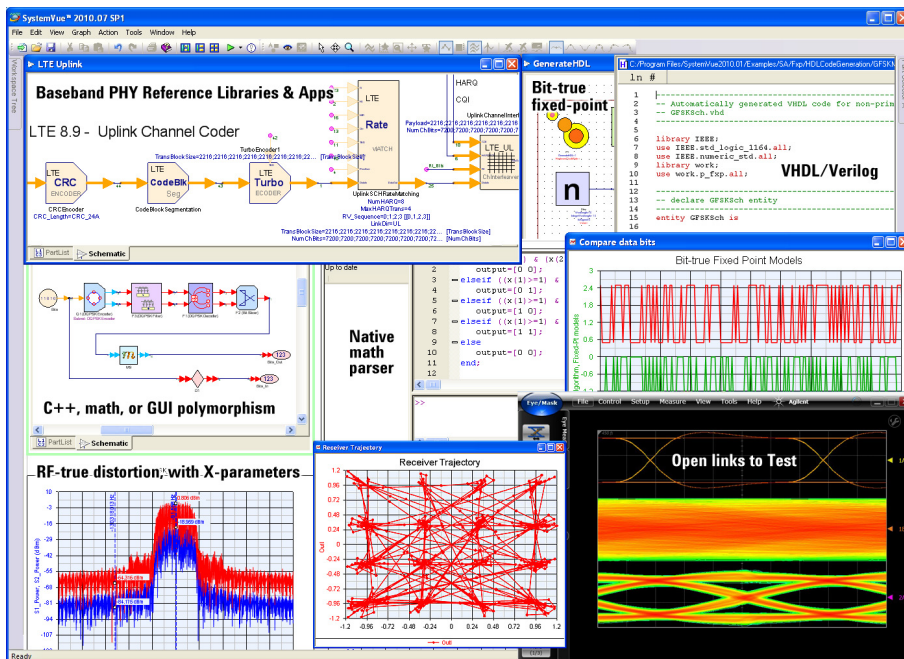


Keysight Technologies

Анализ и проектирование ВЧ и цифровых систем с помощью Keysight SystemVue

Учебное пособие



Анализ и проектирование ВЧ и цифровых систем с помощью Keysight SystemVue

Учебное пособие

Аннотация

Компания Keysight Technologies разработала целый ряд программных продуктов для проектирования, а также схемотехнического и системного анализа высокочастотного оборудования. Это учебное пособие познакомит вас с основами моделирования в среде Keysight SystemVue и расскажет о возможностях эффективного проектирования ВЧ и цифровых систем.

Содержание

Сокращения	6
Введение	7
1. SystemVue	7
2. Назначение.....	7
3. Цели.....	7
Глава 1. Интегрированные симуляторы САПР SystemVue	8
1.1 Введение	8
1.2 Традиционные методы моделирования	8
1.3 Поведенческое моделирование.....	8
1.3.1 Симулятор Data Flow	9
1.3.2 Spectrasys	11
1.3.3 Планировщик частот WhatIF	13
1.3.4 Свипирование по параметрам	14
1.3.5 Поведенческая оптимизация	15
1.4 Заключение	16
Глава 2. Основы проектирования ВЧ систем.....	17
2.1 Введение	17
2.2 Конструкция трансивера	17
2.3 Конструкция приёмника	18
2.3.1. Супергетеродинные приёмники	19
2.3.2 Приёмники с низкой ПЧ.....	19
2.3.3 Приёмники прямого преобразования.....	19
2.3.4 Приёмники с субдискретизацией	20
2.3.5 Сравнение архитектур приёмников.....	20
2.4 Конструктивные особенности системного уровня.....	21
2.4.1 Показатели качества	21
2.4.2 Уравнения дальности	21
2.4.3 Бюджет канала.....	22
2.4.4 Чувствительность и избирательность	23
2.4.5 Подавление зеркальных частот	24
2.4.6 Фазовый шум	24
2.4.7 Постоянное смещение	25
2.5 Нелинейное поведение ВЧ системы.....	25
2.5.1 Интермодуляционные искажения	26
2.5.2 Коэффициент шума	26
2.5.3 Внутриполосные и внеполосные помехи и блокирующие сигналы	27
2.5.4 Соседние каналы	27
2.5.5 Компрессия усиления	28
2.5.6 Динамический диапазон	29
2.6 Компромиссы при конструировании передатчика.....	30
2.6.1 Отношение сигнала к шуму.....	30
2.6.2 Планирование частот	30
2.7 Основные функциональные блоки ВЧ трансивера.....	31
2.7.1 Антенна	31
2.7.2 Усилитель (мощности/малошумящий).....	32

2.7.3	Фильтр	32
2.7.4	Дуплексер	32
2.7.5	Смеситель	33
2.7.6	Гетеродин	33
2.7.7	Детектор	34
2.7.8	Аналого-цифровой преобразователь	34
2.8	Заключение	34
Глава 3.	Проектирование ВЧ систем с помощью САПР SystemVue	35
3.1	Введение	35
3.2	Методы и подходы к проектированию	35
3.2.1	Общие подходы к проектированию	35
3.2.2	Обзор моделирования на системном уровне	36
3.3	Изучение на примере проектирования трёхдиапазонного приёмника LTE	37
3.3.1	Обзор LTE	37
3.3.2	Определение технических характеристик и проектирование приёмника	38
3.4	Заключение	53
Глава 4.	Введение в цифровую связь	54
4.1	Введение	54
4.2	Передатчик	54
4.3	Приёмник	55
4.4	Канал	55
4.5	Другие аспекты	55
Глава 5.	Проектирование передатчика	56
5.1	Основы конструкции передатчика	56
5.2	Входные данные	58
5.3	Кодирование	59
5.4	Формирование символов	60
5.5	Фильтрация формы импульсов	61
5.6	I/Q-модуляция	64
5.7	Анализ результатов	66
Глава 6.	Обзор приёмника	69
6.1	Основы конструкции приёмника	69
6.2	Обзор схемы приёмника	71
6.3	Демодулятор	73
6.4	Согласованная фильтрация	73
6.5	Синхронизация	74
6.6	Коррекция фазовых ошибок и ошибок синхронизации	77
6.7	Коррекция частотной ошибки	78
6.8	Оценка канала	81
6.9	Расформирование символов	82
6.10	Коррекция кода	82
Глава 7.	Моделирование системного уровня	83
7.1	Коэффициент битовых ошибок (BER)	83
7.2	Моделирование в нескольких областях	86
7.3	Генерация кода HDL	88
Глава 8.	OFDM	91

8.1	Введение	91
8.2	Структура фрейма OFDM	94
8.3	Полезные данные OFDM	95
8.4	Пилот-сигналы.....	95
8.5	Преобразование из частотной области во временную.....	96
Глава 9. Основы распределённого спектра CDMA		100
9.1	Введение	100
9.2	Лабораторная работа 1. Генерируемая последовательность битов.....	100
9.3	Лабораторная работа 2. Формирование символов.....	107
9.4	Лабораторная работа 3. Код Уолша	110
9.5	Лабораторная работа 4. Проектирование фильтра	112
9.6	Лабораторная работа 5. Модуляция	115
9.7	Лабораторная работа 6. ВЧ канал	117
9.8	Лабораторная работа 7. Демодулятор.....	121
9.9	Лабораторная работа 8. Полная система	125
9.10	Заключение	132
9.11	Приложение	132
Литература		133

Сокращения

3G	Третье поколение
3GPP	Партнёрский проект третьего поколения
ACPR	Относительный уровень мощности в соседнем канале
ADS	САПР Advanced Design System
AWGN	Аддитивный белый гауссовский шум
BER	Коэффициент битовых ошибок
CMOS	Комплементарная структура металл-окисел-полупроводник (КМОП)
CNR	Отношение несущей к шуму (Н/Ш)
CPU	Центральный процессор (ЦП)
CW	Непрерывный сигнал
DC	Постоянный ток
DR	Динамический диапазон
DSP	Цифровая обработка сигналов
E-UTRA	Развитие наземной сети радиодоступа UMTS
EVM	Амплитуда вектора ошибки
FDD	Дуплексный режим с частотным разделением
FSPL	Потери при распространении сигнала в свободном пространстве
G	Коэффициент усиления
HB	Гармонический баланс
HDL	Язык описания аппаратного обеспечения
IF	Промежуточная частота (ПЧ)
IMD	Интермодуляционные искажения
IP	Интеллектуальная собственность
LO	Гетеродин
LOS	Линия прямой видимости
LTE	Long-Term Evolution – стандарт беспроводной высокоскоростной передачи данных для мобильных терминалов
MDS	Минимально различимый сигнал
MIMO	Несколько входов, несколько выходов
NF	Коэффициент шума
OFDM	Мультиплексирование с ортогональным делением частот
OFDMA	Множественный доступ с ортогональным делением частот
P1dB	Точка компрессии на 1 дБ
PLL	Фазовая автоподстройка частоты (ФАПЧ)
PRB	Блок физических ресурсов
QPSK	Квадратурная фазовая манипуляция
RF	Высокая частота (ВЧ)
SAE	Развитие системной архитектуры
SC-FDMA	Множественный доступ с частотным разделением и одной несущей
SDF	Синхронный поток данных
SINAD	Отношения полного сигнала к полному уровню помех
SISO	Один вход, один выход
SNR	Отношение сигнала к шуму (С/Ш)
SP	Параметры матрицы рассеяния
SPARCA	Анализ распространения спектра и основных причин
TA	Анализ переходных процессов
TDD	Дуплексный режим с временным разделением
THD	Коэффициент нелинейных искажений
TOI	Точка пересечения по интермодуляционным составляющим 3-го порядка
UMTS	Универсальная система мобильных телекоммуникаций
VCO	Генератор, управляемый напряжением (ГУН)
VSWR	Коэффициент стоячей волны (КСВ)
XP	X-параметры*

Введение

1. SystemVue

Программное обеспечение SystemVue представляет собой специализированную САПР, предназначенную для проектирования электронного оборудования на системном уровне (ESL) и позволяющую инженерам-системотехникам и разработчикам алгоритмов совершенствовать физический уровень (PHY) беспроводных устройств и аэрокосмических/оборонных систем нового поколения. SystemVue предлагает уникальные возможности для разработчиков ВЧ устройств, а также пользователей цифровых сигнальных процессоров и ПЛИС/специализированных ИС, использующих в своих аппаратных платформах и радиочастотные схемы, и цифровую обработку сигналов.

САПР SystemVue легко интегрируется в имеющуюся технологию проектирования и позволяет сократить время разработки и верификации физического уровня в два раза.

Дополнительная информация о SystemVue приведена на странице www.keysight.com/find/eesof-systemvue

2. Назначение

Настоящее учебное пособие описывает общие принципы и методы системного моделирования с помощью пакета программ SystemVue компании Keysight Technologies.

В главах 1-3 описано применение этого ПО для проектирования ВЧ систем и моделирования ВЧ блоков. В главах 4-8 описаны основы цифровой связи и дано введение в теорию и общее устройство цифровых радиосистем.

3. Цели

1. Узнать, как использовать симуляторы Keysight для моделирования системного уровня, и начать документирование этой операции с особым акцентом на применении этих симуляторов для проектирования ВЧ систем.
2. Изучить и задокументировать библиотеку ВЧ моделей компании Keysight, познакомиться с характеристиками этих моделей, узнать о поддержке линейных и нелинейных режимов моделирования, поддержке измерения шума и т.п.
3. Предложить рекомендованную методологию проектирования ВЧ систем с помощью Keysight SystemVue.
4. Разработать некоторые сценарии ВЧ моделирования для перспективных беспроводных устройств и устройств аэрокосмического/оборонного назначения.
5. Познакомиться с теоретическими основами цифровой связи и общим устройством типовых цифровых радиосистем.
6. Спроектировать базовые компоненты цифровой радиосистемы с помощью SystemVue и провести типовые проектные исследования.
7. Выполнить моделирование цифровых коммуникационных компонентов в SystemVue и проанализировать полученные результаты.

Глава 1. Интегрированные симуляторы САПР SystemVue

1.1 Введение

Интегрированные САПР компании Keysight, такие как Advanced Design System (ADS) и Genesys, большей частью реализуют традиционные линейные и нелинейные методы моделирования ВЧ схем. Эти методы охватывают не все аспекты поведенческого моделирования цифровых и ВЧ систем. Чтобы дополнить этот набор инструментов проектирования, была создана САПР SystemVue. SystemVue является передовым средством проектирования и моделирования системного уровня, поддерживающим высокоуровневое проектирование трактов ВЧ и модулирующего сигнала. Она включает различные методы моделирования радиочастотных схем и схем модулирующего сигнала, работающие как в частотной, так и во временной областях. САПР SystemVue содержит ядра симулятора Data Flow и симулятора Spectrasys. Кроме того, она включает уникальный инструмент для планирования промежуточных частот, который называется планировщиком частот WhatIF (предлагается также в САПР ADS и Genesys).

1.2 Традиционные методы моделирования

ADS позволяет оценивать работу ВЧ схем с помощью методов линейного и нелинейного моделирования. Вот некоторые из этих методов:

- Расчёт параметров матриц рассеяния (SP): этот метод моделирования оценивает линейную частотную характеристику ВЧ устройств. Он основан на расчёте матрицы рассеяния цепи и параметров шума.
- Гармонический баланс (HB): метод моделирования в частотной области, используемый для расчёта установившегося отклика нелинейных ВЧ устройств. Он рассчитывает, например, такие показатели, как нелинейные искажения (THD) и точка пересечения по интермодуляционным составляющим 3-го порядка (TOI).
- Анализ переходных процессов (TA): метод моделирования во временной области, используемый для оценки изменения отклика ВЧ устройства во времени.
- Анализ цепей по постоянному току: позволяет рассчитывать характеристики ВЧ цепи на постоянном токе.

Большинство этих методов работает медленно и не позволяет выполнять поведенческое описание ВЧ систем. По этой причине САПР SystemVue включает методы моделирования, способные полностью описать поведение ВЧ систем или устройств, а также точно оценить их параметры в частотной и временной областях.

1.3 Поведенческое моделирование

Компания Keysight представила метод проектирования и моделирования ВЧ систем, ориентированный на поведенческое моделирование. Такие САПР Keysight, как SystemVue, поддерживают этот метод через симулятор Data Flow и симулятор Spectrasys. Поведенческие модели поддерживают разные типы спектра. Кроме того, они обладают достаточной гибкостью для поддержки спектров перспективного оборудования. И более того, в Spectrasys каждый порт по умолчанию является одновременно и входом, и выходом по отношению к типу используемого спектра.

Поведенческие модели поддерживают линейные и нелинейные устройства, а также анализ в частотной и временной областях. Поведенческая модель работает с разными типами моделей устройств и систем:

- Схемы: устройство можно моделировать по электрической принципиальной схеме, отражающей его поведение. Схемные блоки могут быть заранее определёнными моделями устройства. Для инкапсуляции цепей нижнего уровня можно использовать специальные компоненты, такие как *RF_LINK* и *CIRCUIT_LINK*, что позволяет использовать их в поведенческих моделях высокого уровня (например, в симуляторе Data Flow).
- Уравнения (например, Mathlang/MATLAB): многие особенности поведения устройства можно описать математическими уравнениями. Однако полностью описать уравнениями поведение устрой-

ства в целом удаётся не всегда.

- Сценарии VB: для настройки поведенческих моделей можно использовать язык сценариев Visual Basic. Эти сценарии позволяют моделировать многие свойства устройств.
- Модели на основе файлов: поведение устройства можно описать с помощью файлов данных, таких как файл Touchstone с S-параметрами (с возможными параметрами шума) или X-параметрами.

1.3.1 Симулятор Data Flow

Симулятор Data Flow используется для анализа коммуникационных систем на уровне алгоритмов с применением анализа ВЧ и модулирующих сигналов во временной области. ВЧ анализ в Data Flow включает анализ модуляционной информации относительно частоты несущей во временной области. Анализ Data Flow ВЧ систем может выполняться либо с помощью моделей потоков ВЧ данных, либо путём совместного моделирования с симулятором ВЧ архитектуры с помощью библиотеки RF Design Kit в SystemVue. Кроме того, принцип моделирования Data Flow ориентирован на расчёт откликов тракта модулирующего сигнала. Это позволяет проектировать и анализировать системы цифровой обработки сигналов (DSP) и системы с фиксированной запятой, а также генерировать код на языке описания аппаратного обеспечения (HDL).

Контроллер симулятора Data Flow взаимодействует с Spectrasys (ВЧ симулятор, встроенный в SystemVue) с помощью компонента *RF_Link*. Используя симулятор Data Flow для моделирования смешанных схем с динамическими ВЧ и модулирующими сигналами, разработчик может найти лучшее схемотехническое решение с возможностью использования готовых фрагментов (интеллектуальной собственности) из прежних разработок, сокращая тем самым время продвижения изделия на рынок.

Блоки и компоненты, представляющие ВЧ устройства в среде моделирования Data Flow, объединяются в граф Data Flow. Работа этого графа основывается на исполнении с управлением данными. Данный блок может исполняться лишь тогда, когда имеет на входах достаточный объём данных. Он выполняет расчёты и передаёт результирующие данные на выходы. На рис. 1.1.а приведён пример моделирования Data Flow, в котором некоторые цифровые компоненты модулирующего сигнала соединены между собой и образуют полосовой фильтр с конечной импульсной характеристикой (КИХ-фильтр). Входные и фильтрованные сигналы показаны во временной (рис. 1.1.б) и частотной (рис. 1.1.в) областях.

Большинство компонентов и блоков SystemVue работают в соответствии с методом Синхронного потока данных (SDF)¹, где число выборок и скорости генерации источника и потребления приёмника настроены заранее и известны до начала моделирования. Тем не менее, некоторые компоненты SystemVue можно использовать для моделирования динамического поведения коммуникационных систем. Среди этих частей можно упомянуть и специальные блоки: *DynamicPack_M* и *DynamicUnpack_M*.²

Технология моделирования Data Flow обладает и другими возможностями, которые реализованы в SystemVue. Вот наиболее интересные из них:

- Использование нескольких типов данных. Эта возможность позволяет одновременно оперировать разными типами данных, например, скалярными, с фиксированной запятой и матрицами.
- Моделирование с фиксированной запятой. SystemVue имеет в своей библиотеке HDL обширный набор компонентов и моделей с фиксированной запятой, которые можно использовать для построения, моделирования и анализа систем с фиксированной запятой. Затем такие системы можно конвертировать непосредственно в HDL и C++.
- Распределённое моделирование. SystemVue предлагает опцию удалённого запуска симуляторов Keysight GoldenGate. Эта опция полезна в том случае, если вы хотите запустить моделирование на машине с большим объёмом памяти или с более быстрым процессором.
- Совместное моделирование с другими языками и симуляторами. SystemVue позволяет выполнять совместное моделирование кода HDL с помощью ModelSim и файлов .m с помощью MATLAB.

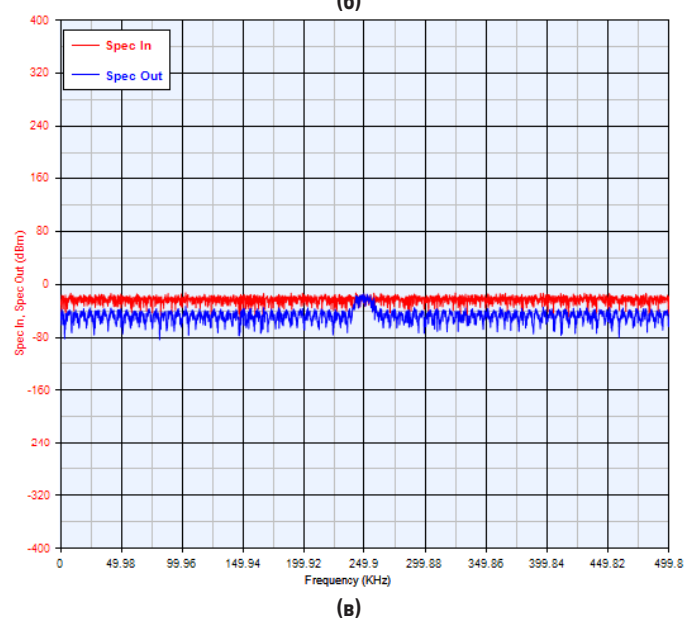
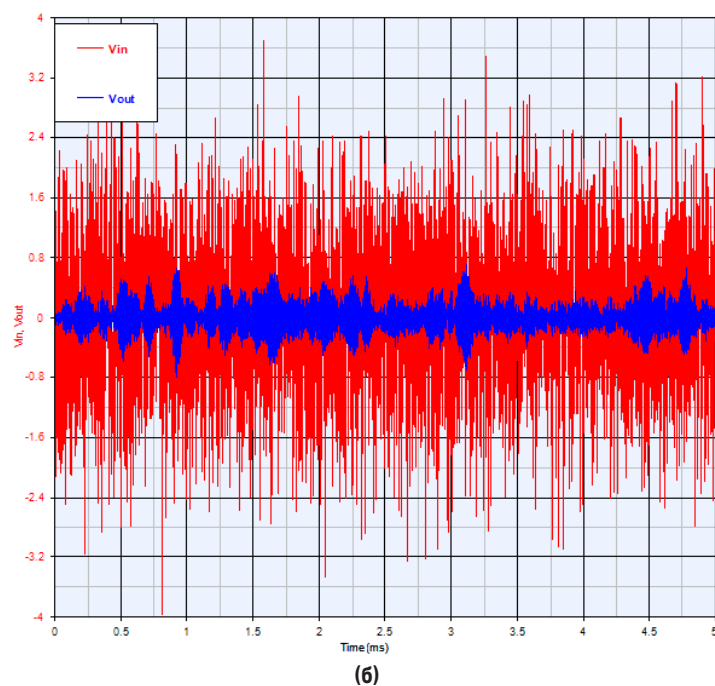
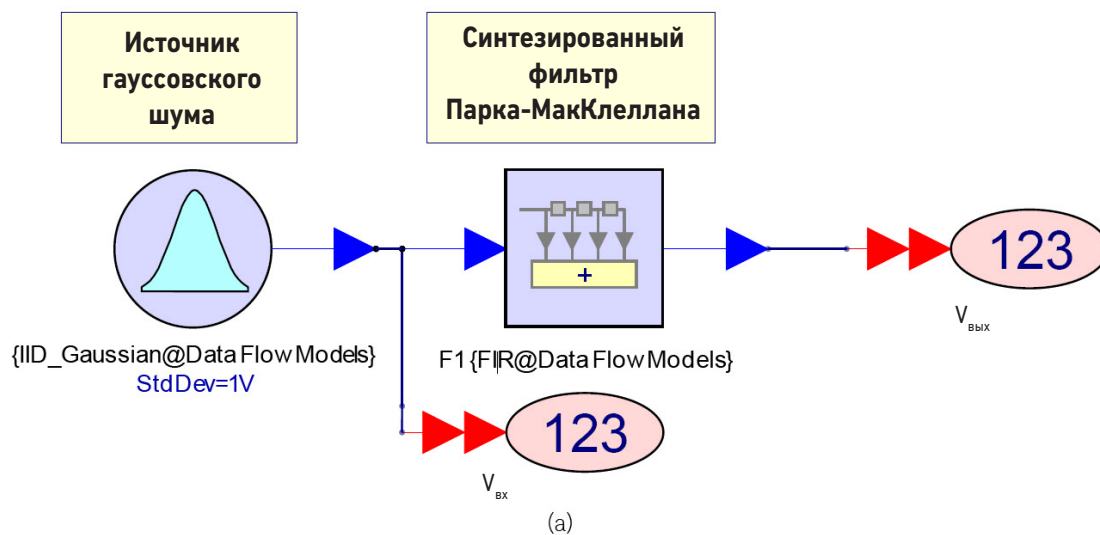


Рис. 1.1. Пример цифрового тракта модулирующего сигнала в симуляторе Data Flow: типовая схема фильтрации модулирующего сигнала (а), входной и отфильтрованный сигнал гауссовского шума (б), спектр входного и отфильтрованного сигнала гауссовского шума (в).

1.3.2 Spectrasys

Spectrasys – это технология компании Keysight, реализующая методы Анализа распространения спектра и основных причин (SPARCA) для поддержки моделей, основанных на измерении, и поведенческих моделей. Spectrasys предоставляет САПР SystemVue уникальную возможность выполнять точное и простое высокоуровневое моделирование и разработку ВЧ устройств и систем.

SPARCA поддерживает следующие типы спектра:

- Сигнал, включая источники сигнала и несущие.
- Интермодуляционные и гармонические составляющие, возникающие из-за нелинейности некоторых ВЧ устройств, таких как смесители и усилители.
- Широкополосный шум, представляющий собой тепловой шум ВЧ цепей.
- Фазовый шум, обусловленный различными причинами при обработке сигнала в ВЧ системах.

SPARCA учитывает тот факт, что каждый спектр распространяется в прямом и обратном направлении между каждым узлом схемы. Затем соответствующим образом рассчитываются все параметры шума, линейные и нелинейные параметры. Кроме того факта, что моделирование SPARCA выполняется очень быстро по сравнению с традиционными методами нелинейного моделирования, которым требуются критерии сходимости и большой объем алгебраических вычислений, этот новый метод обладает некоторыми дополнительными достоинствами. Вот некоторые преимущества применения SPARCA вместо традиционных методов моделирования:

- Полосы рассчитываются для всех спектров
- Легко оценивается широкополосный шум
- Поведенческий анализ влияния фазового шума
- Анализ многолучевого распространения
- Анализ путей утечки сигнала
- Гибкая поддержка перспективных спектров и расширений

На рис. 1.2 показаны поведенческие модели, которые могут обрабатываться в Spectrasys. Система моделирования совместима с ВЧ системами, состоящими из различных ВЧ компонентов и блоков (рис. 1.3), обеспечивая соответствующий отклик.

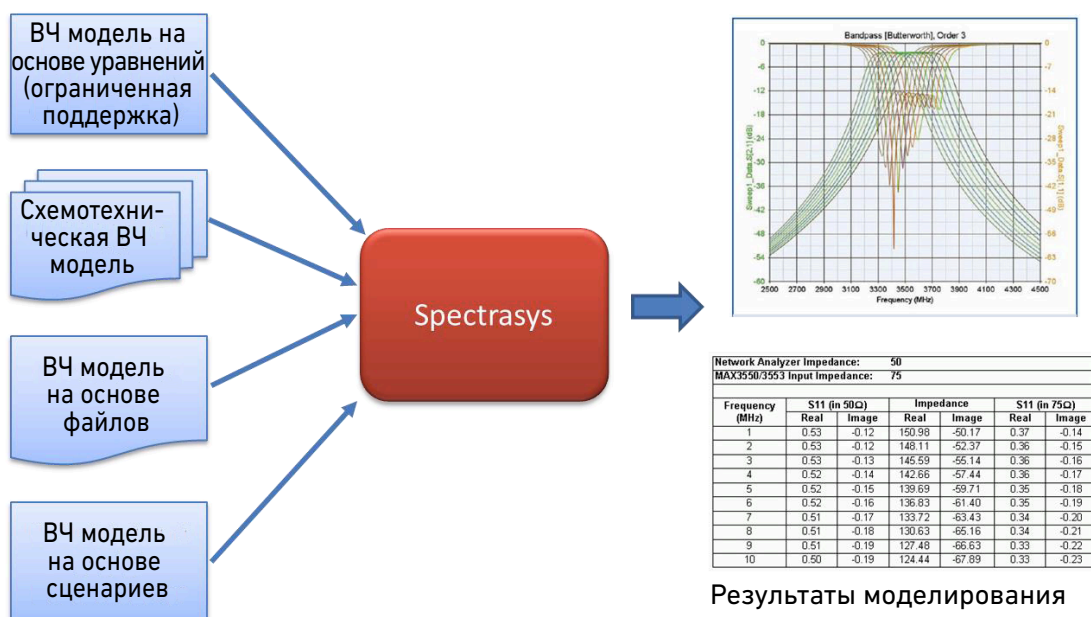


Рис. 1.2. Spectrasys может работать с различными ВЧ моделями.

