frac{T}{2})) - Si(2\pi\Pi(t - \frac{T}{2}))) \quad (2)$$

Графики выходного сигнала при различных значения ПТ представлены на рис. 2.

Как видно из графика, фильтр искажает форму сигнала. При уменьшении ПТ (т.е. сужении полосы фильтра), амплитуда сигнала уменьшается, и появляются межсимвольные помехи  $U_i$ , которые, складываясь с шумом, будут искажать предыдущие и последующие импульсы, что приводит к их неправильному приему, т.е. к увеличению вероятности ошибки.

Найдем оптимальную полосу ПТ, которая обеспечивает максимальное преобладание сигнала над помехами – шумом и межсимвольной интерференцией. Рассмотрим отношение мощности сигнала к мощности шума.

На графике изображены зависимость максимального значения напряжения сигнала  $U_0$  на выходе ИФНЧ от ПТ и отношение сигнал/шум.

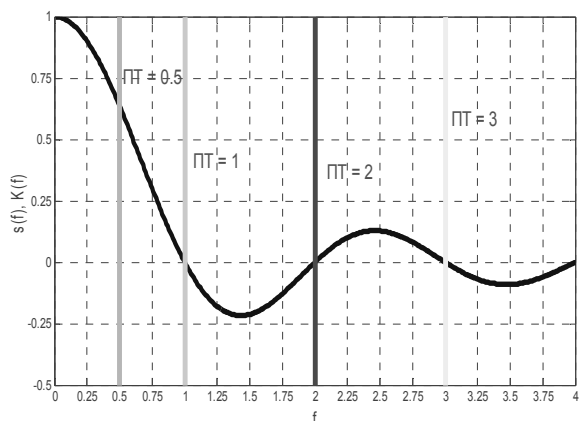


Рис. 1. Спектр сигнала и АЧХ ИФНЧ.

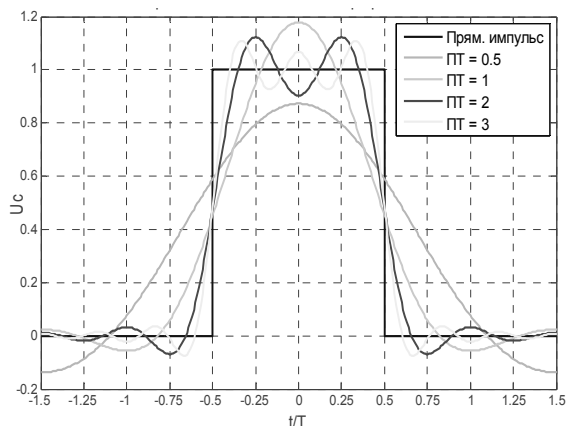


Рис. 2. Сигнал на выходе ИФНЧ

Отношение максимального значения напряжения сигнала на выходе ИФНЧ к ПТ можно представить в виде:

$$U_0 = Um \frac{2}{\pi} Si(\pi ПТ) \quad (3)$$

Мощность шума равна:

$$P_{ш} = G_0 \Pi \quad (4)$$

где  $G_0$  - энергетический спектр шума.

Формула для отношения сигнал/шум:

$$\frac{P_c}{P_{ш}} = \frac{U_0^2}{G_0 \Pi} \quad (5)$$

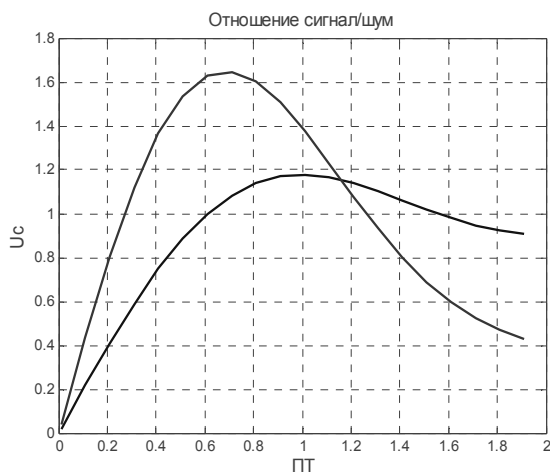


Рис. 3. Отношение сигнал/шум.

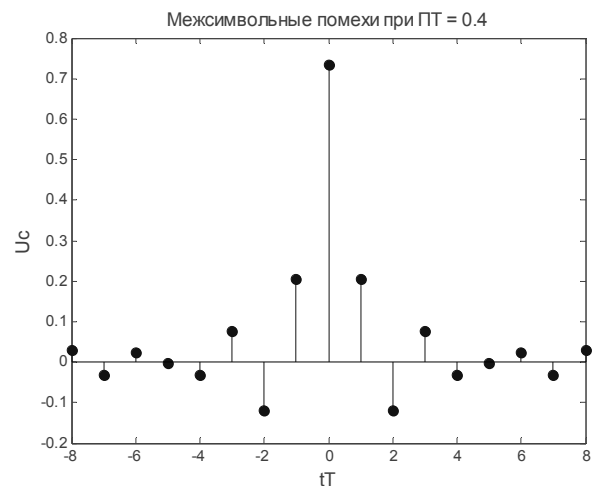


Рис. 4. Межсимвольные помехи  $U_i$ .

Как видно из графика, максимальное значение  $\frac{P_c}{P_{ш}}$  достигается при  $ПТ = 0.7$ .