SPARCA, будучи основой Spectrasys, призвана дополнить существующие методы моделирования. Она обеспечивает большую точность, скорость и производительность на системном уровне и облегчает изучение характеристик и компромиссов, связанных с данной ВЧ архитектурой.

Spectrasys обладает множеством замечательных возможностей, которые помогают разработчикам создавать, моделировать и анализировать системы связи и ВЧ оборудование. Среди этих возможностей можно упомянуть следующие:

- Источники. Spectrasys поддерживает разные типы сложных ВЧ источников в спектральной области и такие спектры, как непрерывный сигнал, широкополосный шум, а также сложные спектральные данные, созданные в результате моделирования или измерения.
- Каналы. Spectrasys выполняет измерения по частотным каналам, которые определяются центральной частотой и полосой.
- Сигнальные тракты. Как уже говорилось выше, Spectrasys считает каждый порт сигнального тракта и входом и выходом и рассчитывает частотную характеристику входного и выходного сигнала. Кроме того, Spectrasys позволяет определять сигнальные тракты произвольной архитектуры. Это дает возможность исследовать частотную характеристику каждого сигнального тракта и помогает получить более полное представление о системе.
- Диаграммы уровней. Эти диаграммы дают быстрое визуальное представление о характеристиках всего каскада. Они показывают измерения каскадно включённых цепей по определённому пользователем тракту.
- Индикация источника спектра. Поскольку каждый спектр отслеживается индивидуально, пользователь может найти источник и путь распространения каждого, интересующего его спектра.
- Широкополосный шум. Spectrasys рассчитывает широкополосный шум разных каскадов данного сигнального тракта с помощью метода моделирования SPARCA и отслеживает уровни шума для каждого компонента устройства.
- Интермодуляционные и гармонические составляющие. Spectrasys рассчитывает для каждого спектра интермодуляционные составляющие, гармоники и точки пересечения по интермодуляционным составляющим. Это позволяет выполнять расширенный анализ результатов моделирования, отображая разные параметры нелинейных поведенческих моделей (такие как развязка, точки пересечения по интермодуляционным составляющим, гармоники и т.п.).
- Когерентность. Поскольку все сигналы в Spectrasys обрабатываются индивидуально, когерентность каждого спектра выявляется во время моделирования. Когерентные сигналы складываются по напряжению и фазе, тогда как некогерентные – по мощности.
- Поведенческий фазовый шум. Система SPARCA поддерживает поведенческий фазовый шум, который можно указать для некоторых моделей источников и задающих генераторов. Фазовый шум представляет собой независимый тип спектра, и поэтому измерения этого спектра могут выполняться в присутствии других спектров различного типа. Такая независимость позволяет изменять фазовый шум при прохождении через смесители, умножители и делители, не влияя на исходный спектр.
- Свипирование и совместное схемотехническое моделирование. Spectrasys позволяет выполнять свипирование и совместное моделирование ВЧ схем с помощью других симуляторов (линейных симуляторов, гармонического симулятора Harbes и т.п.).
- Оптимизация. Оптимизацией называется метод, используемый для решения практических конструктивных вопросов, таких как номинальные значения и допуски на компоненты, которые могут быть чисто теоретическими. В частности, оптимизация используется, если число переменных проектирования слишком велико, и настройка становится менее эффективной из-за высокой размерности пространства переменных.
- Метод Монте-Карло и анализ выхода годных изделий. Это вероятностные методы, используемые для моделирования разброса параметров компонентов, вызванного производственным процессом. Эти методы можно использовать для выявления компонентов, нуждающихся в малых допусках (обычно более дорогих), или для создания устройств, устойчивых к разбросу параметров.

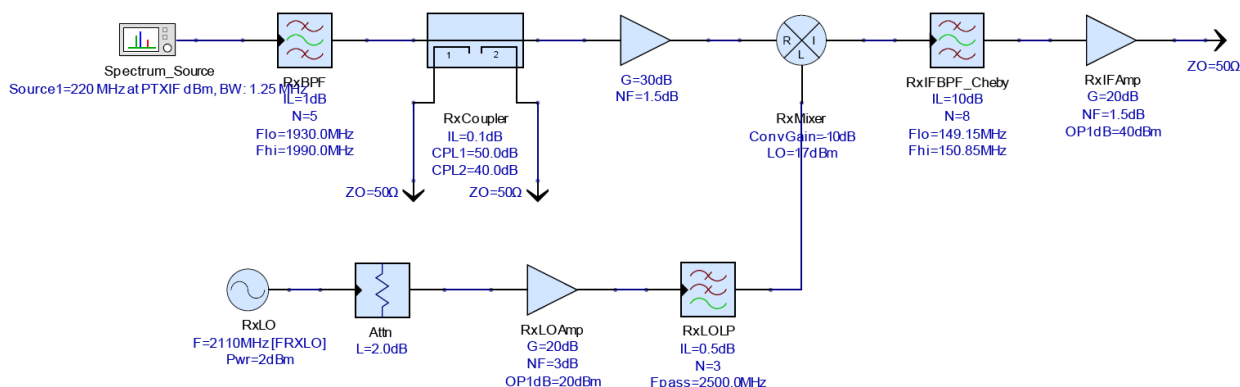


Рис. 1.3. Пример схемы ВЧ приёмника, созданной в SystemVue.

1.3.3 Планировщик частот WhatIF

Планировщик частот WhatIF представляет собой уникальный метод синтеза, используемый для определения полезных и паразитных сигналов ВЧ, ПЧ и гетеродина. Этот метод заменяет «кустарные» средства, а также неудобный метод электронных таблиц, поскольку последний состоит только из аналитических уравнений, программируемых для каждого интересующего вас сигнального тракта. Результаты этого метода весьма ограничены, поскольку он является скалярным и односторонним, а также предполагает наличие нефильтованной и равномерной частотной характеристики. Он не учитывает такие эффекты, как КСВ. По этим причинам планировщик частот WhatIF призван автоматизировать поиск промежуточных частот без паразитных составляющих.

Планировщик частот WhatIF рассчитывает все комбинации гармоник ВЧ сигналов и сигналов гетеродина и показывает их паразитные составляющие, а также свободные от паразитных составляющих участки спектра на одной диаграмме (рис. 1.4). Отображаемая информация включает ширину полосы и амплитуду паразитных составляющих. Планировщик частот WhatIF работает быстро, он очень полезен и удобен для разработчиков систем, позволяя исследовать компромиссы, связанные с используемой системной архитектурой. Он играет важную роль в выборе подходящего частотного плана для проектируемой системы.

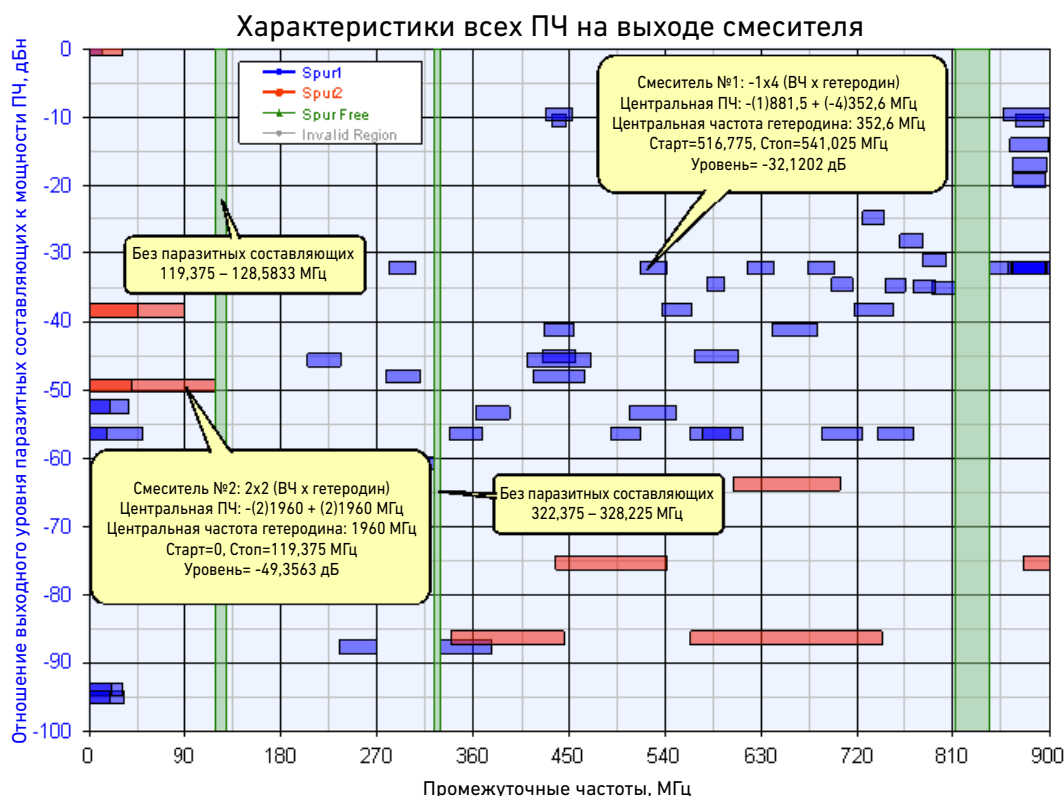


Рис. 1.4. Результаты работы планировщика частот WhatIF для типового ВЧ смесителя.

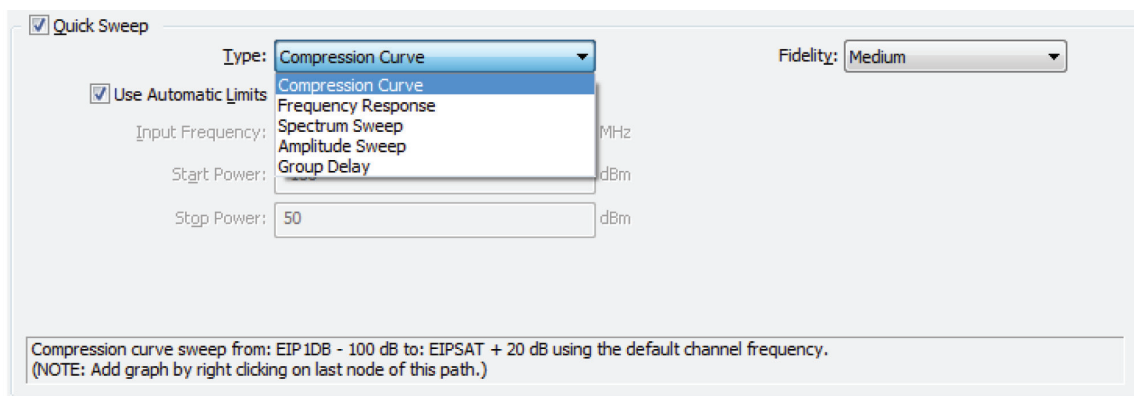
1.3.4 Свипирование по параметрам

Свипирование по параметрам используется для автоматического получения результатов анализа, в зависимости от заданного параметра, перестраиваемого в заданном диапазоне. Как правило, диапазон перестройки задаётся пользователем.

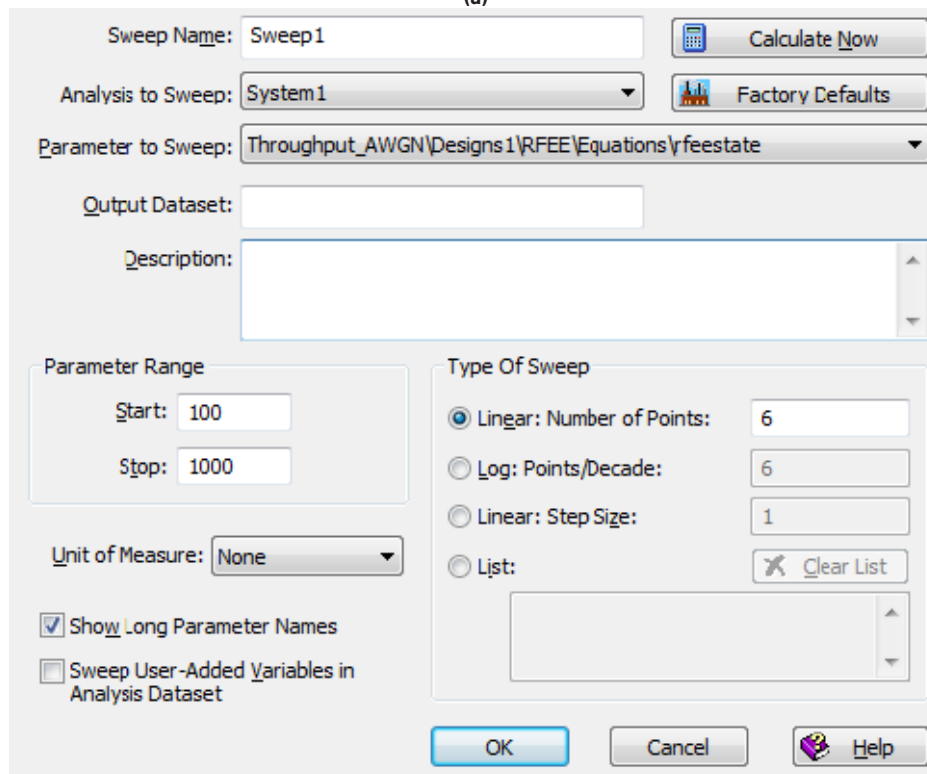
Существуют два способа свипирования по параметрам с помощью SystemVue:

- а) Быстрое свипирование – эта опция проверяет свойства сигнального тракта и позволяет свипировать по нескольким традиционным параметрам, таким как частота, амплитуда, спектр и групповая задержка (рис. 1.5.а).
- б) Свипирование по параметрам – этот метод позволяет перестраивать любой заданный пользователем параметр. Затем рассчитываются результаты анализа для всех указанных пользователем точек (рис. 1.5.б).

Свипирование по параметрам позволяет автоматизировать анализ цепи в разных точках и выполняет оценку отклика в каждой точке. Это помогает оценить характеристики цепи, и изменить её так, чтобы она соответствовала техническому заданию.



(а)



(б)

Рис. 1.5. Свипирование по параметрам: быстрое свипирование (а), свойства свипирования по параметрам (б).

1.3.5 Поведенческая оптимизация

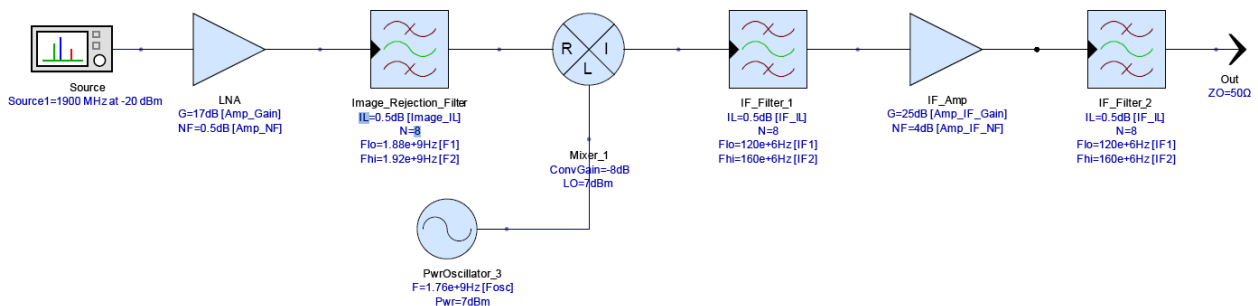
SystemVue позволяет автоматически изменять переменные цепей. В комплект ПО входит инструмент для оптимизации, который можно использовать для автоматического поиска оптимальных значений в широком пространстве переменных. С помощью этого инструмента можно указать заданные значения характеристик (цель оптимизации) и переменные компонентов, которые можно настраивать (рис. 1.6.а и 1.6.б). В ходе процесса оптимизации значения указанных переменных изменяются в соответствии с заданным алгоритмом, и выполняется оперативная оценка функции ошибок для проверки, достигнута ли поставленная цель. Затем автоматически обновляется схема и частотная характеристика цепи (рис. 1.6.в). По завершении процесса оптимизации на диаграмме отображаются старые и новые характеристики, что позволяет принять решение о новой конфигурации цепи (рис. 1.6.г).

General Goals Variables Method								
Default Dataset or Equations: System1_Data								
Use	Measurement	Op	Target	Target Units	Weight	Min	Max	Units
☒	System1_Data_Path1.CNF	<	1.5	dB	5			None
☐		<		None				None

(а)

General Goals Variables Method				
Use	Variable	Min	Max	Units
☒	Equation1.Amp_Gain			None
☒	Equation1.Amp_NF			None
☒	Equation1.Image_IL			None
☒	Equation1.IF_IL			None
☒	Equation1.Amp_IF_Gain			None
☒	Equation1.Amp_IF_NF			None

(б)



(в)

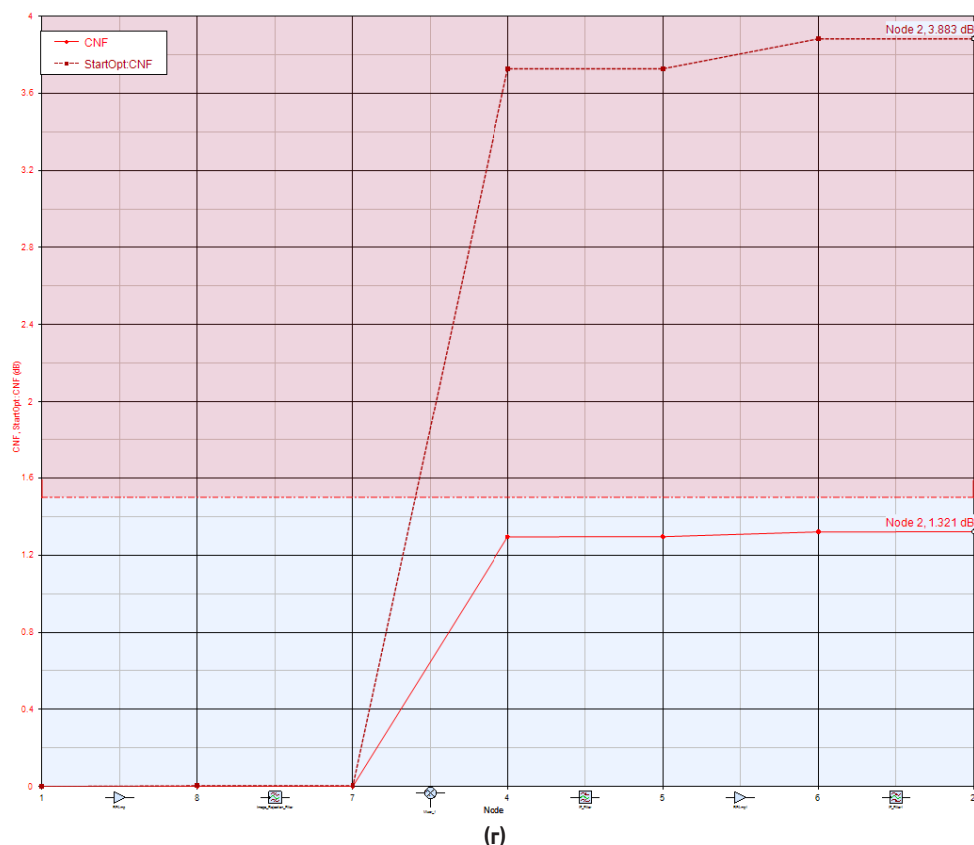


Рис. 1.6. Оптимизация свойств: пример целей оптимизации, например, коэффициент шума нескольких каскадов (а), пример переменных оптимизации (б), пример параметров компонентов, оперативно настраиваемых в ходе оптимизации (в), результаты оптимизации коэффициента шума нескольких каскадов (г).

1.4 Заключение

САПР SystemVue представляет собой передовое ПО компании Keysight, предназначенное для высокоуровневого проектирования и моделирования ВЧ трактов коммуникационных систем. SystemVue позволяет создавать схемы ВЧ и модулирующего сигнала с учетом системных требований. Она объединяет две системы моделирования, позволяющие моделировать произвольные схемотехнические решения. Симулятор Data Flow полностью поддерживает цифровые системы. Он позволяет оценивать ВЧ схемы, состоящие из отдельных ВЧ компонентов. Этот метод моделирования можно использовать совместно с Spectrasys, который является ядром ВЧ моделирования САПР SystemVue. Spectrasys представляет информацию об интермодуляционных составляющих, шумах, гармониках и точках пересечения по интермодуляционным составляющим. Входящие в комплект SystemVue средства оптимизации и настройки помогают находить оптимальные компромиссы. Кроме того, SystemVue содержит планировщик частот WhatIF, который позволяет легко оценивать паразитные составляющие в проектируемой системе. Отображаются свободные от паразитных составляющих диапазоны частот и частоты, подверженные помехам, а также амплитуды различных сигналов. Кроме того, ПО содержит инструменты, которые могут автоматизировать свипирование по параметрам и оптимизацию переменных цепи.

Глава 2. Основы проектирования ВЧ систем

2.1 Введение

В предыдущей главе мы познакомили вас с некоторыми функциями и возможностями интегрированных симуляторов САПР SystemVue. В этой главе мы дадим некоторые необходимые определения и опишем базовые концепции проектирования систем, которые понадобятся далее для иллюстрации возможностей проектирования и моделирования SystemVue.

2.2 Конструкция трансивера

Трансивером называется система радиосвязи, состоящая из приёмника и передатчика. Передатчик преобразует информацию модулирующего сигнала в радиочастотный сигнал и передаёт его по каналу связи, обычно через антенну (рис. 2.1). ВЧ передатчик выполняет три основные функции: модуляция, преобразование частоты и усиление мощности. Типовой передатчик характеризуется тремя основными показателями: точность, излучаемый спектр и выходная мощность.

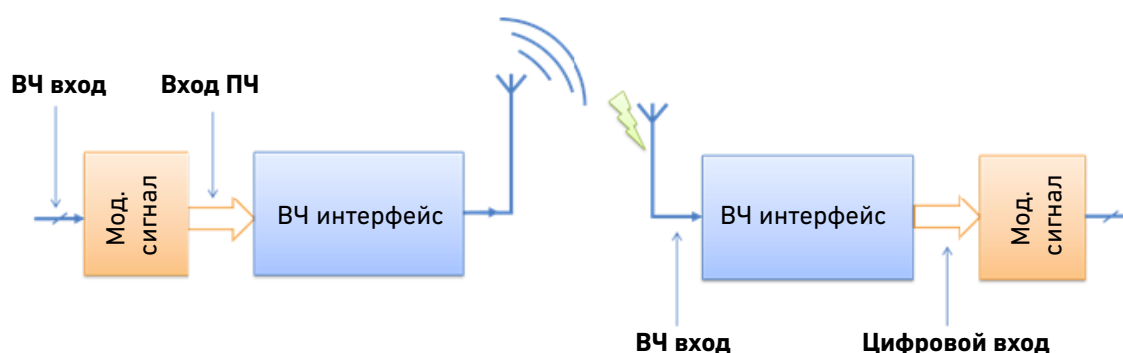


Рис. 2.1. Структурная схема приёмника и передатчика трансивера.

В зависимости от стоимости, исполнения и мощности для проектирования ВЧ передатчика могут использоваться разные архитектуры. На выбор архитектуры передатчика влияет также и число коммуникационных стандартов, которые он должен поддерживать. Кроме того, большинство архитектур передатчика похожи на архитектуры приёмника, например, гетеродинные и гомодинные архитектуры.

Архитектуры передатчиков можно разбить на две основные категории:

- Архитектура со смесителем: к таким архитектурам относятся гетеродинные (т.е. с двойным преобразованием) и гомодинные (т.е. с прямым преобразованием) передатчики. Эти архитектуры могут работать со схемами модуляции с постоянной и непостоянной огибающей, и удобны для работы с несколькими стандартами.
- Архитектуры с фазовой автоподстройкой частоты (ФАПЧ): эти архитектуры более компактны из-за отсутствия дискретных компонентов. Тем не менее, они менее пригодны для работы с несколькими стандартами, так как поддерживают только схемы модуляции с постоянной огибающей.

Архитектуры на основе ФАПЧ обычно используют генератор, управляемый напряжением (ГУН). На рис. 2.2 показана типовая архитектура передатчика с ФАПЧ. В этом передатчике ФАПЧ поддерживает постоянную центральную частоту для точного управления передаваемым сигналом. Подробные примеры такой архитектуры приведены в документах [3], [4] и [5].

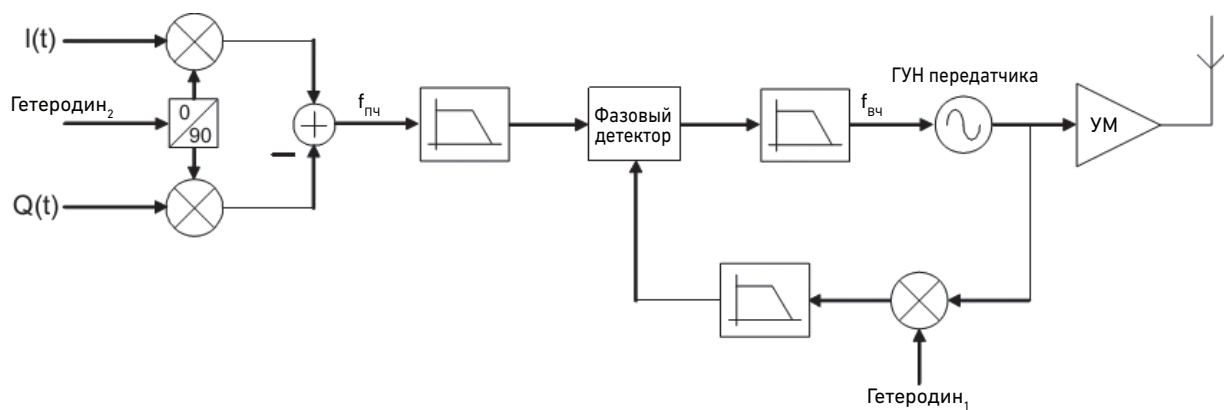


Рис. 2.2. Архитектура с ФАПЧ.³

2.3 Конструкция приёмника

Приёмником называется радиосистема, принимающая некоторую полосу частот и обрабатывающая её для восстановления переданной информации. Основными компонентами приёмника являются антенна, ВЧ интерфейс и демодулятор, в котором выполняется базовая обработка сигнала. Общая структура ВЧ приёмника показана на рис. 2.3.

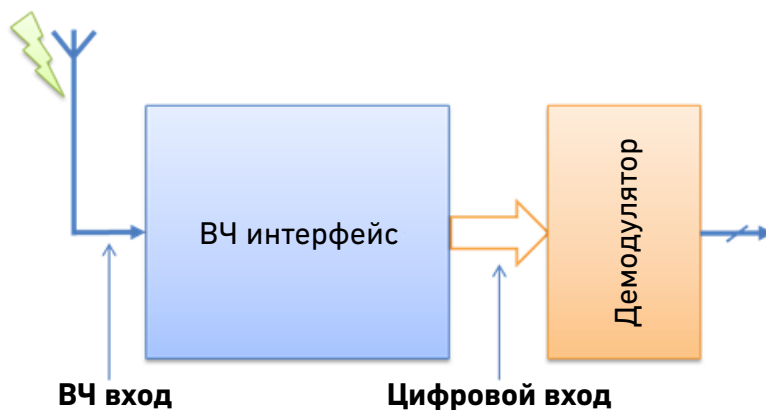


Рис. 2.3. Типовая архитектура приёмника.

За долгие годы разработки ВЧ систем исследовали множество архитектур приёмников. Здесь кратко представлены самые распространённые архитектуры, а более подробное описание можно найти в документе [6].

2.3.1. Супергетеродинные приёмники

Супергетеродинный приёмник (называемый также приёмником двойного преобразования) состоит из двух смесителей, которые понижают частоту сигнала из ВЧ в ПЧ в первом каскаде и из ВЧ в модулирующий сигнал во втором каскаде (рис. 2.4). Для настройки на ВЧ канал и подавления зеркальных составляющих используется несколько фильтров с различными технологиями и характеристиками.

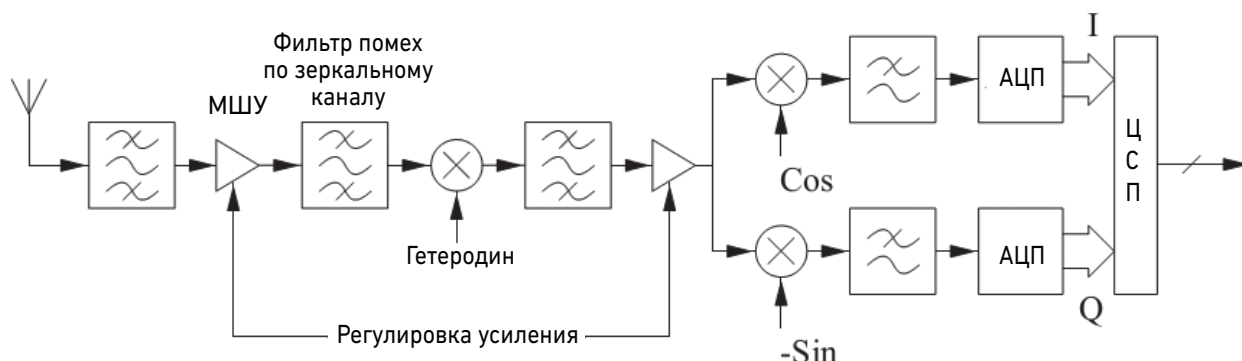


Рис. 2.4. Типовой супергетеродинный приёмник.

2.3.2 Приёмники с низкой ПЧ

Для обеспечения поддержки различных схем модуляции современные разработчики используют в приёмниках архитектуру с низкой ПЧ (рис. 2.5). Эта архитектура отличается переносом ВЧ сигналов на очень низкие промежуточные частоты. С помощью аналого-цифрового преобразователя (АЦП) с высоким разрешением можно непосредственно преобразовать сигнал ПЧ в модулирующий сигнал, который затем подвергается базовой обработке сигнальным процессором.

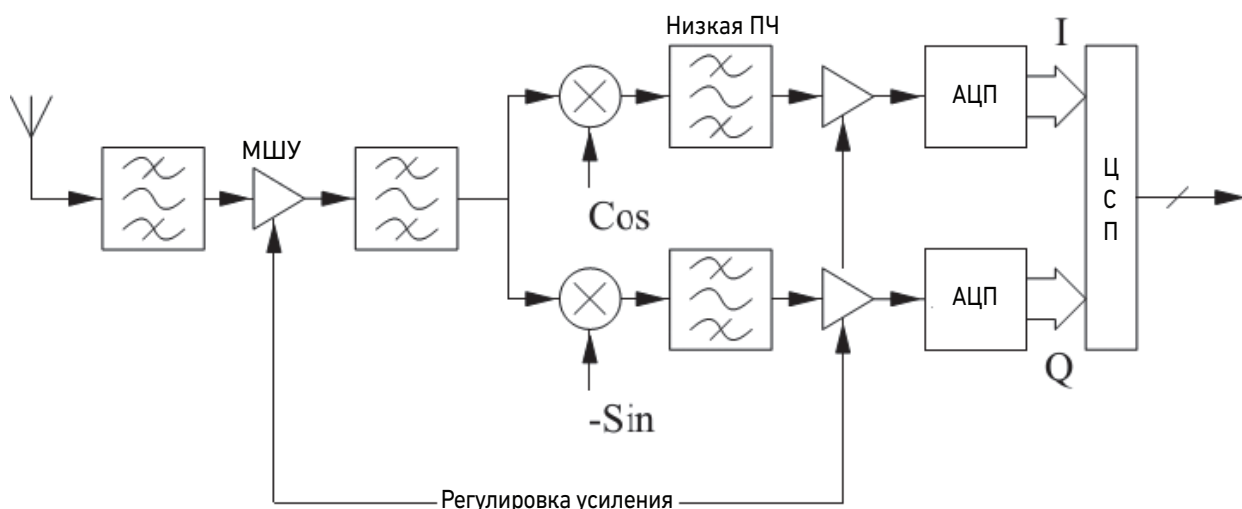


Рис. 2.5. Типовой приёмник с низкой ПЧ.

2.3.3 Приёмники прямого преобразования

Приёмником прямого преобразования (известным также, как приёмник с нулевой ПЧ) называется гомодинный приёмник, который преобразует ВЧ сигнал непосредственно в модулирующий сигнал, содержащий исходную передаваемую информацию (рис. 2.6). При прямом преобразовании ВЧ сигнала в модулирующий сигнал, на выходе смесителя появляется постоянная составляющая. Эту постоянную составляющую (называемую также постоянным смещением) нужно подавить во избежание ухудшения чувствительности демодулятора.

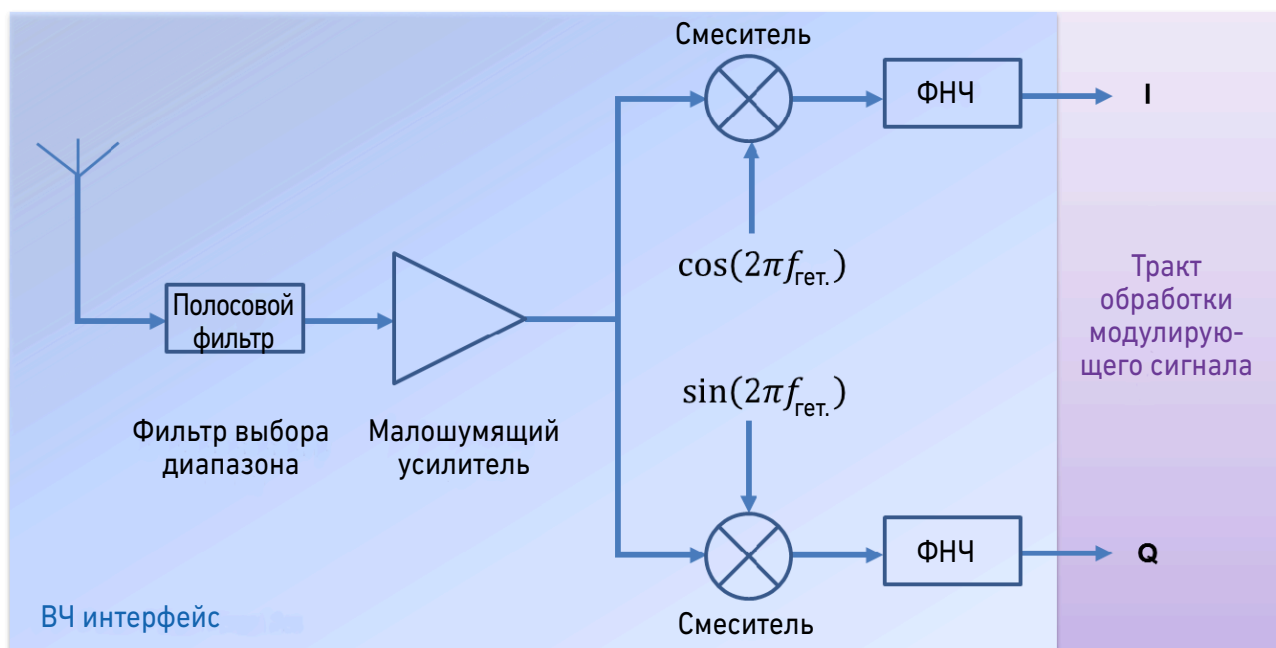


Рис. 2.6. Архитектура типового гомодинного приёмника прямого преобразования.⁷

Подробное описание приёмников прямого преобразования приведено в документе [8].

2.3.4 Приёмники с субдискретизацией

Приёмник с субдискретизацией имеет гомодинную архитектуру, которая выполняет преобразование в ПЧ и затем субдискретизацию, что упрощает применение высокоскоростной цифровой обработки сигнала. В этой архитектуре используются два цифровых смесителя, которые предотвращают возникновение дополнительных шумов и искажений. Преобразование частоты не создаёт постоянной составляющей и шума $1/f$. Оцифровка несущей частоты реализуется интегральными схемами структуры КМОП. Общая схема архитектуры приёмника с субдискретизацией показана на рис. 2.7.

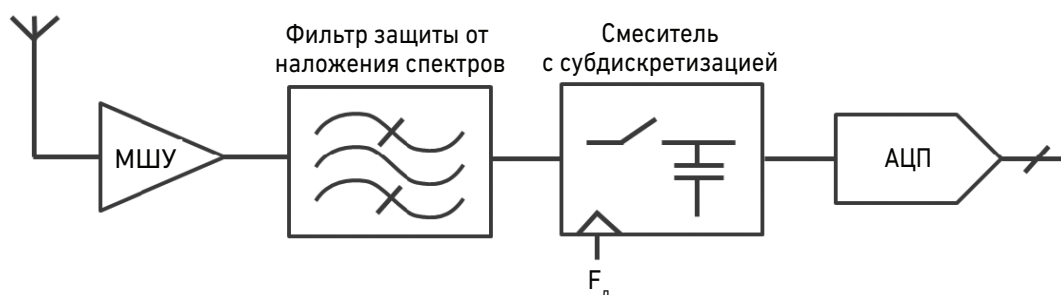


Рис. 2.7. Типовая архитектура приёмника с субдискретизацией: частота дискретизации F_d должна как минимум в два раза превышать частоту несущей.⁹

2.3.5 Сравнение архитектур приёмников

В таблице 2.1, взятой из документа [10], представлено сравнение самых распространённых архитектур приёмников.

Параметр	Гетеродин	Прямое преобразование	Субдискретизация	Низкая ПЧ
Избирательность	Высокая	Высокая	Высокая	Высокая
Аналоговые характеристики	Высокие	Умеренные	Умеренные	Низкие
Гибкость	Низкая	Низкая	Высокая	Высокая
Реализуемость на КМОП	Низкая	Умеренная	Умеренная	Высокая
Шум	Низкий	Умеренный	Высокий	Низкий
Динамический диапазон	Большой	Большой	Большой	Большой

Табл. 2.1. Сравнение широко распространённых архитектур приёмников.

2.4 Конструктивные особенности системного уровня

Проектируя систему радиосвязи, разработчик должен учитывать различные факторы системного уровня.

2.4.1 Показатели качества

Показателями качества систем радиосвязи являются математические критерии, используемые для оценки производительности системы или некоторых её функциональных блоков. Среди этих показателей необходимо упомянуть следующие:

- Коэффициент шума ($K_{\text{ш}}$): описывает уровень шума, создаваемого системой радиосвязи (см. рекомендации по применению Keysight, посвящённые измерению коэффициента шума¹¹).
- Отношение сигнала к шуму (С/Ш): отношение сигнала к уровню шумов, созданных системой радиосвязи.
- Относительный уровень мощности в соседнем канале (ACPR): **отношение общей мощности соседнего канала (интермодуляционная составляющая) к мощности основного канала (полезный сигнал)**.
- Усиление: отношение выходной мощности или амплитуды к входной мощности или амплитуде для данного устройства.
- Точка пересечения по интермодуляционным составляющим 3-го порядка: показатель нелинейности устройств, таких как усилители и смесители, а также коммуникационных систем.
- Точка компрессии на 1 дБ: точка, в которой усиление снижается на 1 дБ.

Некоторые устройства, например усилители, имеют собственные показатели качества (например, КПД, выходной динамический диапазон и устойчивость). В следующем разделе мы представим некоторые показатели качества системного уровня и поясним, как они соотносятся с различными каскадами системы связи.

2.4.2 Уравнения дальности

Канал связи включает весь коммуникационный тракт от источника информации через кодеры и модуляторы передатчика, через канал до приёмника.¹² Для определения бюджета канала, необходимого для надёжной работы системы связи без деградации характеристик канала, нужно учитывать множество параметров. В число этих параметров входит и уравнение дальности, связывающее принимаемую мощность с расстоянием между приёмником и передатчиком.

Оценка фундаментального соотношения мощностей для приёмника и передатчика начинается обычно с предположения о всенаправленном источнике ВЧ сигнала, равномерно передающим сигнал в телесный угол 4π стерадиан. Плотность мощности на расстоянии d определяется функцией $p(d)$, а мощность, извлекаемая приёмной антенной, описывается выражением P_r , где A_{er} представляет собой сечение поглощения (эффективную площадь) приёмной антенны.¹² Концепция уравнения дальности проиллюстрирована рис. 2.8.

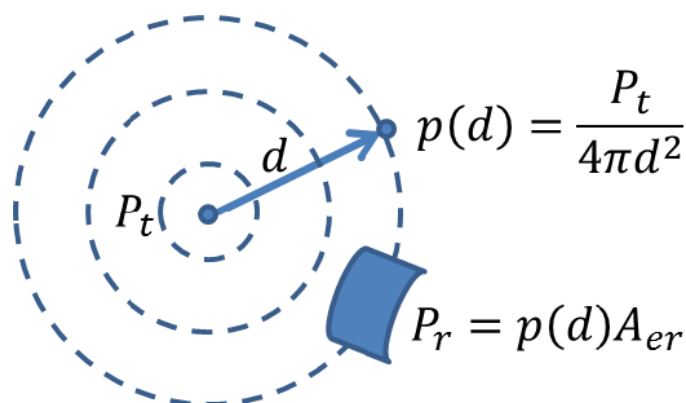


Рис. 2.8. Уравнение дальности выражает зависимость принимаемой мощности от расстояния.¹⁰

2.4.3 Бюджет канала

Как правило, система радиосвязи состоит из передатчика, приёмника и канала передачи (например, для радиосвязи это свободное пространство). Если существует прямой путь между приёмником и передатчиком, он называется линией прямой видимости (рис. 2.9).

Анализ бюджета радиоканала призван ответить на следующие вопросы: «Какой длины может быть канал?» и «Какова его пропускная способность»? На самом деле на характеристики радиосистемы влияет несколько факторов. Эти факторы включают выходную мощность, коэффициент усиления антенны, чувствительность приёмника, условия окружающей среды (например, наличие снега, дождя и особенности рельефа) и многие другие. Бюджет канала определяется тремя основными компонентами (уравнение Е.2.1):

$$\text{Принимаемая мощность (дБм)} = \text{Передаваемая мощность (дБм)} + \text{Усиление (дБ)} - \text{Потери (дБ)} \quad \text{Е.2.1}$$

Усиление складывается, в основном, из усиления антенны и усилителя, тогда как потери состоят из потерь в свободном пространстве, потерь рассогласования, потерь, связанных с условиями окружающей среды (например, дождь и снег), и газового поглощения (например, облака и туман).

В каналах с прямой видимостью основной причиной затухания излучаемого сигнала являются потери в свободном пространстве (FSPL). Они рассчитываются по уравнению Е.2.2:

$$\text{FSPL (дБ)} = 10 \log \left[\frac{4\pi df}{c} \right]^2 \quad \text{Е.2.2}$$

где d – расстояние между приёмником и передатчиком в метрах, f – частота передаваемого сигнала в Гц, c – скорость света в вакууме ($\approx 3 \cdot 10^8$ м/с).

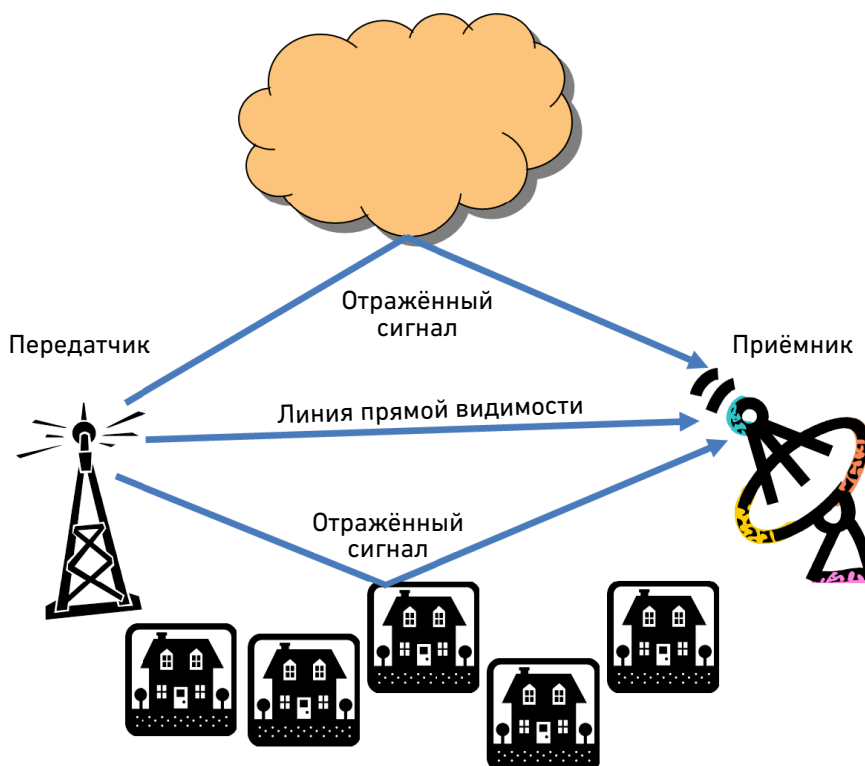


Рис. 2.9. Типовой канал связи.

Канал по линии прямой видимости часто подвержен многолучевому распространению и затуханию. Многолучевое распространение возникает в том случае, если сигнал проходит разными маршрутами и интерферирует в приёмнике. В некоторых случаях сигналы достигают приёмника в противофазе и подавляют друг друга. Затуханием называется подавление передаваемого сигнала случайного характера. Оно может

возникать в результате многолучевого распространения. Для оценки пределов затухания, необходимых для снижения влияния многолучевого распространения, используются такие модели затухания, как модели Накагами и Рэлея. Из-за многолучевого распространения ослабление сигнала может превышать 30 дБ. Таким образом, для преодоления результирующих потерь нужно предусмотреть существенный запас по затуханию (уравнение E.2.3).

$$\text{Запас канала} = \text{Принимаемая мощность} - \text{Чувствительность приёмника} \quad \text{E.2.3}$$

Максимальная пропускная способность, достижимая в данном канале передачи, в общем случае связана с коэффициентом битовых ошибок (BER), который можно получить в приёмнике. BER тесно связан с отношением сигнала к шуму (С/Ш), которое определяется способом модуляции (уравнение E.2.4).

$$\text{С/Ш} = \text{Принимаемая мощность} - \text{Шум канала} \quad \text{E.2.4}$$

Для тех, кто хочет глубже вникнуть в суть вопроса, в документах [13] и [14] приведены подробные примеры расчёта бюджета типового канала с учётом приведённой выше информации.

2.4.4 Чувствительность и избирательность

Чувствительность и избирательность являются основными характеристиками любого радиоприёмника, поскольку они определяют способность приёмника регистрировать требуемый уровень полезных сигналов.

Чувствительность

Общепринятые методы оценки чувствительности приёмника предполагают, что фактором, ограничивающим его чувствительность, является собственный и внешний шум, а не доступный уровень усиления, который можно применить к сигналу. По этой причине для определения чувствительности приёмника используются многие показатели качества. Среди них стоит упомянуть следующие:

- С/Ш
- Отношения полного сигнала к полному уровню помех (SINAD)
- Шум-фактор
- Коэффициент шума
- Отношение несущей к шуму (Н/Ш)
- Минимально различимый сигнал (MPC)
- Коэффициент битовых ошибок (BER)

В общем случае чувствительность определяется как уровень сигнала, необходимый для достижения некоего качества принимаемой информации. В частности, она определяется уровнем мощности, необходимым для достижения требуемого отношения С/Ш.

Избирательность

Избирательность является важным параметром, характеризующим возможность радиоприёмника принимать только полезные сигналы. В зависимости от архитектуры приёмника, избирательность подразумевает способность выбора одной полосы частот или нескольких приёмных каналов. Кроме того, выбранная полоса может быть постоянной или переменной, а также узкой или широкой. В основном, избирательность достигается за счёт применения ВЧ и ПЧ фильтров.

Тип передачи, для которого будет использован приёмник, определяет необходимую полосу пропускания, соответствующую избирательность и тип используемого фильтра. Например, избирательность определяет, насколько будут подавляться внеполосные сигналы, но никак не влияет на внутриполосные помехи.

2.4.5 Подавление зеркальных частот

В гетеродинных приёмниках ВЧ сигнал преобразуется в более низкую ненулевую промежуточную частоту, на которой выполняется обработка сигнала. Как показано на рис. 2.10, такой архитектуре присуща врожденная проблема, которая называется зеркальной частотой. Она возникает при смешивании полезного сигнала с другим сигналом, расположенным по другую сторону от частоты гетеродина с такой же отстройкой от неё. Зеркальная частота может значительно искажать полезный сигнал, особенно, если её мощность превышает мощность полезного сигнала.

Для решения этой проблемы в приёмный тракт до каскада смешения можно добавить фильтр, подавляющий зеркальную частоту. Кроме того, проблему зеркальной частоты можно решить, используя специальную схему смесителя, такую как Хартли или Уивера.

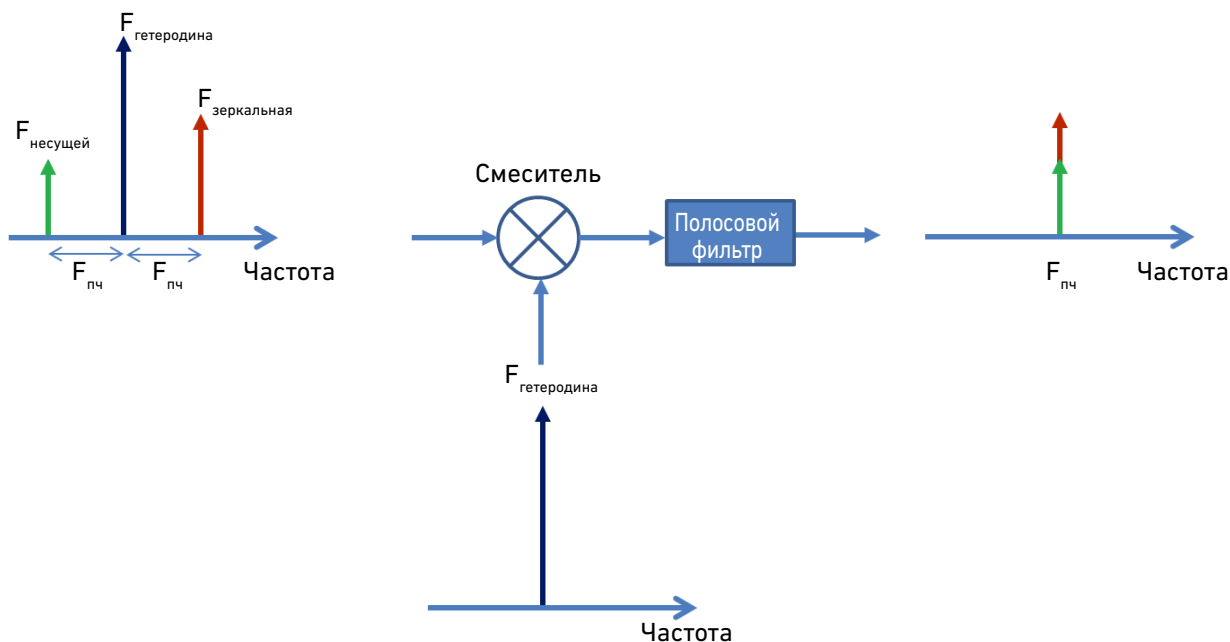


Рис. 2.10. Работа смесителя в типовом гетеродинном приёмнике.

2.4.6 Фазовый шум

Фазовый шум описывает случайные флуктуации частоты гетеродина. Когда эти флуктуации воздействуют на смеситель, с обеих сторон от несущей частоты возникают боковые полосы шума (рис. 2.7а). Боковые полосы шума обычно измеряются в дБн/Гц при заданной отстройке от частоты несущей (рис. 2.7б). Фазовый шум является источником паразитных сигналов, создаваемых им в полосе пропускания. Кроме того, он создаёт серьёзные проблемы в системах OFDM (мультиплексирование с ортогональным делением частот), где он нарушает ортогональность сигналов. Подробное описание фазового шума приведено в документе [15].

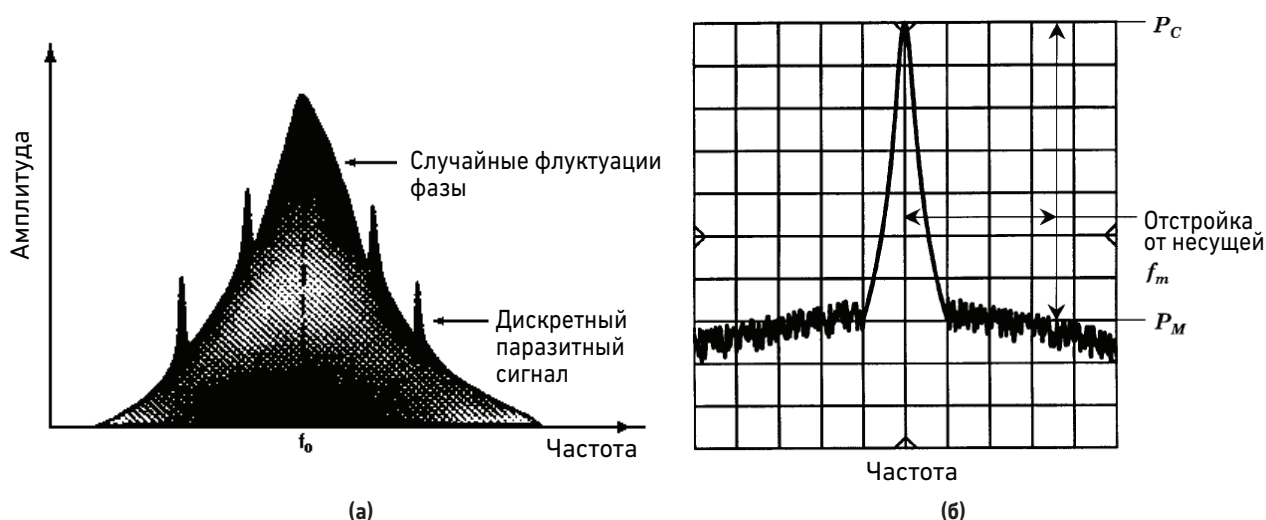


Рис. 2.11. Фазовый шум. Отображение фазового шума на экране анализатора спектра (а). Две боковые полосы фазового шума вокруг несущей частоты (б).