С увеличением полосы (при  $ПТ > 0.7$ ) преобладание сигнала над шумом уменьшается. Отсюда делаем вывод, что оптимальная полоса без учета интерференционных помех равна 0.7.

Найдем оптимальную полосу с учетом межсимвольных помех:

На рисунке 4. представлены межсимвольные помехи на выходе ИФНЧ. После прохождения импульса через прямоугольный фильтр, появляются межсимвольные помехи  $U_i$ , которые искажают как предыдущие, так и последующие  $i$ -тые импульсы.

$$U_i = \frac{U}{\pi} (Si(2\pi ПТ (i + \frac{1}{2})) - Si(2\pi ПТ (i - \frac{1}{2}))) \quad (6)$$

Так как количество межсимвольных помех велико, то  $\sum U_i$  имеет нормальный закон распределения с дисперсией  $\sum (U_i)^2$ .

Рассмотрим отношение мощности сигнала к мощности обеих помех – шума и межсимвольных помех -  $\frac{P_c}{P_{пом}} = \frac{P_c}{P_{ш} + P_{мсп}}$ .

Мощность сигнала равна:  $P_c = U_0^2$

Мощность помех:  $P_{пом} = P_{ш} + P_{мсп}$ . Здесь мощность шума -  $P_{ш} = G_0 \Pi$ , а мощность межсимвольных помех:  $P_{мсп} = \sum (U_i)^2$ , зависит от величины ПТ.

Зависимость  $P_{\text{мсп}}(\text{ПТ})$  представлена на Рис 5. Как следует из графика, минимальная мощность межсимвольной помехи будет при значении  $\text{ПТ} = 1$ . Мощность межсимвольной помехи максимальна при  $\text{ПТ} = 0.65$  и  $1.25$ .

Соотношение сигнал-помеха будет равно:

$$\frac{P_c}{P_{\text{пом}}} = \frac{U_0^2(\text{ПТ})}{G_0 \text{П} + \sum U_l^2(\text{ПТ})} \quad (7)$$

Из выражения (6) следует, что соотношение  $P_c/P_{\text{пом}}$  зависит от величины  $E_b/G_0$ , где  $E_b=U^2T$  энергия входного сигнала.

Приведенные результаты показывают, что оптимальная полоса фильтрации при воздействии шума равна  $0.7/T$ . При малом соотношении  $E_b/G_0$  значение оптимальной полосы равно  $0.75/T$ , т.к. при больших шумах нет смысла расширять полосу канала. При больших значениях  $E_b/G_0$  ( $p_{\text{ош}} < 10^{-3}$ ) оптимальная полоса увеличивается до  $0.9/T$ .

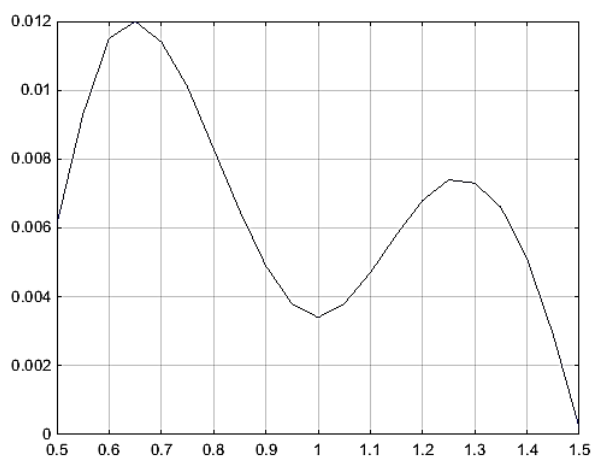


Рис. 5. Зависимость  $P_{\text{мсп}}$  от величины  $\text{ПТ}$

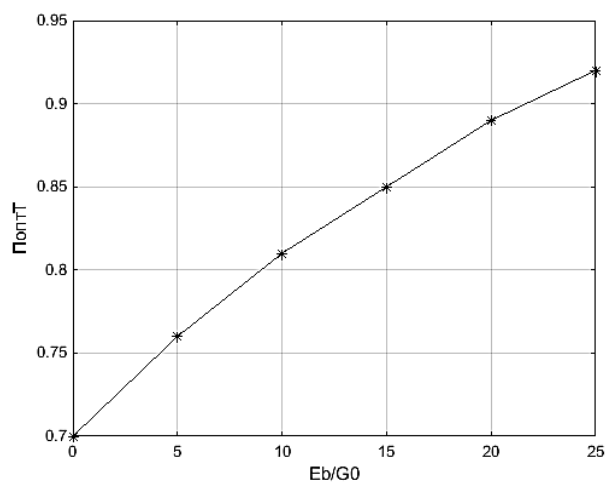


Рис. 6. Зависимость  $\text{П}_{\text{опт}} T$  от величины  $E_b/G_0$

## Литература

1. Власюк, И.В., Белов, А.С. Селекция нарушений с относительной пороговой оценкой структуры сигнала изображений в пределах дискретных фрагментов для телевизионных систем контроля безопасности объектов // Т-Сотм: Телекоммуникации и транспорт. 2010. Т. 4. № 9. С. 43-45.
2. Головкин И.В. Исследование сходимости итеративных алгоритмов обработки сигнально-кодовых конструкций с использованием дискретных частотных сигналов и турбо-подобных кодов // Т-Сотм: Телекоммуникации и транспорт. 2016. Т. 10. № 3. С. 21-29.