2.4.7 Постоянное смещение

Постоянное смещение проявляется в основном в схемах с прямым преобразованием, где нежелательный сильный сигнал может преобразоваться вместе с полезным сигналом прямо в демодулированный сигнал. Этот эффект называется «самосмещением». В этом случае большой уровень преобразованного нежелательного сигнала создаёт сильные помехи полезному сигналу.

Эффект постоянного смещения может порождаться тремя основными механизмами⁷:

1. Утечка частоты гетеродина. Плохая развязка между гетеродином и малошумящим усилителем, особенно на кремниевых подложках, может привести к утечке сигнала гетеродина через ВЧ порт и его самосмещению с основным сигналом гетеродина. Ситуация становится ещё хуже, если прежде чем достичь ВЧ входа смесителя, сигнал гетеродина усиливается малошумящим усилителем.
2. Переизлучение гетеродина. В некоторых случаях сигнал гетеродина может переизлучаться антенной и отражаться от препятствий. Переизлучённый сигнал принимается вместе с полезными сигналами и смешивается с ними. В этом случае переизлучённый сигнал влияет не только на локального пользователя, но и на других пользователей, использующих те же каналы.
3. Сильная внутриполосная помеха. Сильная близлежащая помеха тоже может генерировать большое постоянное смещение, которое может исказить принимаемую информацию.

Кроме того, постоянное смещение может порождаться внутриполосными нелинейными составляющими второго порядка, генерируемыми ВЧ интерфейсом приёмника.

2.5 Нелинейное поведение ВЧ системы

Нелинейная ВЧ система генерирует выходной сигнал, который не является линейной функцией входных сигналов (рис. 2.12). Предположим, что: $V_{вх} = A \cos(\omega_1 t) + A \cos(\omega_2 t)$.

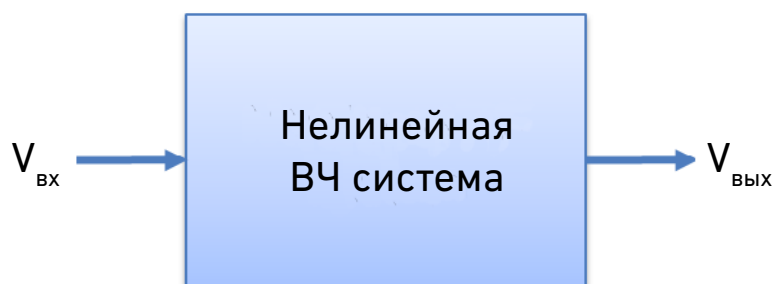


Рис. 2.12. Схема нелинейной системы.

Результирующий выходной сигнал будет определяться уравнением E.2.5:

$$V_{\text{вых}} = aA(\cos(\varpi_1 t) + \cos(\varpi_2 t)) - bA^2(\cos(\varpi_1 t) + \cos(\varpi_2 t))^2 - bA^3(\cos(\varpi_1 t) + \cos(\varpi_2 t))^3 + \dots \quad \text{E.2.5}$$

2.5.1 Интермодуляционные искажения

Интермодуляционные искажения – это эффект, присутствующий в нелинейных коммуникационных системах. Интермодуляционные искажения возникают, когда два и более сигналов занимают один и тот же канал передачи. На выходе смесителя нелинейность проявляет себя в виде двойных составляющих основной частоты, которые называются интермодуляционными искажениями. Такие нежелательные сигналы могут вызывать блокировку приёмного тракта.

На рис. 2.13 показаны интермодуляционные составляющие первого, второго и третьего порядка для двухтональных несущих. Частоты интермодуляционных составляющих двухтональных несущих можно рассчитать по формуле E.2.6:

$$mf_2 \pm nf_1, \text{ где } m, n = 0, 1, 2, 3... \quad \text{E.2.6}$$

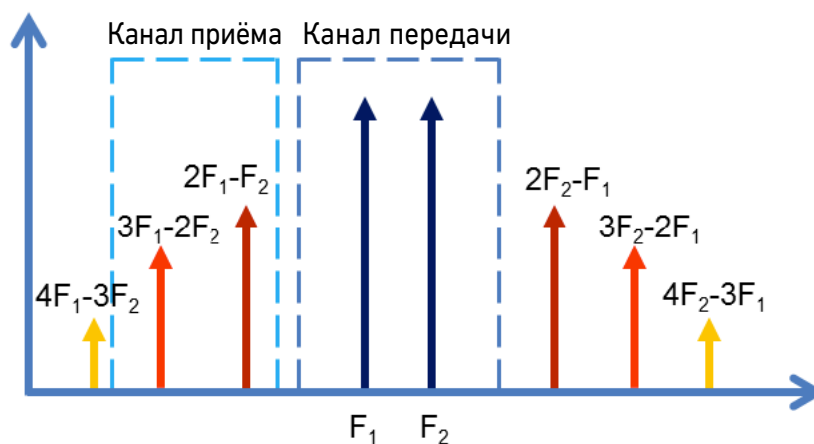


Рис. 2.13. Интермодуляционные составляющие.

2.5.2 Коэффициент шума

Коэффициентом шума определенного устройства называется мера ухудшения отношения С/Ш, вызванного этим устройством в тракте передачи и/или приёма. Его можно рассчитать по шум-фактору F с помощью уравнения E.2.7.

Шум-фактор представляет собой отношение входного С/Ш к выходному С/Ш. Коэффициент шума равен разности между входным и выходным С/Ш, как показано в уравнении E.2.8.

$$F = C/Ш_{\text{вх}} / C/Ш_{\text{вых}} \quad \text{E.2.7}$$

$$K_{\text{ш}} = 10 \log(F) = C/Ш_{\text{вх}}, \text{ дБ} - C/Ш_{\text{вых}}, \text{ дБ} \quad \text{E.2.8}$$

Каждое ВЧ устройство характеризуется коэффициентом шума, поскольку его электрическая схема добавляет шум к полезному сигналу и может иметь собственное усиление (рис. 2.14). При каскадном включении нескольких компонентов результирующий коэффициент шума рассчитывается по формуле Фриза, представленной уравнением E.2.9. Более подробная информация приведена в документе [11].

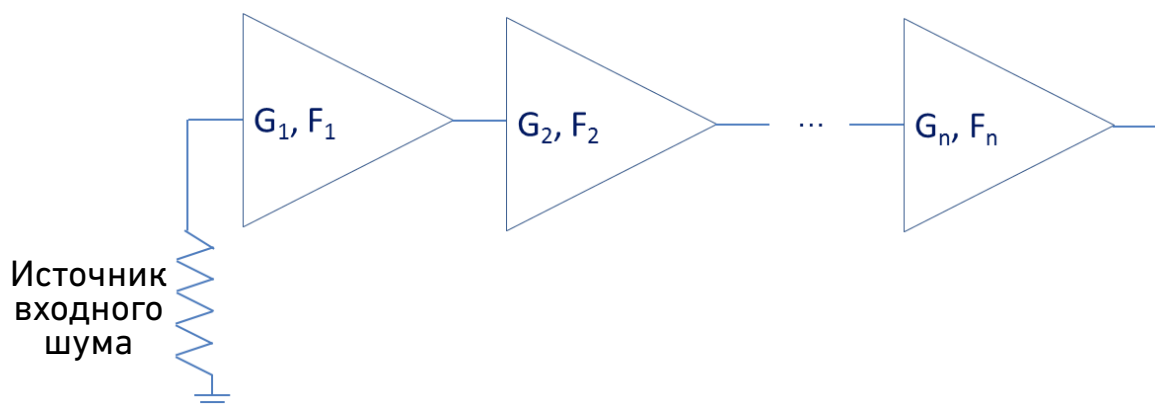


Рис. 2.14. Коэффициент шума для нескольких последовательных каскадов в ВЧ тракте.

$$F = F_1 + \frac{F_2 - 1}{G_1} + \frac{F_3 - 1}{G_1 G_2} + \frac{F_4 - 1}{G_1 G_2 G_3} + \frac{F_n - 1}{G_1 G_2 G_3 \dots G_{n-1}} \quad \text{E.2.9}$$

2.5.3 Внутриполосные и внеполосные помехи и блокирующие сигналы

Каждый передатчик создаёт в каждом приёмнике некоторые помехи. С этими помехами нужно бороться, чтобы предотвратить потерю чувствительности приёмников. Как описано в документе [16], такие помехи можно разбить на две основные категории:

- Внутриполосные помехи: порождаются перекрытием частот приёмника и передатчика. Оба устройства могут работать совершенно нормально, но только если они разнесены на относительно большое расстояние, когда мешающие сигналы передатчика достаточно сильно затухают и не подавляют полезные сигналы.
- Внеполосные помехи: этот тип помех не связан с перекрытием частот приёмника и передатчика. Такие помехи порождаются во-первых, внутриполосными излучениями, принимаемыми из внеполосного диапазона из-за несовершенства фильтров и аттенюаторов приёмника, и во-вторых, внеполосными излучениями, принимаемыми внутри полосы, из-за несовершенства и нелинейности передатчика.

2.5.4 Соседние каналы

Соседними каналами называются ближайшие частотные диапазоны, расположенные выше и ниже по частоте относительно рассматриваемого канала (рис. 2.15). На эти каналы может влиять расширение спектра, а также гармонические и интермодуляционные составляющие, нарушая передачу других пользователей. Поэтому этим побочным эффектам нужно уделять особое внимание.

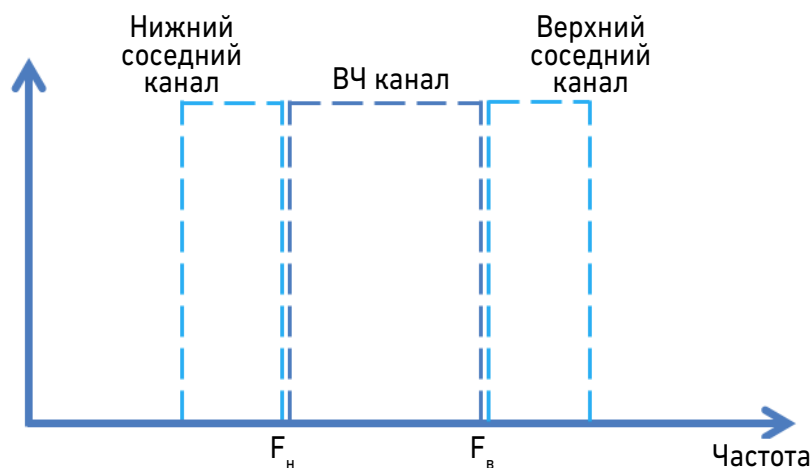


Рис. 2.15. Верхний и нижний соседние каналы.

Относительным уровнем мощности в соседнем канале (ACPR) называется уровень помех или мощности в соседнем частотном канале. Обычно ACPR определяется, как отношение средней мощности в соседнем частотном канале (или при определённой отстройке) к средней мощности в передающем частотном канале. ACPR каскадного включения можно рассчитать по следующей формуле: ¹⁷

$$ACPR_{\text{выхода}} = 10 \log \left[\left(\sum_{k=1}^n \left[10^{\frac{ACPR_k}{10}} \right]^{\frac{1}{2}} \right)^2 + 10^{\frac{ACPR_{\text{выхода}}}{10}} \right] \quad \text{E.2.10}$$

Концепция ACPR иллюстрируется рис. 2.16 (а) и (б).

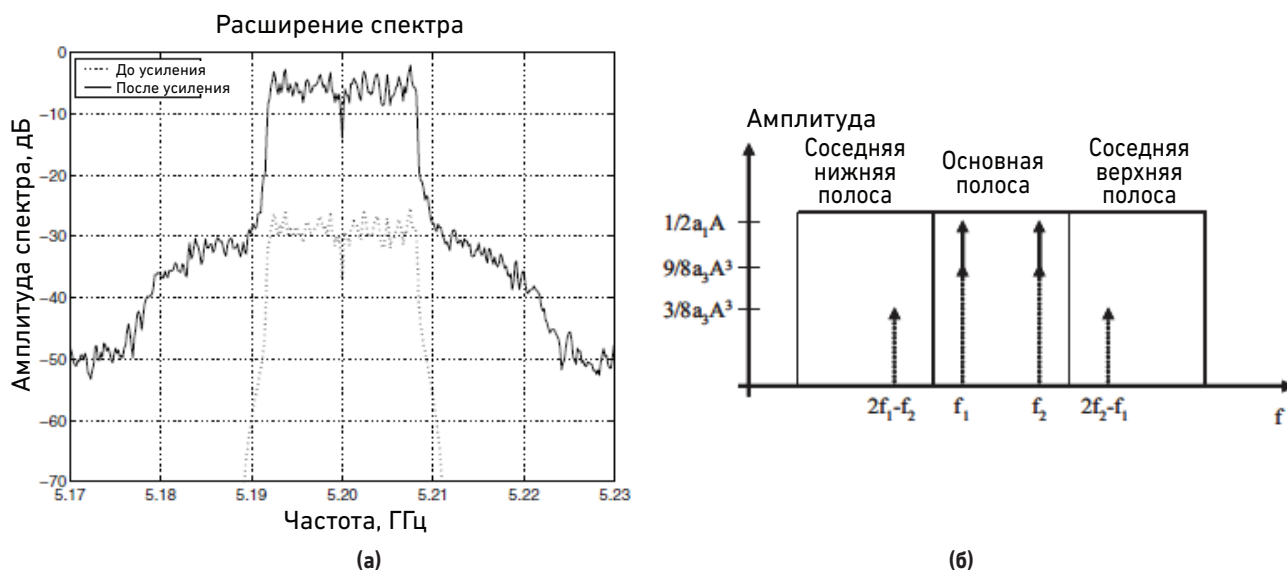


Рис. 2.16. Расширение спектра, вызванное усилением сигнала OFDM (а), внутриполосная и внеполосная интермодуляция: ACPR равен разности между мощностью основного внутриполосного тона и интермодуляционными составляющими в соседнем канале (б).¹⁷

2.5.5 Компрессия усиления

Компрессией усиления называется явление, проявляющееся в нелинейных ВЧ системах. В линейных системах любое изменение входной мощности вызывает пропорциональное изменение выходной мощности. Однако в нелинейных системах выходная мощность линейно зависит от входной мощности только до определенного предела – насыщения. В этой точке выходная мощность перестаёт линейно зависеть от входной.

Усиление

Усилением называется отношение выходной мощности или амплитуды к входной. Обычно оно обозначается буквой G и измеряется в децибелах.

$$G_{дБ} = 10 \log(P_{\text{вых}}/P_{\text{вх}}) \quad \text{E.2.11}$$

Усиление нескольких каскадов равно линейной сумме усиления каждого каскада:

$$G_{\text{tot}, дБ} = \sum_{i=1}^n G_{i, дБ} \quad \text{E.2.12}$$

Точка пересечения по интермодуляционным составляющим третьего порядка

Точка пересечения по интермодуляционным составляющим третьего порядка (обычно обозначается как **IP3** или **TOI**) является характеристикой нелинейных устройств и систем. Она равна отношению нелинейных

составляющих третьего порядка к линейно усиленному сигналу. Точка пересечения по интермодуляционным составляющим второго порядка ($IP2$) подобна $IP3$, но учитывает составляющие второго порядка. При каскадном включении $IP3$ можно рассчитать по формуле Е.2.13, взятой из документа [17]:

$$\frac{1}{IP3_{tot}} = \frac{1}{IP3_n} + \frac{1}{G_n IP3_{n-1}} + \frac{1}{G_{n-1} IP3_{n-2}} + \dots + \frac{1}{G_n} + \dots + G_2 IP3_1 \quad \text{Е.2.13}$$

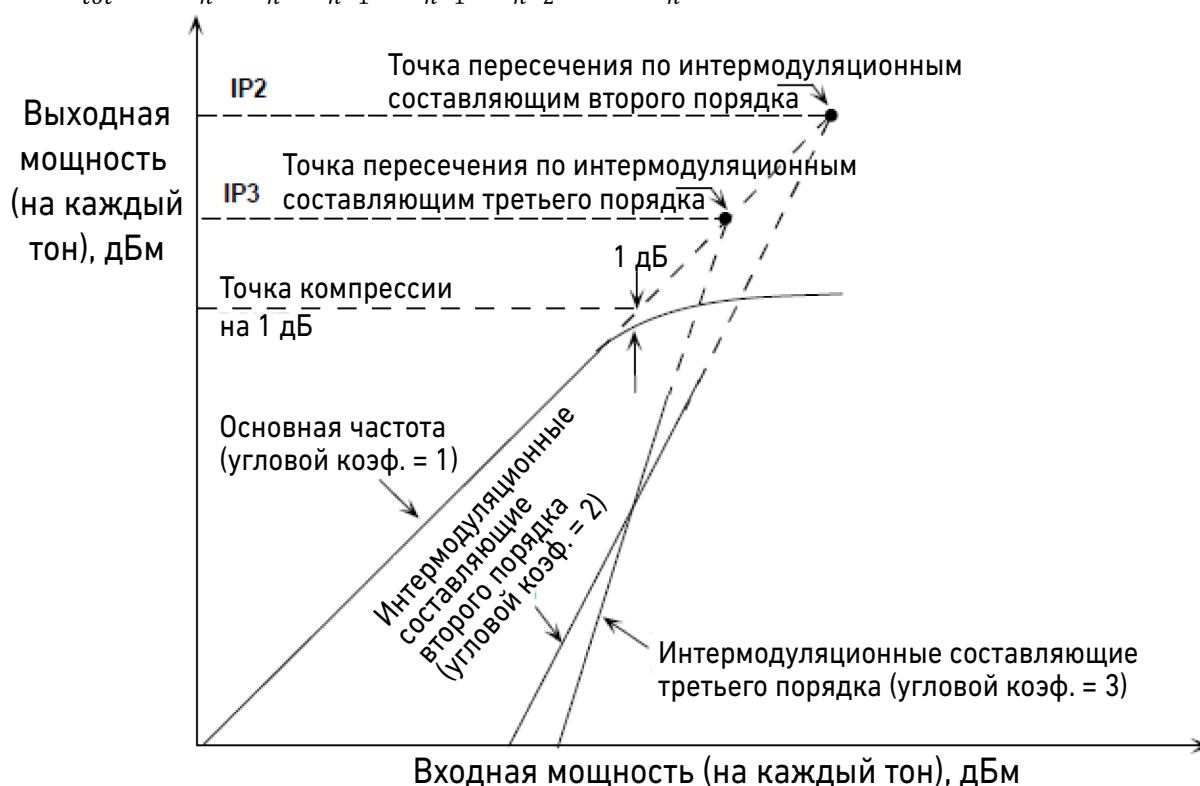


Рис. 2.17. Точка пересечения по интермодуляционным составляющим третьего порядка.

❖ Точка компрессии на 1 дБ

Обычно точка компрессии на 1 дБ обозначается как $P_{1дБ}$ и определяет входную мощность нелинейной системы, при которой усиление малого сигнала отклоняется от прямой линии и уменьшается на 1 дБ. Определение точки компрессии на 1 дБ нелинейной системы проиллюстрировано на рис. 2.17. Для нескольких каскадов $P_{1дБ}$ можно рассчитать рекурсивно по формуле Е.2.14.

$$P_{1дБ, \text{вых.}} = \frac{1}{10 \log \left[\frac{1}{G_n P_{1дБ, n-1}} + \frac{1}{P_{1дБ, n}} \right]} \quad \text{Е.2.14}$$

2.5.6 Динамический диапазон

Динамический диапазон (ДД) равен полезному диапазону сигнала, который приёмник может принять и корректно обработать. В общем случае динамический диапазон определяется как разность между точкой компрессии на 1 дБ и уровнем собственных шумов для данного каскада (Е.2.15).

$$ДД = P_{1дБ} - MPC \quad \text{Е.2.15}$$

2.6 Компромиссы при конструировании передатчика

2.6.1 Отношение сигнала к шуму

Как следует из уравнения Е.2.16 и показано на рис. 2.18, отношение сигнала к шуму (обозначаемое как С/Ш) определяет отношение уровня полезного сигнала к уровню собственных шумов. Оно равно отношению интегральной мощности сигнала к интегральной мощности шума в интересующей нас полосе.

$$C/Ш_{дБ} = 10 \log[P_{\text{сигнала}}/P_{\text{шума}}] = P_{\text{сигнала}}, \text{ дБ} - P_{\text{шума}}, \text{ дБ} \quad \text{Е.2.16}$$

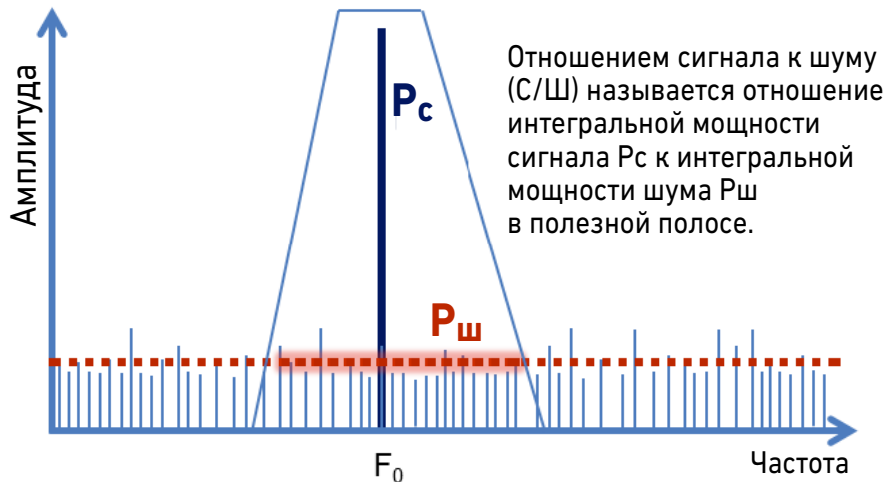


Рис. 2.18. Отношение сигнала к шуму.

2.6.2 Планирование частот

В абонентском терминале передатчик использует для передачи сигналов канал, называемый восходящим, а приёмник использует другой канал, называемый нисходящим (или прямым каналом). Полосы восходящего и нисходящего каналов как правило равны между собой (рис. 2.19). Для организации связи нужно выбрать пару из нисходящего и восходящего каналов. Для каждого коммуникационного стандарта определяется полоса восходящего и нисходящего канала, разнесение каналов и полоса разделения. Например, в сотовой связи CDMA¹⁷ восходящие и нисходящие каналы занимают диапазоны 824 – 849 МГц и 869 – 894 МГц соответственно, причем полоса разделения равна 20 МГц, а разнесение каналов составляет 30 кГц.

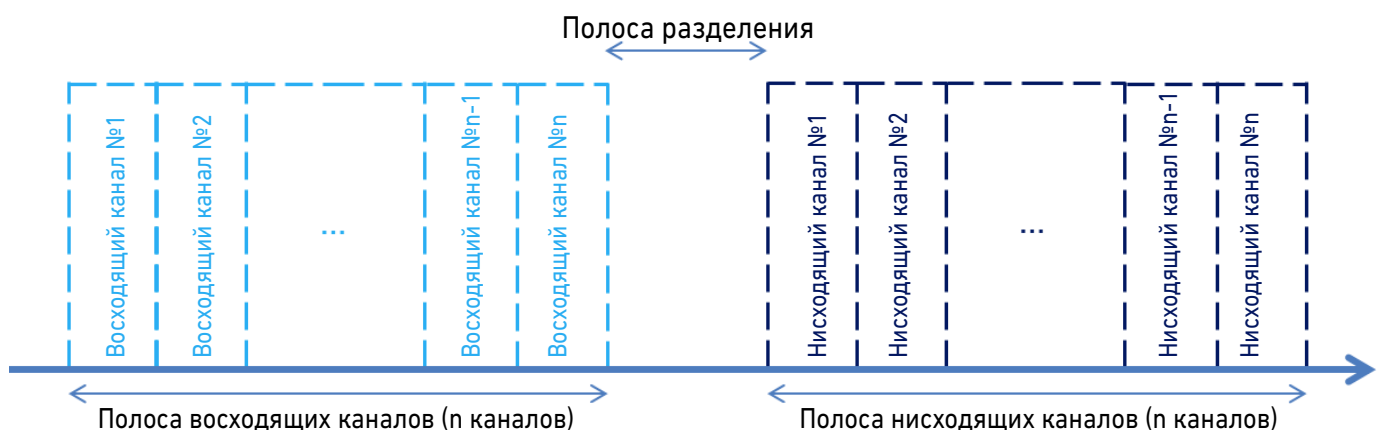


Рис. 2.19. Распределение восходящих и нисходящих каналов и полосы разделения.

Распределение частот в ВЧ устройствах оказывает решающее влияние на планирование частот, которое заключается в оптимальном выборе промежуточных частот, при которых паразитные составляющие в супергетеродинных приёмниках будут минимальны. Эта проблема дополнительно усложняется, если трансивер использует полнодуплексную схему, в которой сигналы приёмника и передатчика могут создавать гармоники и оказывать отрицательное воздействие друг на друга.

Для минимизации искажений выбор промежуточной частоты (ПЧ) должен опираться на два основных правила¹⁷:

- Правило 1. Если приёмник и передатчик используют общий гетеродин, то для улучшения характеристик ПЧ_{пр} и ПЧ_{пер} должны выбираться согласно уравнению E2.17:

$$ПЧ_{пр} = ПЧ_{пер} + DF_{пр-пер},$$

где $F_{зем} > F_{пр} > F_{пер}$
или $F_{пер} > F_{пр} > F_{зем}$

E2.17

- Правило 2. Для предотвращения возникновения внутриполосных помех в приёмнике, ПЧ_{пр} должно соответствовать уравнению E2.18:

$$ПЧ_{пр} > B_{пер} + B_s + B_{пр}$$

или $ПЧ_{пр} < B_s$

E2.18

Для проверки частотного плана нужно выполнить анализ паразитных сигналов и измерить гармоники, попадающие в полезный диапазон. Для сигналов основной частоты этот расчёт выполняется достаточно просто:

$$f_s = m \cdot f_A \pm n \cdot f_B,$$
E2.19

где А и В – сигналы основной частоты.

2.7 Основные функциональные блоки ВЧ трансивера

Как описывалось выше, входные и выходные тракты всех приемников и передатчиков состоят из различных ВЧ блоков, которые выполняют базовые аналоговые функции, такие как преобразование частоты, выбор частоты и усиление мощности. В этом разделе мы рассмотрим самые важные ВЧ блоки. Более подробную информацию можно найти в документах [19] и [20].

2.7.1 Антенна

Антенна является интерфейсом между каналом связи (свободным пространством) и входными цепями приёмника. Передатчику нужна согласующая схема на выходе усилителя мощности, обеспечивающая максимальную мощность излучения в канал связи. Приёмнику нужен фильтр для выбора полосы и малошумящий усилитель для точного выбора нужного канала и усиления сигнала до различного уровня. Наиболее важными характеристиками антенны являются: коэффициент усиления, диаграмма направленности, потери на излучение, резистивные потери, полоса пропускания и КПД. Более подробная информация об антеннах приведена в документах [21], [22] и [23].

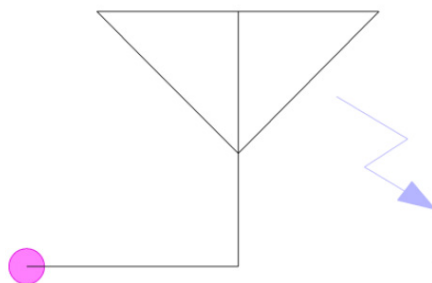


Рис. 2.20. Условное графическое обозначение антенны.

2.7.2 Усилитель (мощности/малошумящий)

Усилителем называется устройство, которое повышает уровень мощности сигнала. В передатчике оно называется усилителем мощности, поскольку используется для усиления сигнала до максимального значения, при котором сохраняется линейность на выходе. Однако в приёмнике усилитель нужен для усиления сигнала до приемлемого значения, при котором возможна обработка с минимальным уровнем шумов, позволяющим избежать деградации качества модулированных данных (EVM). Усилители являются нелинейными устройствами, поэтому нужно уделять особое внимание их нелинейному поведению. К наиболее распространённым параметрам усилителей относятся: коэффициент усиления, коэффициент шума, компрессия усиления, интермодуляционные и нелинейные искажения. Подробный анализ ВЧ усилителей выполнен в документах [24] и [25].

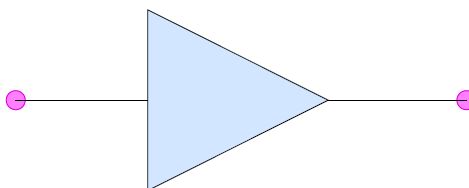


Рис. 2.21. Условное графическое обозначение усилителя.

2.7.3 Фильтр

Фильтрами называются устройства, которые пропускают только нужный диапазон частот, а нежелательные сигналы значительно ослабляют. Идеальный фильтр пропускает полезные сигналы без потерь и полностью подавляет нежелательные сигналы. Однако большинство реальных фильтров имеет небольшое затухание в полосе пропускания (т.е. в полезном диапазоне частот) и достаточно большое затухание в полосе ослабления (т.е. в нежелательном диапазоне). Самыми важными характеристиками фильтра являются затухание в полосе пропускания и в полосе ослабления, а также полоса пропускания и групповая задержка. В зависимости от задачи могут использоваться разные типы фильтров. Фильтры нижних и верхних частот пропускают частоты, расположенные ниже или выше заданной частоты среза. Полосовой фильтр пропускает частоты в диапазоне между двумя заданными частотами. Действие режекторного фильтра противоположно — он ослабляет частоты в диапазоне между двумя заданными частотами.

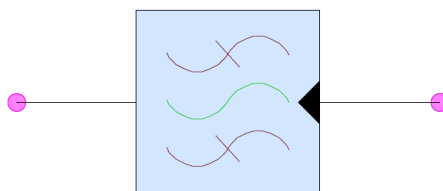


Рис. 2.22. Условное графическое обозначение фильтра.

Фильтры достаточно подробно описаны в литературе по радиоэлектронике. Дополнительную информацию можно найти в документах [26] и [27].

2.7.4 Дуплексер

Дуплексером называется блок фильтров, используемый только в системах FDMA с одной антенной, в которых приёмник и передатчик работают одновременно. Дуплексер используется для предотвращения потери чувствительности приёмника и утечки принимаемого сигнала на выход передатчика, изолируя каскад приёма от каскада передачи с минимальными потерями в полезной полосе. Важным параметром дуплексера являются вносимые потери. Более подробную информацию можно найти в документе [28].

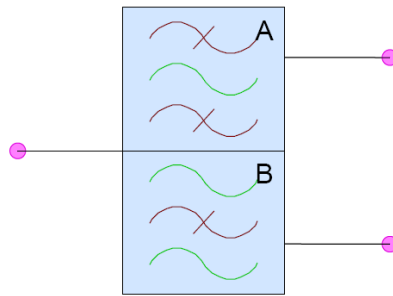


Рис. 2.23. Условное графическое обозначение дуплексера.

2.7.5 Смеситель

Смесителем называется нелинейное устройство, выполняющее преобразование частоты. Смеситель используется для наложения модулирующего сигнала на ВЧ несущую или для преобразования ВЧ сигнала в ПЧ. При смешении двух сигналов смеситель генерирует верхнюю и нижнюю боковые полосы (т.е. сумму и разность входных частот). Для выделения нужной полосы используется фильтр нижних или верхних частот. Обычно на вход смесителя подаётся сигнал гетеродина, который позволяет выполнять понижающее или повышающее преобразование частоты.

Для построения смесителей применяются разные технологии: смеситель можно выполнить на диодах или транзисторах. В обоих случаях это устройство является нелинейным. Его нелинейное поведение нужно контролировать (например, нужно знать точку компрессии на 1 дБ, интермодуляционные искажения и т.п.). Подробнее смесители описаны в документах [20] и [29].

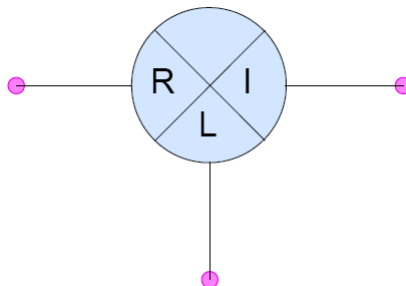


Рис. 2.24. Условное графическое обозначение смесителя.

2.7.6. Гетеродин

Гетеродином называется устройство, генерирующее опорный сигнал, который подаётся на один из входов смесителя и используется для повышения или понижения частоты. На выходе гетеродина присутствует сигнал большого уровня, который подаётся на смеситель и позволяет генерировать сигнал основной частоты и различные гармоники.

Обычно гетеродин характеризуется такими параметрами, как фазовый шум, диапазон перестройки и стабильность частоты. Подробная информация приведена в документах [30] и [31].

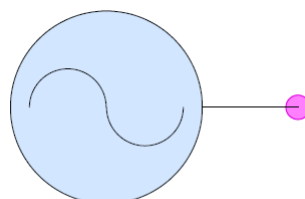


Рис. 2.25. Условное графическое обозначение генератора.

2.7.7 Детектор

Детектором называется устройство, восстанавливающее форму принятого сигнала. В зависимости от типа модуляции детектор может использоваться для восстановления фазы или амплитуды сигнала. Подробная информация приведена в документах [32] и [33].

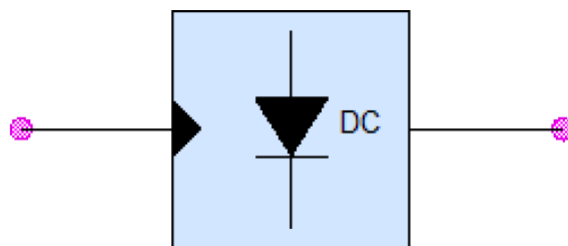


Рис. 2.26. Условное графическое обозначение детектора.

2.7.8 Аналого-цифровой преобразователь

Аналого-цифровой преобразователь (АЦП) является наиболее важным компонентом современных радиостанций. Он преобразует аналоговый сигнал в цифровые данные, которые можно использовать для обработки сигнала. Типовой АЦП содержит четыре основных узла:

1. Предварительный фильтр: сглаживающий фильтр, предотвращающий наложение спектров высокочастотных сигналов.
2. Схема выборки и хранения: поддерживает стабильный уровень входного аналогового сигнала во время его преобразования в эквивалентный цифровой код. Этот период называется временем преобразования.
3. Дискретизатор: разбивает опорный уровень на подуровни. Обычно существуют 2^N подуровней, где N равно числу двоичных разрядов выходного кода. Дискретизация заключается в выборе подуровня, соответствующего значению выборки входного аналогового сигнала.
4. Кодер: цифровой препроцессор кодирует соответствующие цифровые разряды в зависимости от результатов дискретизации.

Подробная информация об АЦП приведена в документах [34] и [35].

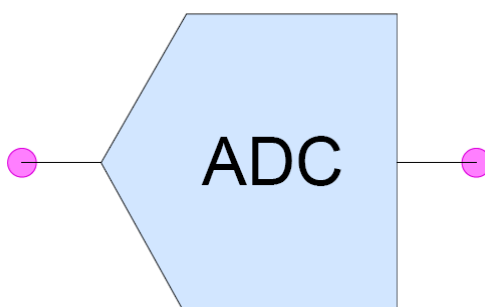


Рис. 2.27. Условное графическое обозначение аналого-цифрового преобразователя.

2.8 Заключение

В этой главе мы описали основы проектирования ВЧ устройств и, в частности, подробно рассмотрели аспекты системного уровня, которые должны учитывать разработчики ВЧ систем. Эта глава не даёт глубоких знаний по ВЧ проектированию, однако читатель может познакомиться с литературой, ссылки на которую приведены в этой главе, и узнать больше о представленных понятиях и концепциях. И, наконец, рекомендации по применению 1312³⁶ компании Keysight дают практический обзор большинства концепций, упомянутых в этой главе.

Глава 3. Проектирование ВЧ систем с помощью САПР SystemVue

Изучение на примере проектирования входных цепей устройства LTE

3.1 Введение

В главе 1 мы описали некоторые функции и возможности САПР SystemVue. В главе 2 мы представили обзор основ ВЧ устройств. В этой главе мы сначала опишем общие подходы к проектированию ВЧ устройств, а также принципы проектирования системного уровня, а затем подробно рассмотрим возможности SystemVue на примере проектирования и анализа трёхдиапазонного сканирующего приёмника LTE.

3.2 Методы и подходы к проектированию

В общем случае радиосистема состоит из двух основных блоков: тракта модулирующего сигнала, который отвечает за основную обработку сигнала, и высокочастотного интерфейса, который обеспечивает взаимодействие по радиоканалу с другими системами с помощью радиоволн (например, с другими радиостанциями или базовыми станциями). Процесс проектирования типовой радиосистемы включает этапы анализа, реализации, интеграции и проверки НЧ и ВЧ подсистем (рис. 3.1).



Рис. 3.1. Процесс проектирования типовой радиосистемы делится обычно на процессы проектирования двух подсистем: тракта модулирующего сигнала и ВЧ тракта.

Основное внимание будет уделено проектированию ВЧ системы.

3.2.1 Общие подходы к проектированию

В разные годы ВЧ системы проектировались по-разному. Каждая методология влияла на продуктивность, надёжность конструкции и процесс проверки. Тремя самыми распространёнными подходами к проектированию ВЧ систем были следующие:

1. Проектирование снизу вверх. В этом традиционном подходе процесс проектирования начинается с разработки отдельных блоков. Затем эти блоки объединяются в более сложные подсистемы.

мы, которые будут частью конечной системы. Такой подход может быть очень эффективным для малых систем, но может порождать серьёзные проблемы в больших системах. На этом уровне проверка процессов становится сложнее, а устранение ошибок может оказаться очень дорогим³⁷. Моделирование высокого уровня может отнять много времени. Например, моделирование одного усилителя мощности будет занимать несколько секунд, но анализ BER всей системы может занять целый час.

2. Проектирование сверху вниз. В этом подходе система определяется на уровне структурной схемы. Затем она оптимизируется, и формулируются требования к нижележащим блокам. После этого реализуются и тестируются отдельные блоки, чтобы на последующих этапах объединить их в полную ВЧ систему. Зачастую этот подход используется в том случае, когда проектирование системы начинается с исследования коэффициента усиления, коэффициента шума и бюджета искажений. С самого начала можно определить важные компромиссы, что позволяет добиться соответствия спецификациям и повышает эффективность проектирования. Однако такой подход может привести к рассогласованию процесса проектирования из-за несоответствия системной архитектуры и её реализации. Такое рассогласование замедляет процесс проектирования³⁷.
3. Проектирование в зависимости от характеристик/ограничений. В этом подходе проектирование выполняется сверху вниз, а проверка – снизу вверх. На каждом уровне система разделяется на подсистемы, которые нужно реализовать. Затем собранные блоки проверяются на соответствие спецификациям. Такой подход помогает разработчикам выявлять ошибки на ранних этапах процесса проектирования³⁸.

Эта справка об общих подходах к проектированию ВЧ систем поможет разработчикам адаптировать процесс разработки к требованиям проекта к продуктивности, эффективности и взаимодействию между системным инженером и разработчиками отдельных компонентов.

3.2.2 Обзор моделирования на системном уровне

Как правило, ВЧ системы состоят из подсистем или модулей, связанных между собой и реализующих функциональность системы. Эти модули состоят из схем и компонентов низкого уровня, таких как транзисторы, линии передачи и дискретные элементы. Затем для описания ВЧ системы в общем виде определяются три уровня абстрагирования (рис. 3.2)³⁹:

- Системный уровень. На этом уровне ВЧ система считается «чёрным ящиком», состоящим из набора функциональных блоков, обеспечивающих нужную функциональность (например, приёмник с низкой ПЧ). Система описывается общими параметрами, такими как покаскадный коэффициент шума и покаскадный коэффициент усиления.
- Уровень модуля/блока. Блок (называемый также модулем) является составной частью системы. Он обладает определенной функциональностью, которая помогает построить функциональность системы. Например, фильтр является блоком, выделяющим заданный диапазон частот. Приёмник содержит разные фильтры для выбора нужного диапазона. Для расчёта коэффициента шума всего приёмника используются вносимые потери фильтра, поскольку коэффициент шума этого фильтра оказывает влияние на коэффициент шума приёмника.
- Уровень схемы/элемента. На уровне схемы учитываются физические характеристики блока. Затем блок рассматривается как набор цепей и элементов, взаимодействующих друг с другом для реализации функциональности блока. Например, типовой усилитель состоит из транзисторов, линий передачи и дискретных элементов (например, конденсаторов и резисторов). Каждый элемент обладает своим набором параметров, таких как рабочий ток, напряжение и сопротивление. В сочетании с параметрами других элементов и в зависимости от способа монтажа этих элементов, эти параметры могут по-разному влиять на параметры блока.



Рис. 3.2. Уровни абстрагирования ВЧ систем.

На системном уровне блоки обычно выполняют определенные функции и описываются такими параметрами, как коэффициент усиления, коэффициент шума и вносимые потери. Эта модель захватывает характеристики системы и позволяет эффективно исследовать пространство проектирования, маскируя лишние параметры, связанные обычно с физической реализацией⁴⁰. Такой подход обладает двумя важными преимуществами.

1. Он хорошо подходит для проектирования сверху вниз, когда разработчик начинает с выбора соответствующей архитектуры для проектируемой системы.
2. Он позволяет проанализировать выбранную архитектуру и решить, соответствует ли она заданным требованиям. Если нет, то, возможно, нужно выполнять оптимизацию или предложить другую архитектуру.

САПР SystemVue компании Keysight представляет собой средство моделирования, которое позволяет моделировать на системном уровне и анализировать блоки ВЧ и модулирующего сигнала. Чтобы помочь разработчикам в создании архитектуры ВЧ системы, SystemVue включает набор моделей и инструментов, которые позволяют выполнять анализ, оценку и оптимизацию ВЧ архитектур на системном уровне. Эта САПР обладает мощными возможностями, которые помогают разработчикам оценивать линейные и нелинейные характеристики ВЧ систем. Например, она позволяет анализировать широкополосный шум, рассчитывать интермодуляционные искажения и выбирать ПЧ для предотвращения внутриполосных и внеполосных помех. Подробную информацию по этому вопросу можно найти в главе 1, а также в меню и учебных материалах самой программы.

3.3 Изучение на примере проектирования трёхдиапазонного приёмника LTE

В первом разделе этой главы мы дали обзор подходов к ВЧ проектированию и моделированию системного уровня. В этом разделе мы представим пример проектирования приёмника LTE с помощью САПР SystemVue. Основная цель этого изложения – показать, как моделирование системного уровня помогает в ВЧ проектировании, и как SystemVue помогает анализировать и оптимизировать реальные ВЧ архитектуры.

3.3.1 Обзор LTE

Проект долговременного развития (LTE) третьего поколения (3G), организованный партнёрским проектом третьего поколения (3GPP), нацелен на предоставление услуг сотовой связи, соответствующих растущим требованиям к скорости передачи данных и малым задержкам в сотовых сетях связи. Для достижения высоких скоростей передачи и надёжности, LTE использует разные технологии.

- OFDM (мультиплексирование с ортогональным делением частот). Технология OFDM обеспечивает широкую полосу передачи данных и обладает высокой устойчивостью к отражениям и помехам. В ней используются два типа методов доступа: OFDM для нисходящего канала и SC-FDMA (множественный доступ с частотным разделением и одной несущей) для восходящего.
- MIMO (несколько входов, несколько выходов). MIMO используется для решения таких проблем, как многолучевое распространение и затухание, а также для повышения пропускной способности. Для разделения разных сигналов, поступающих в приёмник разными путями, используется несколько антенн.
- SAE (развитие системной архитектуры). Высокая скорость передачи данных и малые задержки в LTE достигаются за счёт вынесения некоторых функций за пределы сети на периферию для повышения производительности.

Кроме того, LTE использует традиционные схемы доступа TDD (дуплексный режим с временным разделением) и FDD (дуплексный режим с частотным разделением).

Спецификации радиointерфейса LTE определены проектом 3GPP в документе 41 для абонентского оборудования и базовых станций⁴². Подробная информация об LTE приведена в документах [43], [44], [45] и [46].

3.3.2 Определение технических характеристик и проектирование приёмника

В документе [47] представлен недорогой сканирующий приёмник для систем LTE TDD. Разработчики приёмника использовали для анализа и моделирования своей архитектуры САПР ADS компании Keysight. Приёмник поддерживает три диапазона LTE (34, 39 и 40 диапазоны E-UTRA⁴¹) и использует для выбора нужного диапазона два переключателя. Однако предлагаемый приёмник рассчитан на работу только в диапазоне от 2320-2370 МГц (в третьем диапазоне) и не охватывает 40^{ой} частотный диапазон (2300-2400 МГц). По всей видимости, этот компромисс связан с выбором стандартной промежуточной частоты 140 МГц. В дальнейшем мы будем рассматривать весь 40^{ой} диапазон, и используем возможности SystemVue, чтобы проиллюстрировать разные методы анализа системного уровня.

Технические характеристики приёмника

Технические характеристики приёмника LTE приведены в таблице 3.1.

Рабочие диапазоны	Диапазон 1: 1880 – 1920 МГц (диапазон E-UTRA №39)
	Диапазон 2: 2010 – 2025 МГц (диапазон E-UTRA №34)
	Диапазон 3: 2300 – 2400 МГц (диапазон E-UTRA №40)
Модуляция	QPSK
Скорость кодирования	1/3
Полоса канала	10 МГц
Конфигурация антенны	Один вход, один выход (SISO)
Тип канала	Аддитивный белый гауссовский шум (AWGN)
С/Ш на входе	10 дБ
Относительная пропускная способность	≥ 95 %
Число блоков физических ресурсов (PRB)	50
Схема дуплексирования	TDD

Табл. 3.1. Технические характеристики приёмника LTE.

Выбор ВЧ архитектуры

Мы начали проектирование приёмника LTE с выбора ВЧ архитектуры, которая будет использоваться для его реализации. В настоящем примере мы выбрали традиционную супергетеродинную архитектуру с двумя промежуточными частотами. Первая ПЧ (ПЧ1) реализована типовым смесителем в ВЧ интерфейсе приёмника, а вторая (ПЧ2) формируется в демодуляторе (рис. 3.3).

	Нижняя частота F_L , МГц	Верхняя частота F_H , МГц	Центральная частота F_0 , МГц	Полоса канала BW, МГц
Диапазон 1	1880	1920	1900	40
Диапазон 2	2010	2025	2017,5	15
Диапазон 3	2300	2400	2350	100

Табл. 3.2. Диапазоны, поддерживаемые приёмником LTE

Чтобы показать, как работает планировщик частот WhatIF, мы рассмотрели свободные от паразитных составляющих области, в которых можно выбрать ПЧ только для первого диапазона. Как показано на рис. 3.5, существует несколько свободных от паразитных составляющих областей (в зелёных столбцах), в которых можно выбрать ПЧ. Самая нижняя область лежит между 75 и 347 МГц.

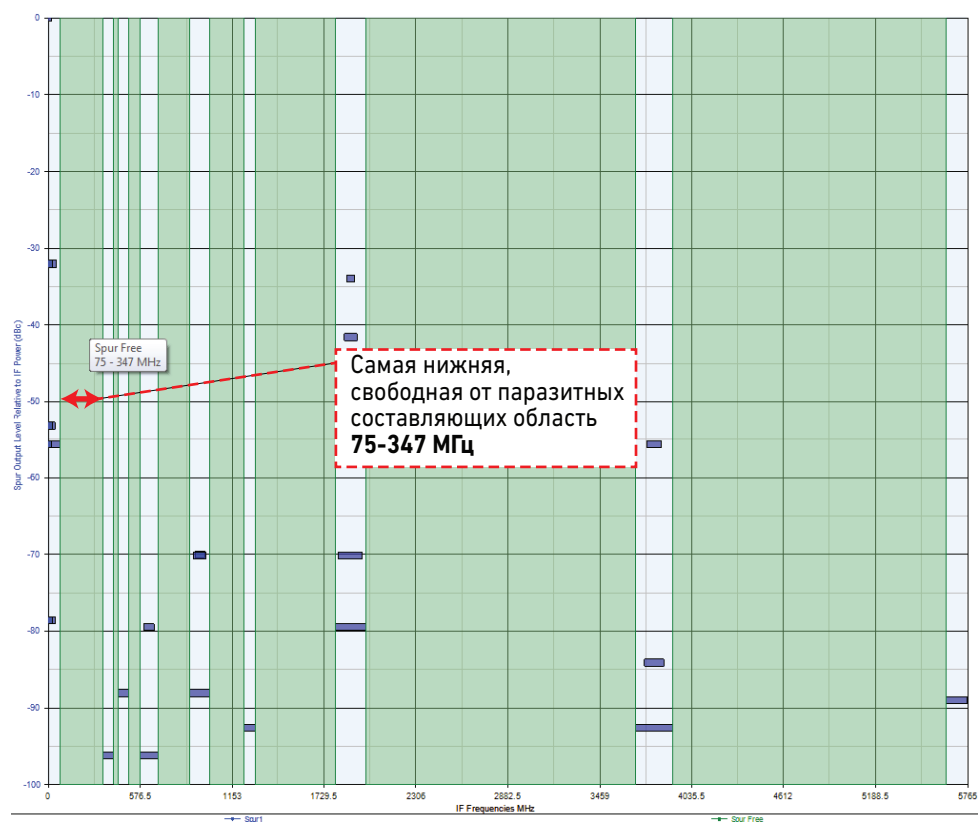
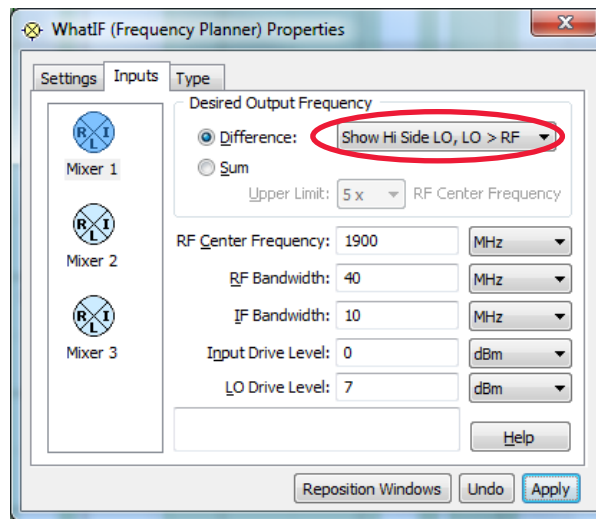
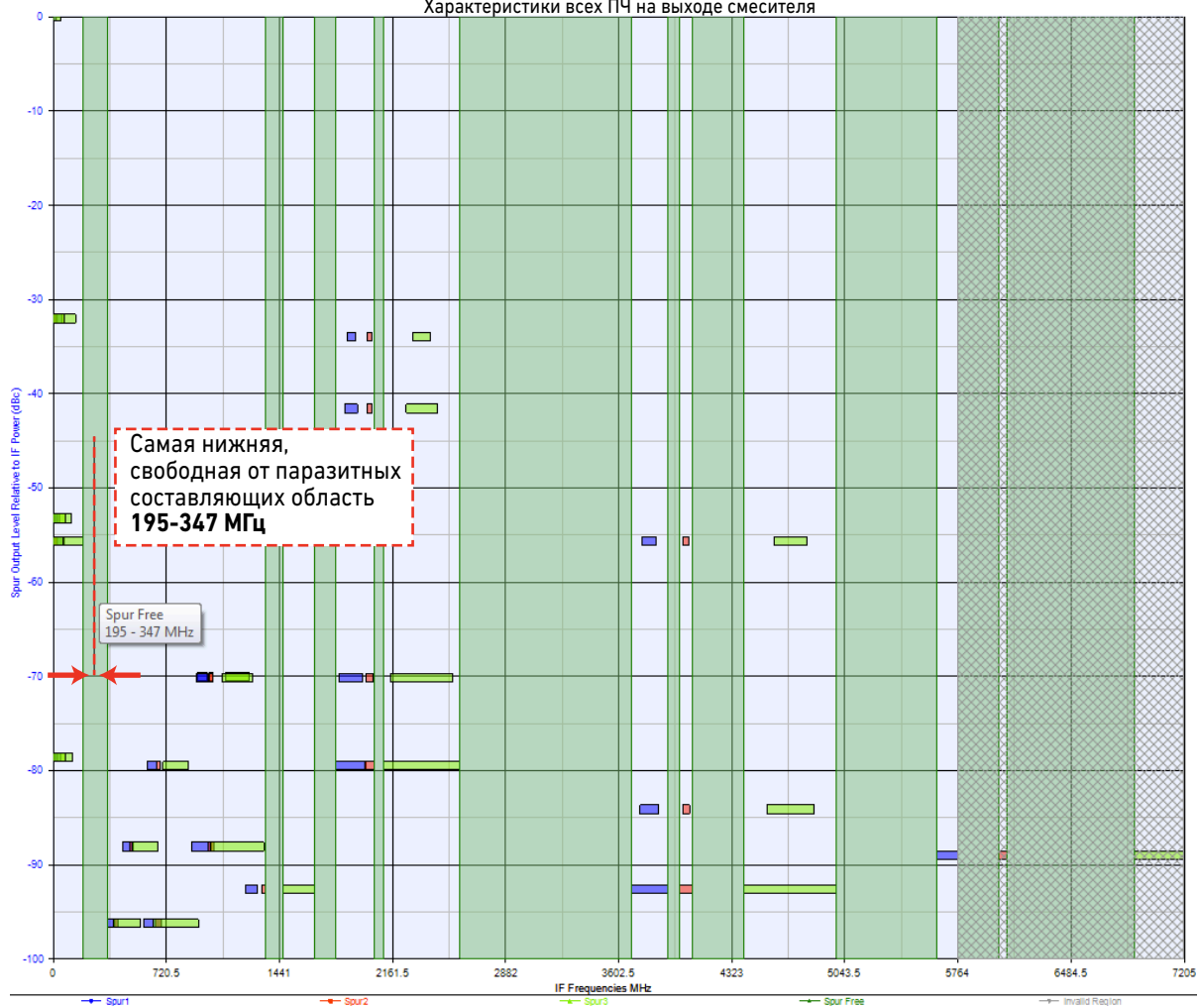


Рис. 3.5. Выходная частота смесителя для диапазона 1.



(a)

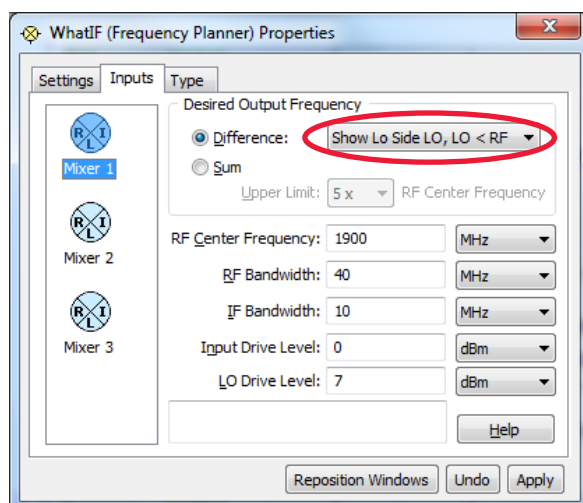
Характеристики всех ПЧ на выходе смесителя



(б)

Рис. 3.6. Все ПЧ для диапазонов 1, 2 и 3 (частота гетеродина выше ВЧ): планировщик частот WhatIF (а), результирующая карта ПЧ (б).

Однако нам нужно выбрать ПЧ, которая подходила бы для всех трёх диапазонов. С помощью планировщика частот WhatIF мы можем искать ПЧ для пяти разных диапазонов.



(a)

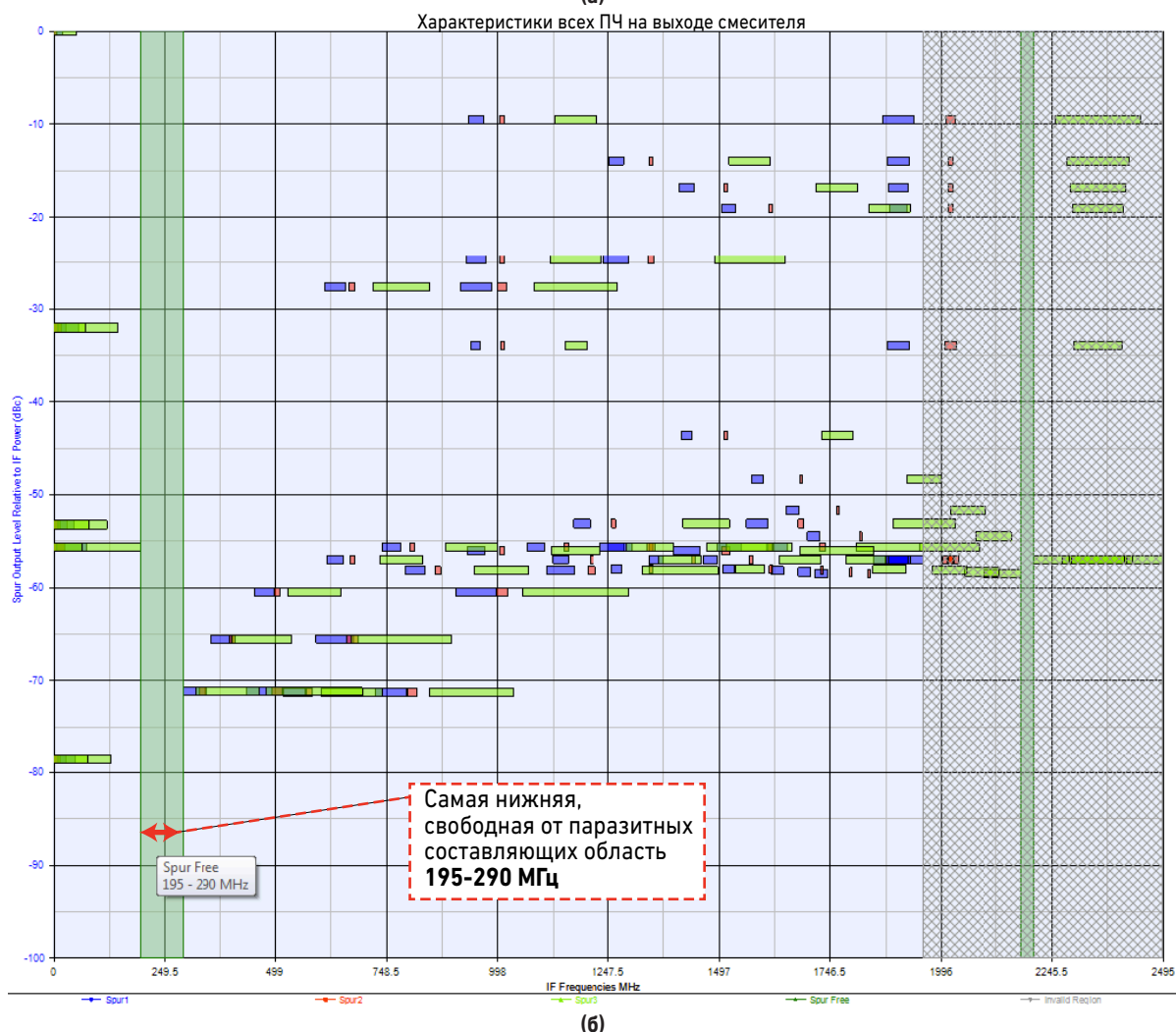


Рис. 3.7. Все ПЧ диапазоны 1, 2 и 3 (частота гетеродина ниже ВЧ): планировщик частот WhatIF (a), результирующая карта ПЧ (б).

Имеются две возможные конфигурации: смесители, у которых частота гетеродина выше частоты ВЧ сигнала, и смесители, у которых частота гетеродина ниже частоты ВЧ. Самая нижняя, свободная от паразитных составляющих область на результирующей карте лежит между 195 и 347 МГц. На рис. 3.6б показан выход ПЧ для трёх диапазонов при частоте гетеродина выше ВЧ, а на рис. 3.7б – тот же выходной сигнал при частоте гетеродина ниже ВЧ. Самая нижняя, свободная от паразитных составляющих область на результирующей карте лежит между 195 и 290 МГц.

При использовании смесителя с частотой гетеродина выше частоты ВЧ сигнала получается больше свободных от паразитных составляющих областей, в которых можно выбрать ПЧ, по сравнению со смесителем, у которого частота гетеродина ниже частоты ВЧ. Первый вариант можно использовать для предотвращения возможных утечек ВЧ сигнала, но второй вариант более практичен и обходится дешевле. По этим причинам мы выбрали конфигурацию смесителя с частотой гетеродина ниже частоты ВЧ и выбрали ПЧ равную 210 МГц.

Оптимизация архитектуры

В этом примере мы рассмотрим, как использовать супергетеродинную архитектуру для реализации многодиапазонного приёмника LTE. Мы начали с анализа «идеальной» конфигурации, в которой все компоненты считаются идеальными. Фильтры имеют высокое подавление, а их частотные характеристики приближаются к идеальной форме. Коммутаторы почти не вносят потерь. Нелинейные компоненты, такие как усилители, смесители и генераторы, обладают широкими диапазонами линейности. Структурная схема такой «идеальной» конфигурации показана на рис. 3.8а.

На втором этапе мы заменили «идеальные» компоненты реальными устройствами. Мы использовали усилители и коммутаторы компании TriQuint. Смеситель был выбран от Hittite, а задающий генератор от Crystek. В связи с особенностями диапазонов LTE и промежуточной частотой, использованной в этой конструкции ($F_0 = 210$ МГц и $BW_{ПЧ} = 10$ МГц), мы разработали специальные фильтры в Keysight Genesys. Затем мы использовали модуль Circuit_Link, чтобы включить специальные фильтры в схемотехническую модель SystemVue. На рис. 3.9 показано, как импортировать синтезированный в Genesys фильтр в SystemVue с помощью модуля Circuit_Link. На рис. 3.8б показана структурная схема реального приёмника LTE.

Теперь давайте проанализируем на системном уровне «идеальную» и «реальную» схемы, уделяя особое внимание следующим параметрам:

- Покаскадный коэффициент шума ($\kappa K_{ш}$)
- Покаскадный коэффициент усиления (κK_y)
- Отношение несущей к шуму (Н/Ш)
- Мощность шума в канале (CNP)
- Мощность в канале (CP)
- Динамический диапазон каскада (SDR)

Чтобы смоделировать перечисленные выше параметры, мы использовали SystemVue для каждого диапазона и каждой конструкции. Результаты моделирования приведены в табл. 3.3.

		κK_y , дБ	$\kappa K_{ш}$, дБ	Н/Ш, дБ	SDR, дБ	CP, дБм	CNP, дБм
Идеальная схема	Диапазон 1	51,184	1,057	15,918	137,347	-45,702	-61,734
	Диапазон 2	50,606	1,104	15,871	137,7	-46,279	-62,265
	Диапазон 3	51,341	0,903	16,072	139,162	-45,549	-61,731
Реальная схема	Диапазон 1	48,53	1,606	15,22	139,615	-48,338	-63,69
	Диапазон 2	44,28	3,121	13,705	142,492	-52,535	-66,425
	Диапазон 3	49,547	1,236	15,589	140,706	-47,33	-63,042

Табл. 3.3. Результаты моделирования некоторых параметров системного уровня для обеих схем.

На рис. 3.10 - 3.13 показано изменение каждого параметра в разных каскадах обеих схем. Кроме того, на рис. 3.14 показаны кривые компрессии, которые помогают прогнозировать области линейной и нелинейной работы каждой схемы.

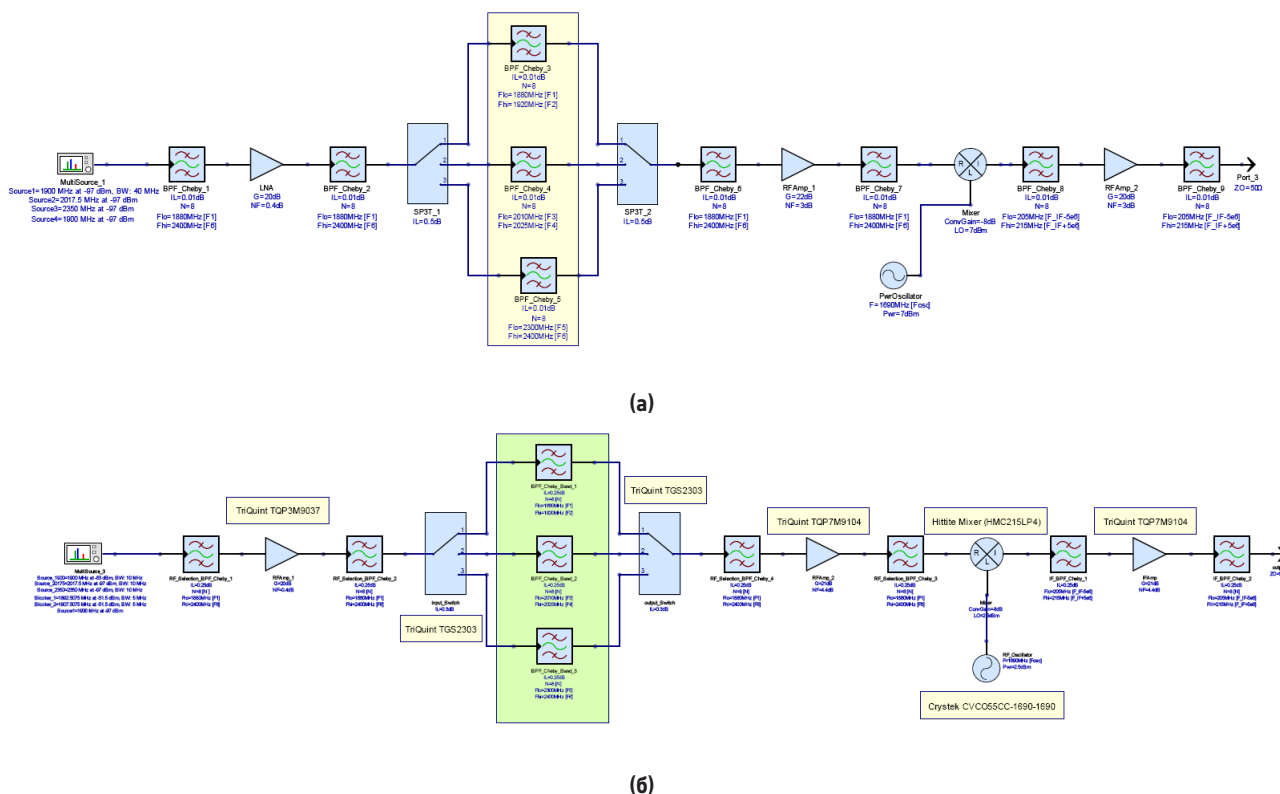


Рис. 3.8. Структурная схема трансивера LTE: идеальная (а) и реальная (б).

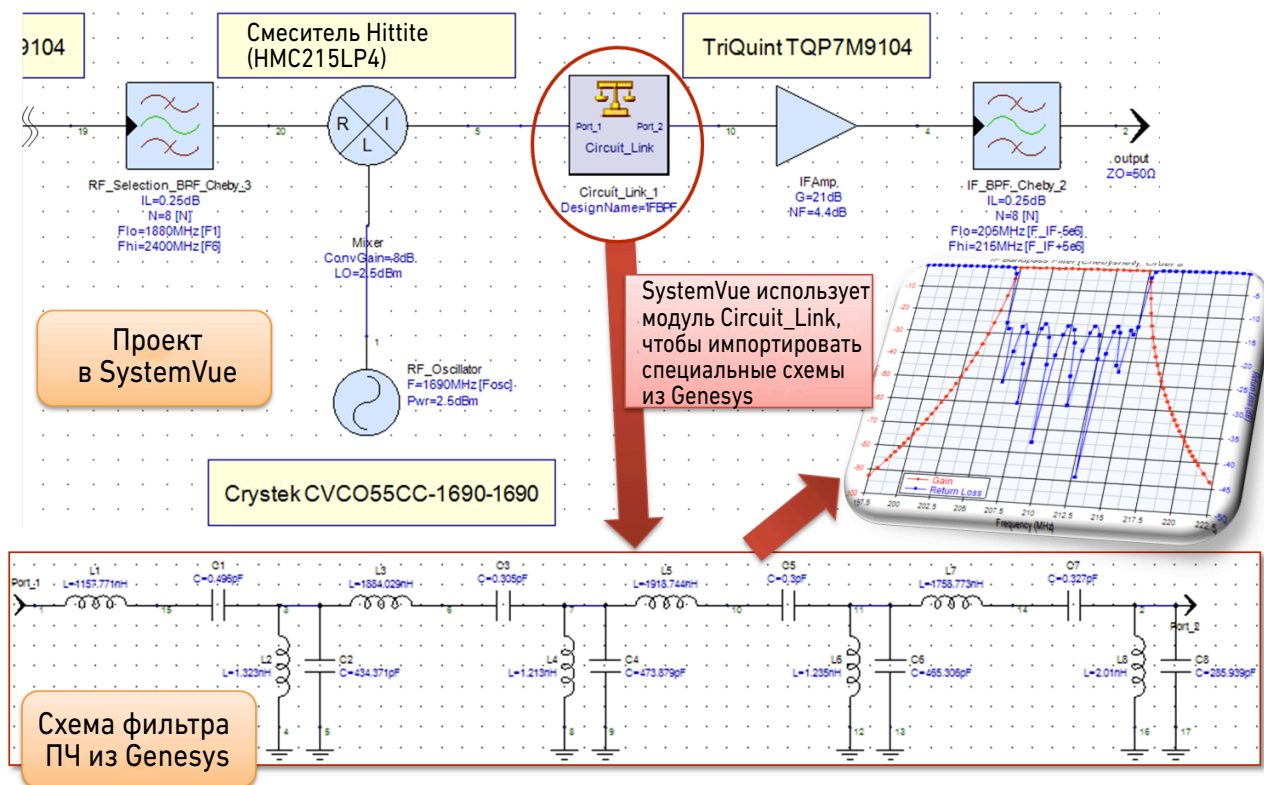


Рис. 3.9. SystemVue использует модуль Circuit_Link для импорта специальных схем (например, из Genesys).

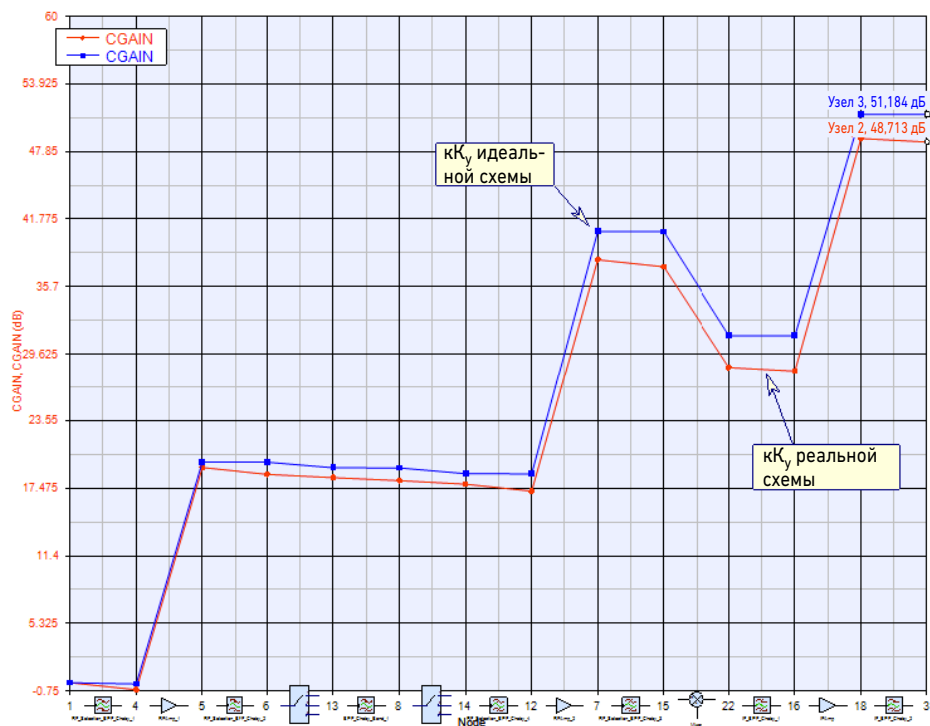


Рис. 3.10. Покаскадное усиление (κ_K).

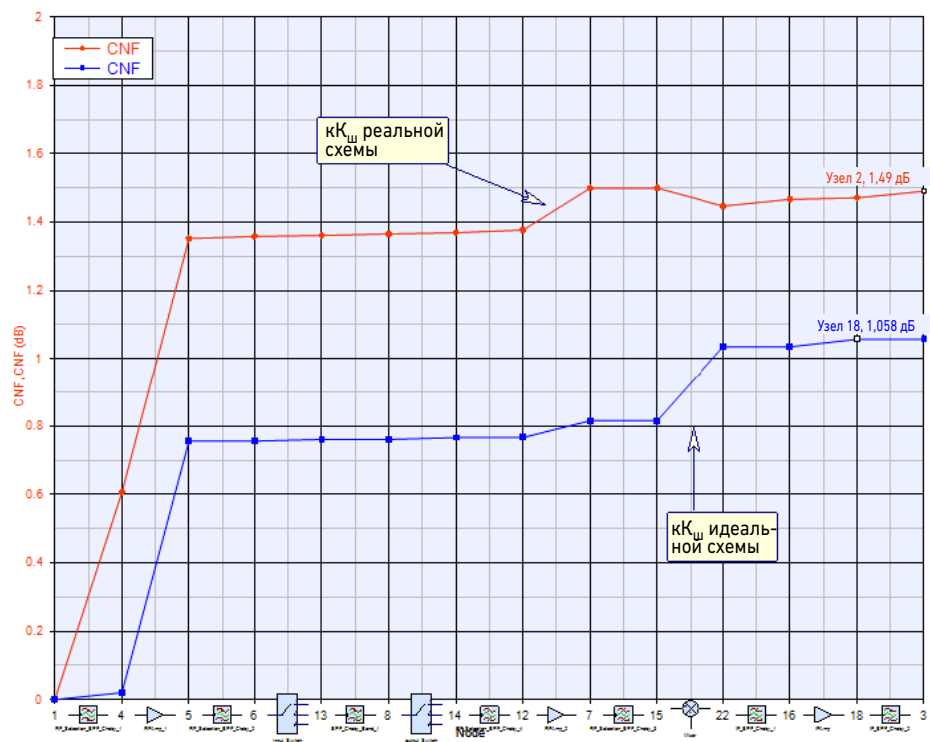


Рис. 3.11. Покаскадный коэффициент шума (κ_K) для диапазона 1.

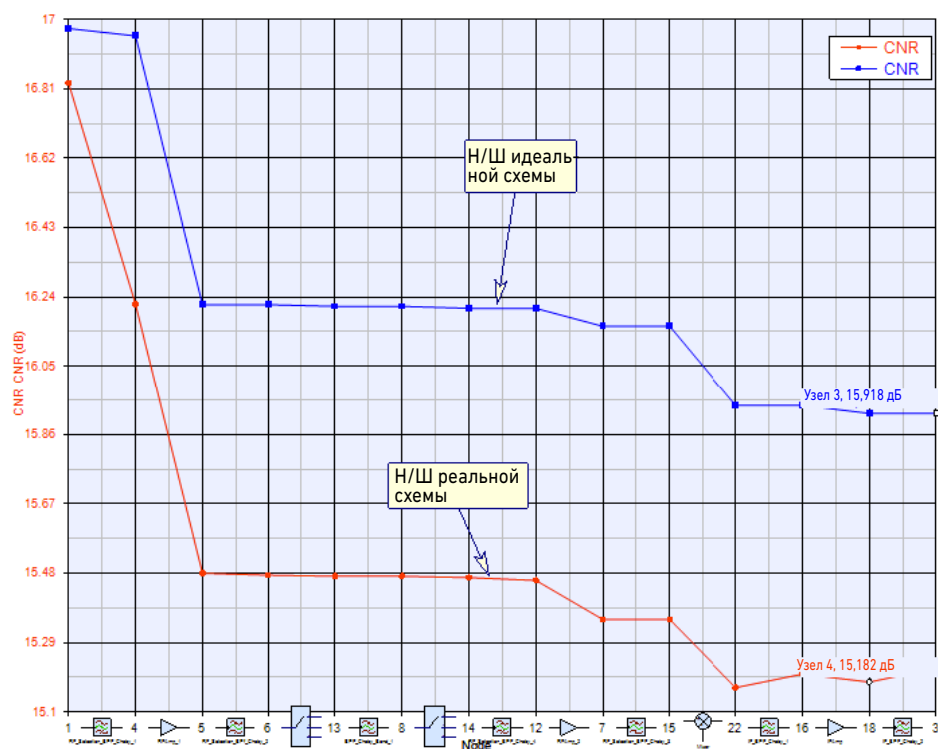


Рис. 3.12. Отношение несущей к шуму (Н/Ш) для диапазона 1.

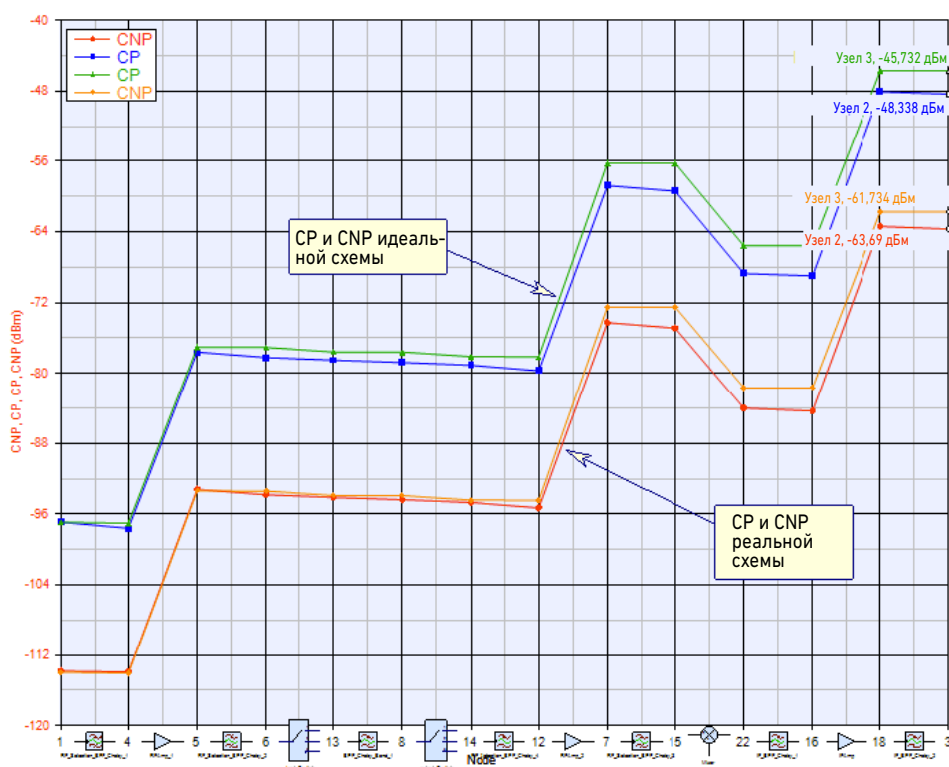


Рис. 3.13. Мощность в канале и мощность шума в канале для идеальной и реальной схемы.

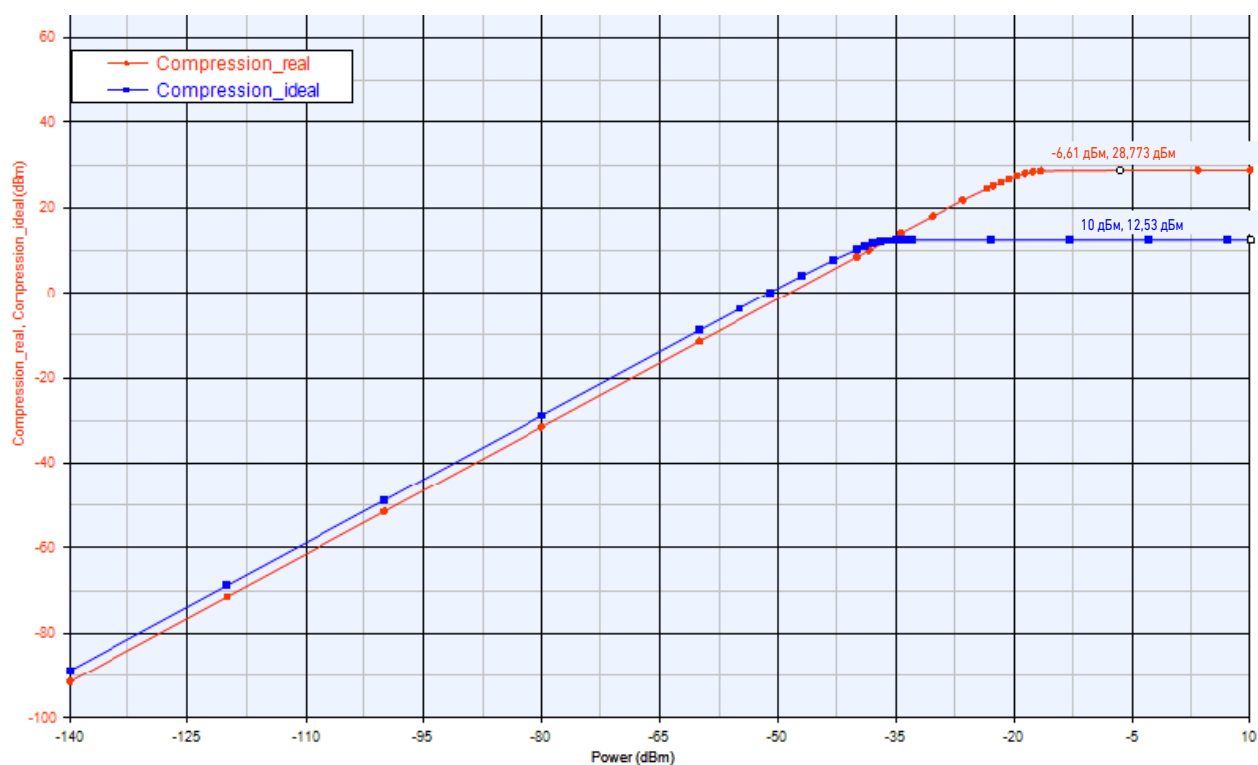


Рис. 3.14. Кривая компрессии для диапазона 1: идеальная схема (красный), реальная схема (синий).

Оценка характеристик

Раздел 7 технических характеристик абонентского оборудования LTE⁴¹ определяет, что приёмник LTE должен отвечать требованиям к чувствительности, избирательности по соседнему каналу (ACS) и блокировке (в узкой полосе, внутри полосы и за пределами полосы), а также требованиям к уровню паразитных и интермодуляционных составляющих. В следующих разделах мы покажем, как использовать SystemVue для тестирования и проверки схемы приёмника на соответствие стандарту LTE.

В табл. 3.4 представлены стандартные значения чувствительности в предположении, что в качестве схемы модуляции используется QPSK и полоса канала равна 10 МГц. Кроме того, в таблице указана максимальная входная мощность для относительной пропускной способности более 95 %.

Диапазон E-UTRA	Полоса канала, МГц	Стандартная чувствительность для QPSK, дБм	Максимальная входная мощность, дБм
34	10	-97	-25
39	10	-97	-25
40	10	-97	-25

Табл. 3.4. Стандартные значения чувствительности и максимальной входной мощности для диапазонов TDD 34, 39 и 40.

На рис. 3.15а показан входной тон частотой 1900 МГц с полной мощностью -97 дБм. На выходе тон с уровнем -48 дБм имеет частоту 210 МГц, а тон гетеродина подавляется до -138 дБм.

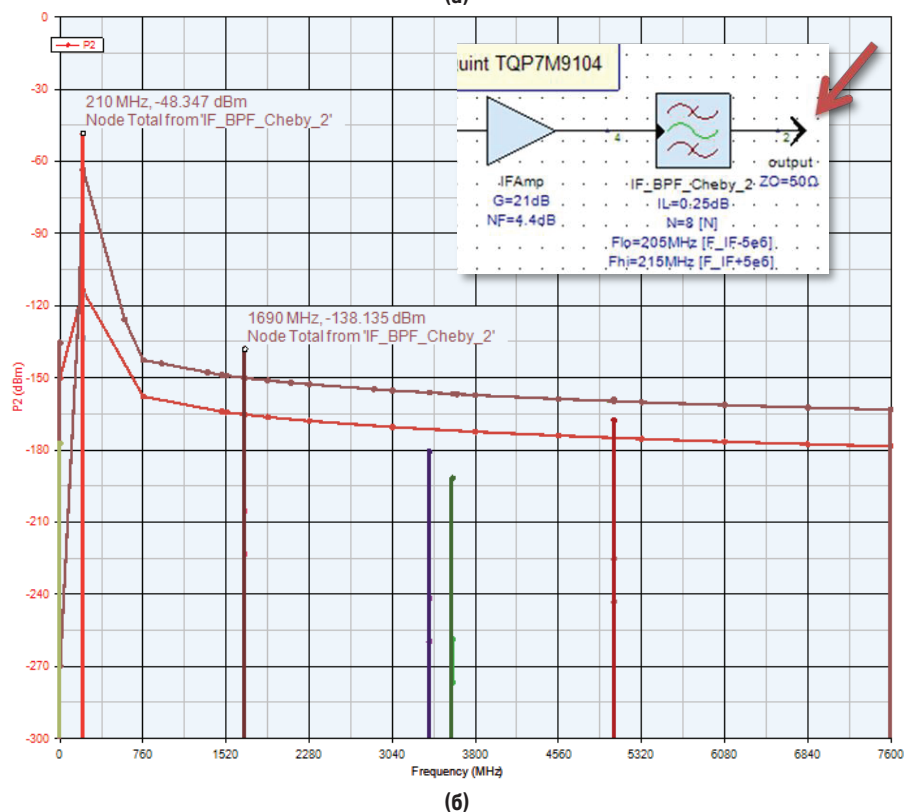
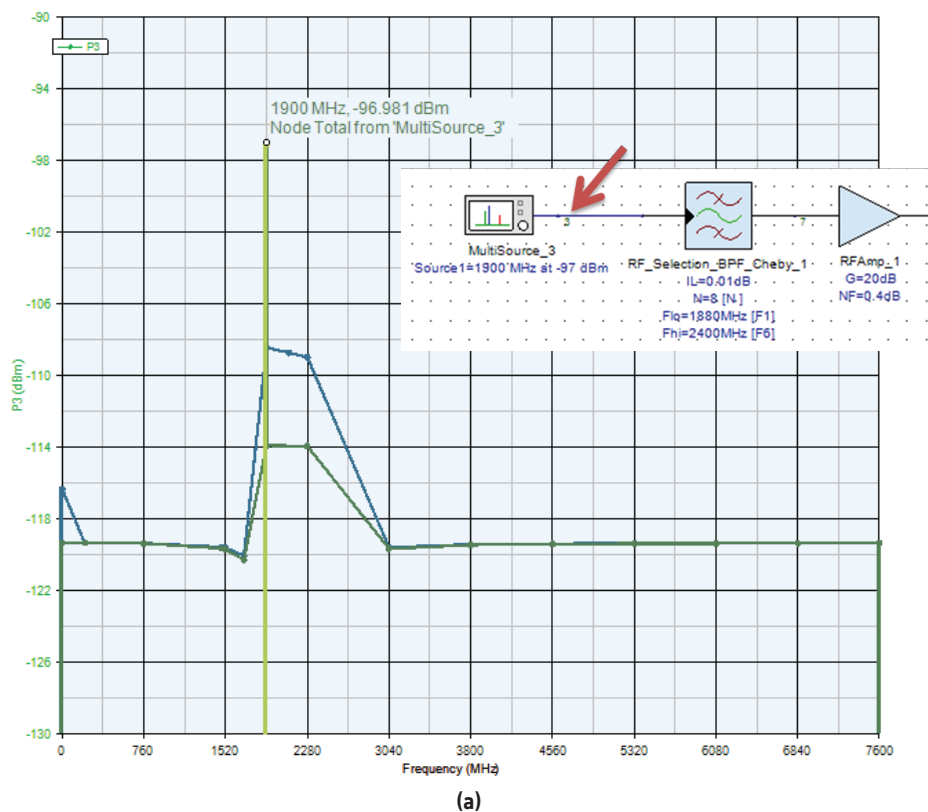


Рис. 3.15. Входной и выходной спектр тона частотой 1900 МГц: входной сигнал (а) и выходные сигналы (б), включая тон гетеродина частотой 1690 МГц.

Избирательность по соседнему каналу

Избирательность по соседнему каналу (ACS), называемая также подавлением соседнего канала (ACR), определяет способность приёмника выделять сигнал LTE на определённой частоте канала в присутствии сигнала в соседнем канале с заданной отстройкой частоты.

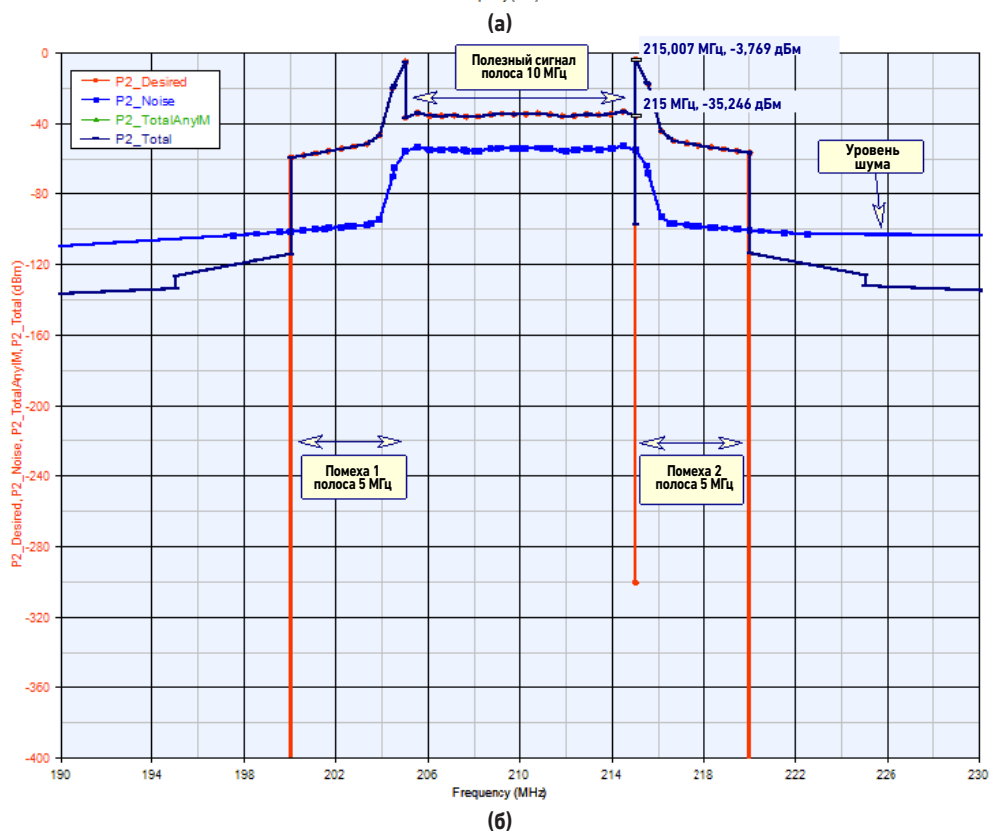
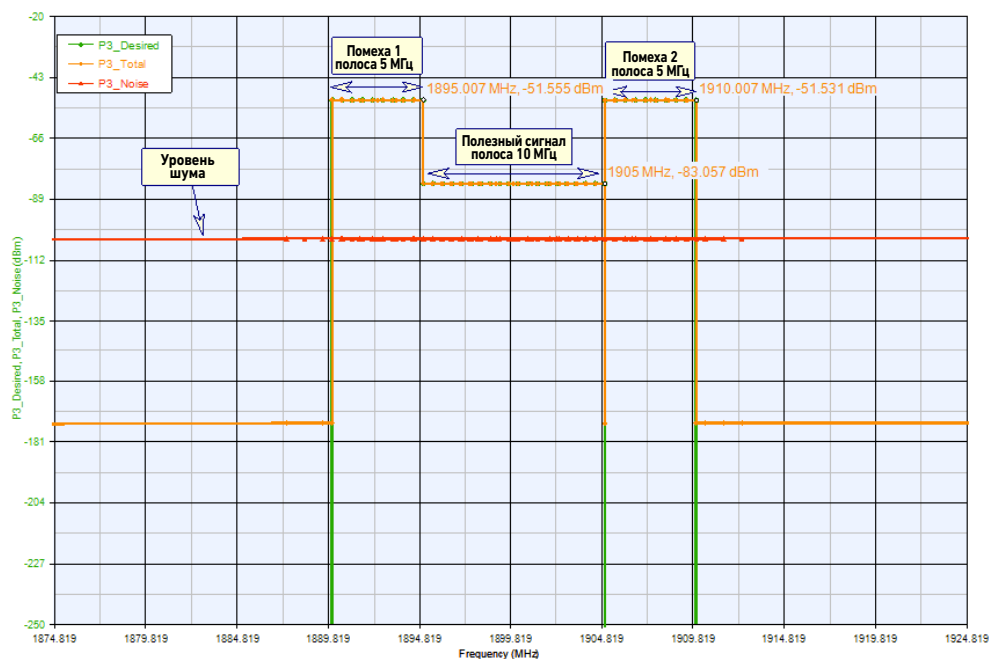


Рис. 3.16. Избирательность по соседнему каналу: полезный входной сигнал и сигналы помехи (а); выходной спектр (ACS ≈ 30 дБ) (б).

В документе 41 указано, что значение ACS в любом случае должно быть больше или равно 33 дБ, при условии, что максимальная допустимая входная мощность равна -25 дБм, а относительная пропускная способность больше или равна 95 %.

Параметр приёмника	Единицы измерения	Полоса канала		
		...	10 МГц	...
Мощность входного сигнала	дБм		Стандартная чувствительность + 14 дБ (т.е. = -97 + 14 = -83)	
Мощность помехи	дБм		Стандартная чувствительность + 45,5 дБ (т.е. = -97 + 45,5 = -51,5)	
Полоса помехи	МГц		5	
Отстройка частоты помехи	МГц		$\pm(7,5 + 0,0075)$	

Табл. 3.5. Сводка контрольных параметров ACS (по табл. 7.5.1-2 документа [39]).

Опираясь на контрольные параметры, представленные в табл. 3.5, мы использовали симулятор Spectrasys для оценки ACS реальной схемы. На рис. 3.16а показаны входные сигналы, которые мы использовали для проверки ВЧ тракта (два сигнала помехи шириной 5 МГц с уровнем мощности примерно -51,5 дБм), а также полезный сигнал (канал шириной 10 МГц с уровнем мощности примерно -83 дБм). Выходные сигналы показаны на рис. 3.16б. В схеме обе помехи и полезный сигнал усиливались, и их частота понижалась. Помехи располагались снизу и сверху по частоте от полезного диапазона – примерно от 200,0075 до 205,0075 МГц и от 215,0075 до 220,0075 МГц соответственно. Уровень мощности обеих помех равен примерно -3,7 дБм. Уровень полезного сигнала равен примерно -35,25 дБм. ACS равен примерно 31,45 дБ, что не отвечает требованиям стандарта (т.е. $ACS \leq 33$ дБ). По этой причине мы добавили два достаточно узкополосных полосовых фильтра, чтобы дополнительно выиграть 3-5 дБ и улучшить общую избирательность системы (рис. 3.18).

Характеристики блокировки

Характеристики блокировки определяют способность приёмника выделять полезный сигнал на указанной частоте канала в присутствии нежелательного (блокирующего) сигнала, не связанного с паразитными составляющими и сигналами соседних каналов. Блокирующие сигналы можно разделить на несколько категорий: внеполосные, узкополосные и внеполосные.

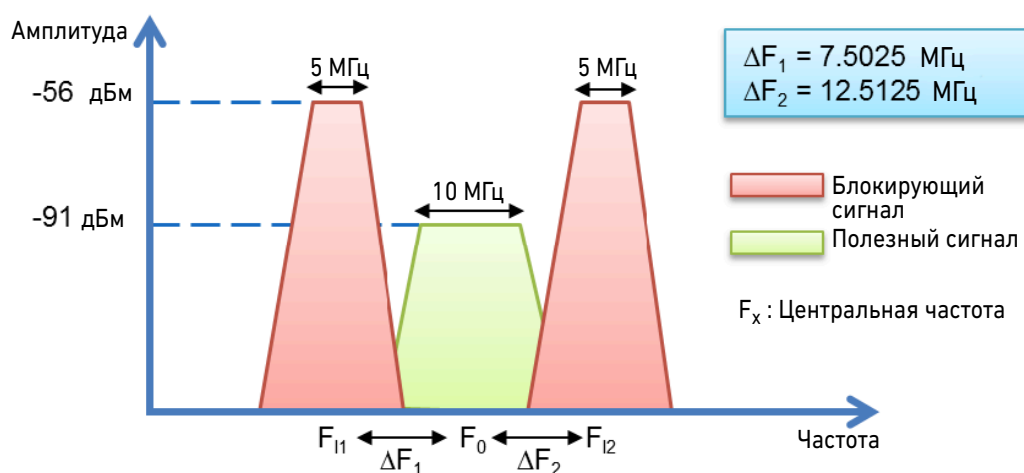


Рис. 3.17. Характеристики внеполосной блокировки.

В этом разделе мы сфокусируемся на том, как использовать SystemVue для измерения влияния внеполосных блокирующих сигналов на наш приёмник LTE. На рис. 3.17 показан тест внеполосной блокировки, детально описанный в таблицах 7.6.1.1-1 и 7.6.1.1-2 спецификаций абонентского оборудования⁴¹.

Чтобы оценить реакцию приёмника LTE на внеполосные блокирующие сигналы, измерялась относительная пропускная способность (называемая также коэффициент пропускной способности). Чтобы приёмник прошёл тест, относительная пропускная способность должна быть больше или равной 0,95 (95 %).

Для оценки параметров приёмника использовалась схема измерения полного потока данных LTE. На рис. 3.18 показаны компоненты, использующие библиотеку LTE для модуляции, передачи, приёма

и демодуляции сигнала LTE TDD. Стоит упомянуть, что стенд для измерения потока данных LTE использует, в основном, симулятор Data Flow, который автоматически взаимодействует с симулятором Spectrasys для оценки ВЧ системы.

Чтобы оценить характеристики приёмника, нужно рассмотреть следующие основные сценарии:

- Анализ влияния внутриполосных блокирующих сигналов
- Анализ влияния стандартного теплового шума

В ходе исследования мы постепенно продвигались вперёд, включая (или выключая) в каждой итерации компоненты блокирующих сигналов и плотности шума. Результаты нашего моделирования приведены в таблице 3.6. Мы использовали линейную оценку канала в модуле приёмника нисходящего канала LTE. Согласно строкам 1-4 таблицы в отсутствии шума блокирующие сигналы не оказывают заметного влияния на характеристики приёмника. Однако при включении компонента плотности шума коэффициент пропускной способности становится меньше 95 %. Это влияние подтверждается строками 9 и 10, где ВЧ схема не учитывается, а влияние шума оценивается по отношению к стандартной схеме LTE.

№	Плотность шума	Блокирующий сигнал 1	Блокирующий сигнал 2	Пропускная способность, бит/с	Коэффициент, %
1	Нет	Нет	Нет	1603200	100
2	Нет	Нет	Да	1603200	100
3	Нет	Да	Нет	1603200	100
4	Нет	Да	Да	1603200	100
5	Да	Нет	Нет	1406328	87,72
6	Да	Нет	Да	1480224	92,329
7	Да	Да	Нет	1418736	88,494
8	Да	Да	Да	1437072	89,638
9	Нет	-	-	1603200	100
10	Да	-	-	1423128	88,768

Табл. 3.6. Результаты моделирования в случае линейной оценки канала.

Чтобы улучшить характеристики системы, мы заменили линейную оценку канала в модуле нисходящего приёмника LTE оценкой канала типа «Минимальная среднеквадратическая ошибка» (MMSE), которая, как известно, более устойчива к шуму. Результаты моделирования приведены в табл. 3.7. Как и ожидалось, характеристики приёмника улучшились. Коэффициент пропускной способности стал равным 100 %, несмотря на присутствие теплового шума и блокирующих сигналов.

Заметьте, что мы провели не все тесты блокировки.

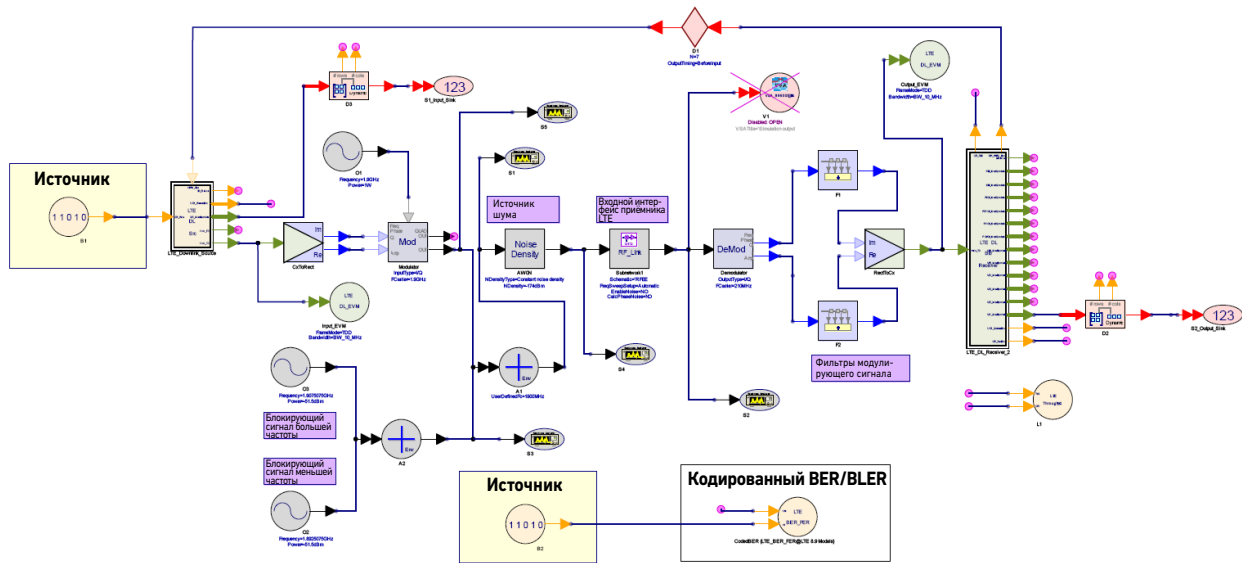


Рис. 3.18. Схема симулятора Data Flow LTE для тестирования входного интерфейса приёмника LTE.

№	Плотность шума	Блокирующий сигнал 1	Блокирующий сигнал 2	Пропускная способность, бит/с	Коэффициент, %
1	Нет	Нет	Нет	1603200	100
2	Нет	Нет	Да	1603200	100
3	Нет	Да	Нет	1603200	100
4	Нет	Да	Да	1603200	100
5	Да	Нет	Нет	1603200	100
6	Да	Нет	Да	1603200	100
7	Да	Да	Нет	1603200	100
8	Да	Да	Да	1603200	100
9	Нет	-	-	1603200	100
10	Да	-	-	1603200	100

Табл. 3.7. Результаты моделирования в случае оценки канала MMSE.

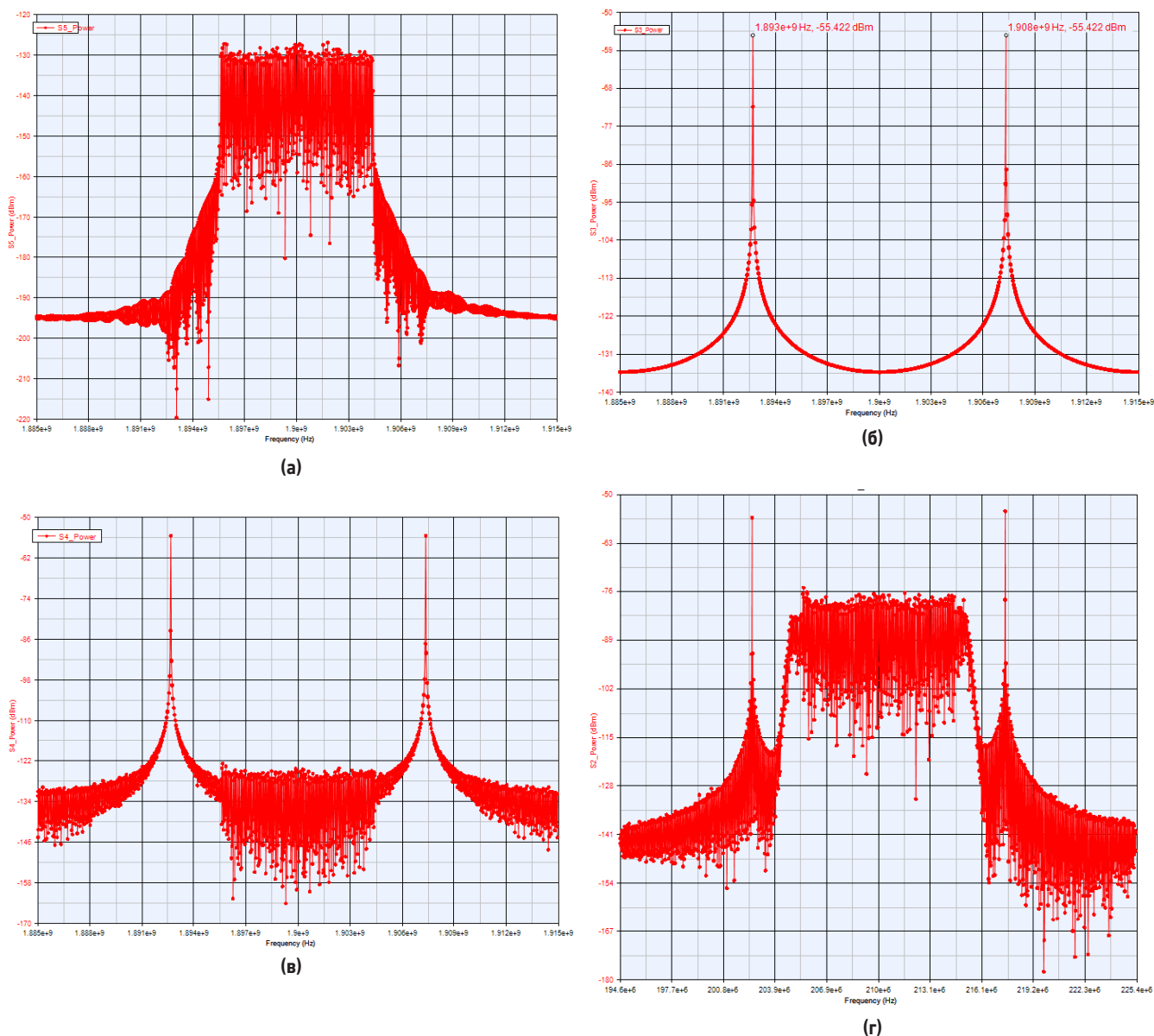


Рис. 3.19. Спектр диапазона 1: сигнал передатчика (а), блокирующие сигналы (б), сигнал с шумом (в), сигнал ПЧ (г).

На рис. 3.19 показан сигнал LTE диапазона 1. Исходный сигнал показан на рис. 3.19а. На рис. 3.19б показаны блокирующие сигналы. На рис. 3.19в показан принимаемый сигнал с шумом и блокирующими сигналами. И на рис. 3.19г показан преобразованный по частоте сигнал, измеренный на выходе ВЧ интерфейса.

3.4 Заключение

В этой главе мы представили концепцию методологии проектирования и моделирования системного уровня. Мы выбрали реальный случай и показали, как проектировать и оценивать некоторые аспекты ВЧ схемы на системном уровне с помощью функций и возможностей САПР SystemVue. Существует множество других аспектов и проблем проектирования (например, влияние фазового шума и нелинейностей), которые тоже можно исследовать. Все эти и другие задачи охвачены учебными видеороликами, материалами и примерами, прилагаемыми к САПР SystemVue.

Дополнительную информацию о САПР Keysight SystemVue можно найти на страницах:

- <http://www.keysight.com/find/eesof-systemvue>
- <http://www.keysight.com/find/eesof-systemvue-videos>

Глава 4. Введение в цифровую связь

4.1 Введение

В этой главе мы познакомим вас с основами цифровой связи и представим теорию и структуру цифровой радиосистемы. Большинству современных вычислительных устройств необходимы какие-либо беспроводные интерфейсы, а с ростом доли мобильных устройств эта тенденция только нарастает. В связи с этим возникает потребность в лучших, более эффективных и более надёжных методах беспроводной передачи данных. Одним из самых распространённых способов передачи данных на большие расстояния является цифровая радиосвязь.

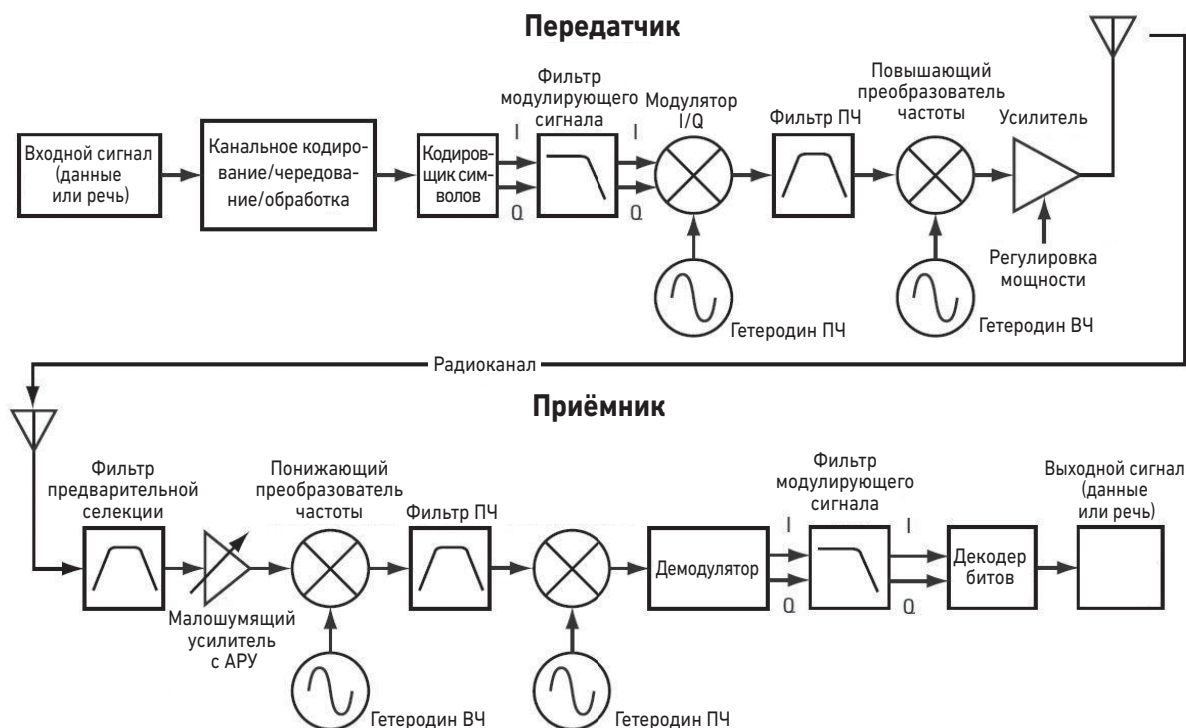


Рис. 4.1. Структурная схема базовой цифровой системы радиосвязи.

На рис. 4.1 показана общая структурная схема цифровой радиокommunikационной системы, которая обычно состоит из передатчика, радиоканала и приёмника. На пути от входа передатчика до выхода приёмника цифровые сигналы испытывают множество трансформаций. Структурная схема демонстрирует процесс преобразования сигнала от момента его подачи до момента декодирования. При внимательном рассмотрении на рис. 4.1 можно увидеть симметрию цифровой радиостанции. До некоторой степени приёмник можно считать обратной реализацией передатчика. Следовательно, проблемы измерений аналогичны для обеих частей цифровой радиосистемы. Тем не менее, в разных узлах системы существуют специфические проблемы, обусловленные конструктивными особенностями приёмников и передатчиков. Например, приёмник имеет схемы коррекции и синхронизации, которые помогают восстановить и декодировать зашумленные и задержанные по времени сигналы⁴⁸.

4.2 Передатчик

Цифровой радиопередатчик получает низкочастотный модулирующий сигнал и преобразует его в сигнал, эффективно передаваемый по радиоканалу. Когда данные попадают в передатчик, он кодирует их так, чтобы упростить процесс передачи, что помогает корректировать ошибки и делает данные более защищёнными. Перед преобразованием сигнала в высокую частоту (ВЧ), передаваемую через радиоканал, сигнал оцифровывается, что позволяет воспользоваться преимуществами цифровой модуляции. Оцифрованный сигнал кодируется для повышения эффективности использования имеющейся полосы и минимизации влияния шума и помех, присутствующих в канале. Кодированный сигнал фильтруется, модулируется и снова

преобразуется в аналоговый сигнал, частота которого преобразуется в нужную частоту передачи. И, наконец, перед подачей в антенну ВЧ сигнал фильтруется и усиливается.

4.3 Приёмник

Цифровой радиоприёмник можно реализовать разными способами, но некоторые компоненты в нём присутствуют всегда. Приёмник должен выделять полезный ВЧ сигнал в присутствии возможных помех. Поэтому первым компонентом приёмника является фильтр предварительной селекции, который подавляет принимаемые антенной внеполосные сигналы. Малошумящий усилитель (МШУ) усиливает сигнал до нужного уровня, добавляя к нему некоторый минимальный шум. Смеситель понижает частоту ВЧ сигнала до промежуточной частоты (ПЧ), смешивая ВЧ сигнал с сигналом гетеродина (LO). Фильтр ПЧ подавляет возникающие в смесителе нежелательные частотные составляющие и сигналы соседних частотных каналов. После фильтра ПЧ в конструкции приёмника возможны вариации.

4.4 Канал

Между передатчиком и приёмником расположен канал, который представляет собой некоторую среду передачи, например медный кабель или открытое пространство. Поскольку наш учебник посвящен цифровой радиосвязи, мы не будем углубляться в структуру канала. Тем не менее, канал всё-таки диктует параметры, на которых строится система связи. Например, при передаче по медному кабелю целостность сигнала обеспечивается значительно лучше, чем при передаче по эфиру, поэтому данные могут передаваться быстрее и использовать менее устойчивые к помехам схемы модуляции. В эфире передаваемые сигналы подвержены затуханию и отражению, поэтому должны модулироваться с применением менее быстрых, но более устойчивых к помехам схем модуляции. Канал является важной частью, диктующей требования ко всей системе. Он должен обеспечивать хорошую целостность сигнала и низкий коэффициент битовых ошибок (BER).

4.5 Другие аспекты

Кроме схем модуляции существуют и другие конструктивные факторы, которые надо учитывать. Как уже говорилось, тип кодирования входных данных может помочь в предотвращении потерь данных, в повышении надёжности канала и даже может обеспечить чередование нескольких потоков данных в одной передаче. Но не только тип кодирования, но и структура фрейма или способ структурирования самих данных может привести к снижению BER за счёт включения данных синхронизации. За счет выбора фильтров и их параметров можно снизить шум, сузить занимаемую полосу частот и/или увеличить расстояние передачи. Некоторые фильтры могут иметь меньшую стоимость и больший КПД, но надёжно работать только на малых расстояниях. Поэтому, в зависимости от спецификаций, существует множество вариантов реализации системы передатчик-приёмник.

Глава 5. Проектирование передатчика

5.1 Основы конструкции передатчика

Передатчик можно считать половиной цифровой системы связи, которая отвечает за кодирование, преобразование и передачу сигнала по каналу. Передатчик можно реализовать разными способами, однако в соответствии с назначением настоящего документа мы опишем схему передатчика лишь кратко. Кроме того, мы покажем, как создать передатчик в САПР SystemVue, а в конце каждого раздела будем знакомить вас с основами теории.

Общий пример структурной схемы передатчика приведён на рис. 5.1. В первом функциональном блоке передатчика непрерывный аналоговый сигнал преобразуется в дискретный цифровой поток. Этот процесс называется оцифровкой.

Затем выполняется кодирование голоса для обеспечения сжатия данных. После этого применяется канальное кодирование с целью минимизации влияния шумов и помех в канале связи. Канальное кодирование добавляет к входным данным дополнительные биты и удаляет избыточные биты. Эти дополнительные биты используются для коррекции ошибок или для передачи обучающих последовательностей, обеспечивающих идентификацию или коррекцию частотной характеристики. Это может упростить синхронизацию данных (или выделение тактовой частоты символов) в приёмнике.

Тактовая частота символов обеспечивает точную синхронизацию передачи отдельных символов. В момент прохождения фронтов тактовой частоты передаваемая несущая имеет корректные значения I/Q (амплитуды и фазы) для конкретного символа (конкретной точки сигнального созвездия). Затем значения I/Q (амплитуды и фазы) передаваемой несущей изменяются так, чтобы представить другой символ. Интервал между этими двумя моментами называется периодом тактовой частоты символов. Обратной величиной периода является тактовая частота символов. Фаза тактовой частоты корректна в том случае, если она совпадает с оптимальным значением и позволяет детектировать символы.

Следующим шагом преобразования сигнала в передатчике является фильтрация. Фильтрация очень важна для повышения эффективности использования занимаемой полосы частот. Без фильтрации сигналы имели бы слишком крутые фронты и, следовательно, имели бы очень широкий частотный спектр. Для упрощения на рис. 5.1 показан один фильтр, но на практике используются два фильтра, по одному в каналах I и Q . Эти фильтры формируют компактный и спектрально эффективный сигнал, которым можно модулировать несущую.

Затем выходной сигнал канального кодера подаётся на модулятор. Поскольку в радиосигнале присутствуют независимые составляющие I и Q , половина информации передаётся составляющей I , а другая половина – составляющей Q . Это одна из причин, по которой цифровые радиостанции хорошо работают с таким типом цифрового сигнала. Компоненты I и Q отделены друг от друга.

Остальная часть передатчика мало отличается от типового ВЧ или СВЧ передатчика. Сигнал преобразуется в более высокую промежуточную частоту (ПЧ), а затем – в радиочастоту. Все нежелательные сигналы, возникающие в процессе преобразования, отфильтровываются⁴⁹. Структурная схема передатчика представлена на рис. 5.1.

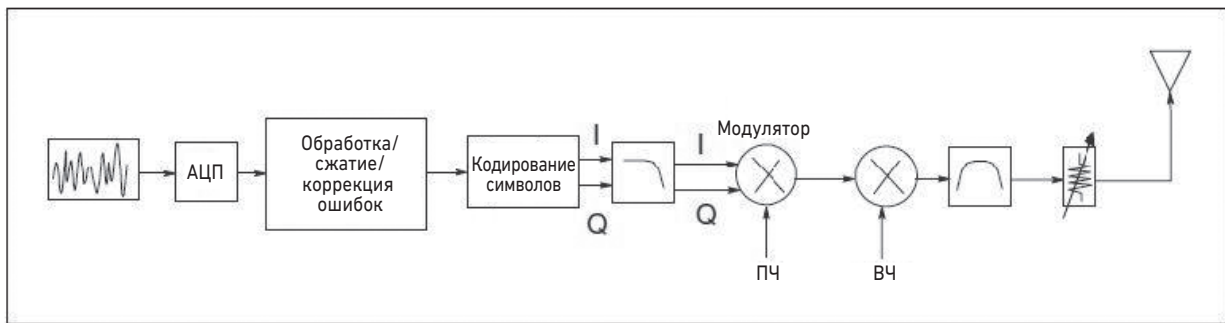


Рис. 5.1. Общая структурная схема передатчика

Пример LinearModulation

Цель этой главы заключается в демонстрации создания собственного передатчика в САПР SystemVue. Используемый нами пример основан на примере линейной модуляции LinearModulation. Этот базовый пример охватывает большую часть аспектов проектирования передатчика.

Чтобы найти этот и другие примеры в SystemVue, нужно выбрать пункт Open Example (открыть примеры) в меню Help (справка) или использовать сочетание клавиш Ctrl+E. Чтобы открыть пример LinearModulation:

1. откройте меню Help -> Open Example/Ctrl+E (как показано ниже на рис. 5.2);

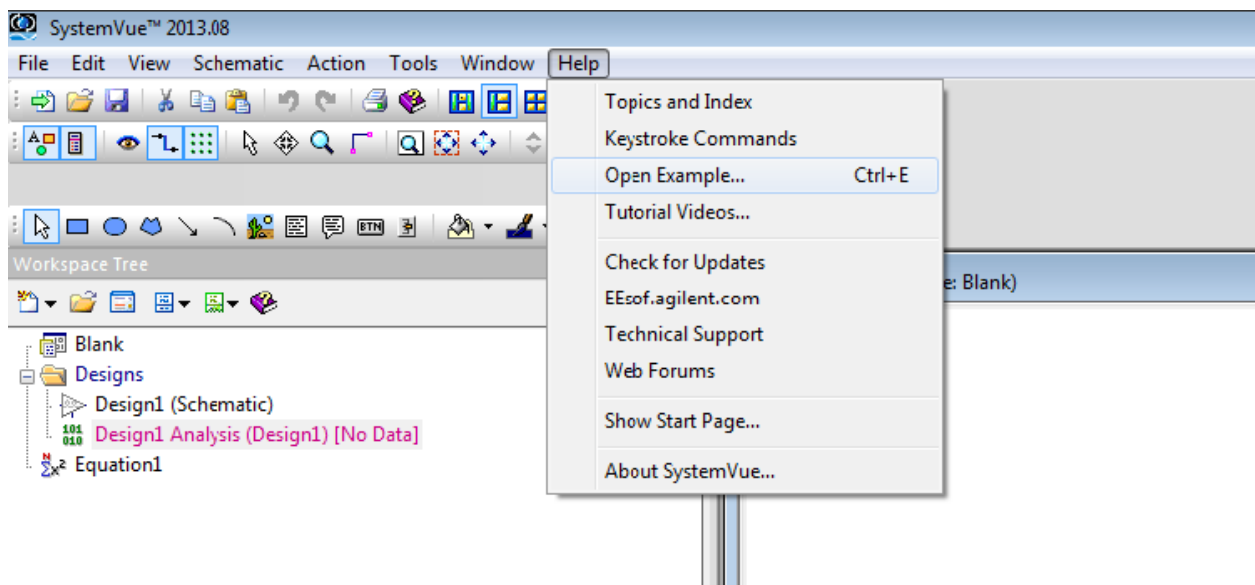


Рис. 5.2. Пункт меню "Open Example".

2. перейдите на вкладку Comms (связь);
3. откройте папку DigitalModulation (цифровая модуляция). Здесь вы и найдёте нужный пример.

В результате вы найдёте пример рабочей среды со схемой, показанной на рис. 5.3, которую мы возьмём за основу для проектирования нашего приёмника.

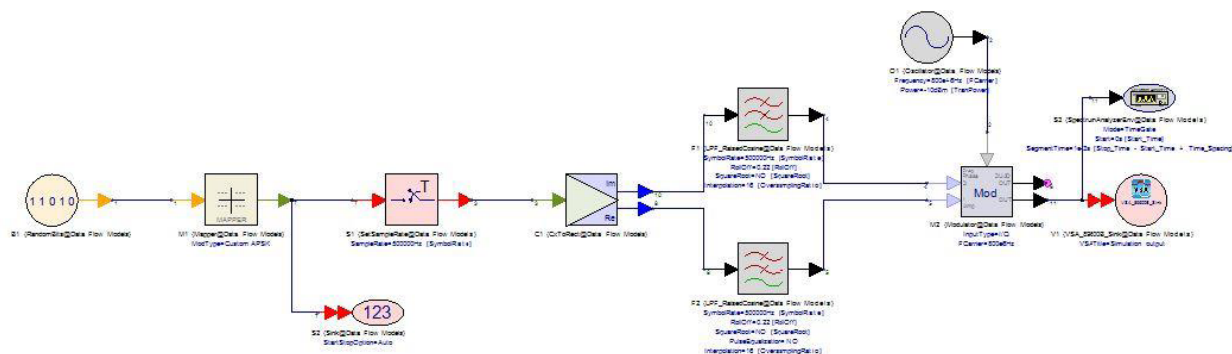


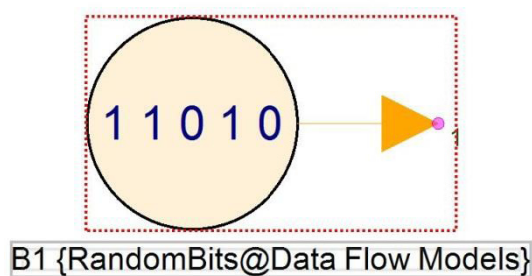
Рис. 5.3. Схема из примера LinearModulation.

Важно отметить, что наша финальная схема будет близка к схеме, показанной на рис. 5.3, хотя и не будет точной её копией.

5.2. Входные данные

Для начала:

1. Создайте новую рабочую среду в SystemVue и сохраните её под именем "Cook_Transmit".
2. Назовите схему "Transmitter" (передатчик), анализ потока данных "Transmitter_DF", папку "Source" (источник) и уравнение "TransVari".
3. Перейдите в анализ потока данных, выберите проектирование передатчика и набор данных "Transmitter_Data".
4. В селекторе компонентов выберите "Bits" (биты) или нажмите "B" и затем щёлкните, чтобы установить блок (рис. 5.4).
5. Дважды щёлкните на Bits и включите "Advanced parameters" (расширенные параметры). Измените "SampleRateOption" (частота дискретизации) в "0:Untimed" (0:Несинхронизировано).



(a)

Name	Value	Units
ProbOfZero	0.5	()
BitRate	Sample_Rate	Hz
ShowAdvancedParams	1:YES	()
SampleRateOption	0:UnTimed	()
SampleRate	Sample_Rate	Hz
InitialDelay	0	()
BurstMode	0:OFF	()

(б)

Рис. 5.4. Модель Bits (a), часть окна свойств Bits (б).

Это будут наши данные, которые мы хотим передать проектируемым передатчиком.

Разбиение на фреймы. В отличие от того, что показано в приведённом выше примере, где отправляются случайные биты без какого-либо значения, в реальном передатчике обычно передаётся реальная информация, разделённая на блоки. Как правило, для этого выполняется структурирование данных, которые вы хотите передать, и добавление к данным дополнительных битов. Такая структура данных называется фреймовой структурой. Фреймовая структура работает, как протокол, который информирует приёмник о том, как принимать и расшифровывать принимаемые данные, используя дополнительную информацию для синхронизации и коррекции ошибок. Базовая фреймовая структура показана на рис. 5.5.



Рис. 5.5. Схема базовой фреймовой структуры.

На этой схеме интервал ожидания и защитный интервал представляются строками нулевых битов, которые обозначают начало и конец пакета (хотя, в зависимости от передатчика, порядок отправки интервала ожидания и защитного интервала может изменяться). Преамбула имеет достаточно большую длину и используется для сортировки, оценки канала, коррекции АЧХ и для передачи прочей информации, помогающей приёмнику сортировать пакеты. Полезные данные – это как раз сами передаваемые данные.

Для реализации этой схемы в SystemVue нужно либо воспользоваться редактором уравнений, либо вставить блок "MathLang" и написать код самостоятельно. Вместо этого мы используем универсальный функциональный блок из библиотеки DigMod, который называется "DigMod_SourceL_RF". Этот блок специально предназначен для того, чтобы выступать в роли идеального источника. В нём можно устанавливать различные параметры, такие как длина преамбулы, длина полезных данных и тип модуляции преамбулы (Mod).

5.3 Кодирование

Метод, который мы используем для канального кодирования, называется свёрточным кодом (рис. 5.6). Для его реализации:

1. найдите и вставьте блок "ConvolutionalCoder";
2. соедините выход Bits с входом ConvolutionalCoder.

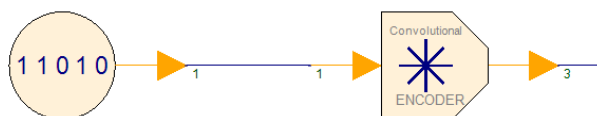


Рис. 5.6. Блок Bits, включённый последовательно с блоком ConvolutionalCoder.

Канальное кодирование и свёрточный код. При передаче через большинство сред на сигнал воздействует шум. Канальное кодирование позволяет сжать сигнал и структурировать данные таким образом, чтобы сделать их устойчивыми к воздействию помех. Кодирование может выполняться разными способами, но в данном примере мы используем только свёрточное кодирование.

Свёрточный код позволяет исправлять ошибки. Сначала данные структурируются с помощью полиномиальных уравнений. Обычно члены уравнения приравниваются к регистрам на основе их мощности, и каждое уравнение настраивается так, чтобы иметь отдельный выход. Пример таких уравнений показан на рис. 5.7. Здесь показан кодер в действии.

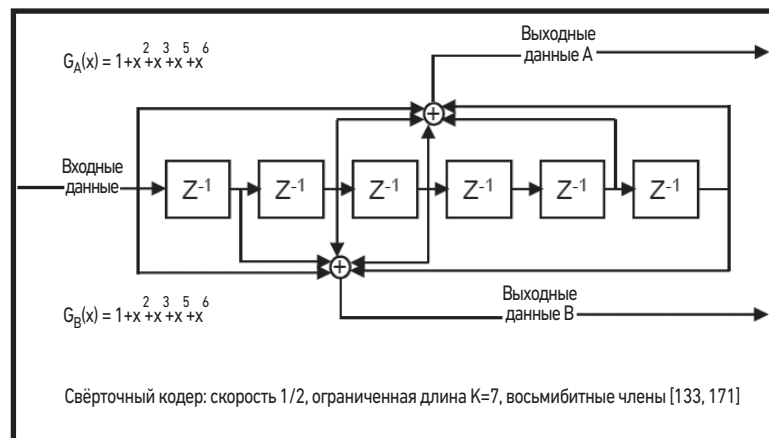


Рис. 5.7. Схема свёрточного кодирования с уравнениями, представляющими разные выходы.

Свёрточный кодер вводит данные в последовательность регистров. С каждым периодом тактовой частоты данные перемещаются из регистра в регистр, которые, в свою очередь, подают их на сумматоры. Каждый сумматор представляет определенное уравнение и настраивается в соответствии с параметрами этого уравнения. Если взглянуть на степени членов уравнения, то можно увидеть, что они соответствуют показанному на схеме выходу регистра. Поэтому, например, если рассмотреть уравнение G_A , можно увидеть, что здесь присутствует член $5^{\text{й}}$ степени, который представляется сумматором, подключенным к выходу данных A. К выходу данных B это не применимо, поскольку в G_B член $5^{\text{й}}$ степени отсутствует. Это изменяет потоки битов независимо от их длины, эффективно удваивая каждый входной бит. Вы можете изменять отношение входа к выходу и степень членов полинома, редактируя этот блок. Расширенное свёрточное кодирование можно выполнить в MathLang или путём добавления сумматоров и регистров.

5.4 Формирование символов

Теперь мы хотим преобразовать наши кодированные данные в символьную форму.

1. Найдите и поместите в схему "Mapper: Complex Signal Mapper" (Формирователь: Формирователь комплексных сигналов).
2. Подключите формирователь к кодеру.
3. Отредактируйте блок формирователя и измените тип модуляции на 16 QAM.

По окончании этой процедуры структурная схема должна выглядеть так, как показано на рис. 5.8.

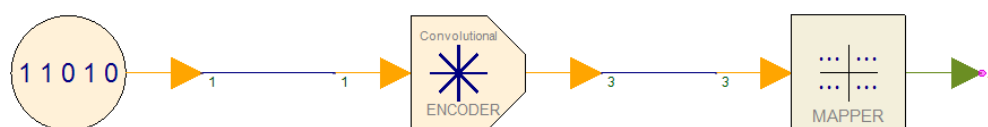


Рис. 5.8. Последовательно включенные блоки Bits, Convolutional Coder и Mapper.

Формирование символов. Формированием символов называется операция группирования битов в символы. Число битов, используемых для формирования группы, обычно произвольно, но от него зависит скорость передачи и целостность сигнала. Как правило, чем больше битов входит в группу, тем больше информации эта группа может передать и, как правило, в этом случае снижается вероятность ошибок. Символы можно представить графически на диаграмме, которая называется сигнальное созвездие. Сигнальное созвездие является неотъемлемым атрибутом IQ-модуляции.

Блок Mapper в SystemVue можно настроить так, чтобы он поддерживал разные типы модуляции от QAM до специальных видов APSK.

5.5 Фильтрация формы импульсов

Чтобы настроить фильтрацию в канале, выполните следующие действия.

1. Установите блок "SetSampleRate" (установить частоту дискретизации) и соедините его с формирователем символов.
2. Установите блок "CxToRect" и соедините его с блоком SetSampleRate (рис. 5.9).

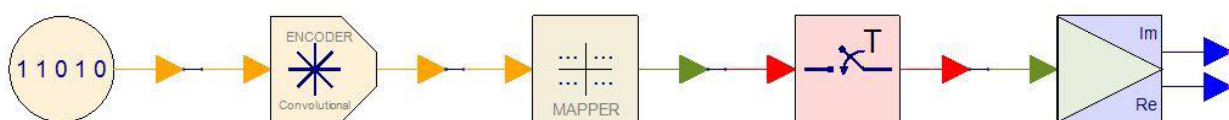


Рис. 5.9. Схема передатчика на текущем этапе проектирования.

Обычно в свойствах блока SetSampleRate достаточно установить постоянное значение SampleRate (частоты дискретизации), но мы будем использовать символьную скорость и в других местах этой модели.

Поэтому мы создадим в окне уравнения переменную символьной скорости.

3. Дважды щёлкните на TransVari в дереве рабочей среды и в текстовом редакторе установите значение переменной «SymbolRate», равное 500e3 (рис. 5.10).

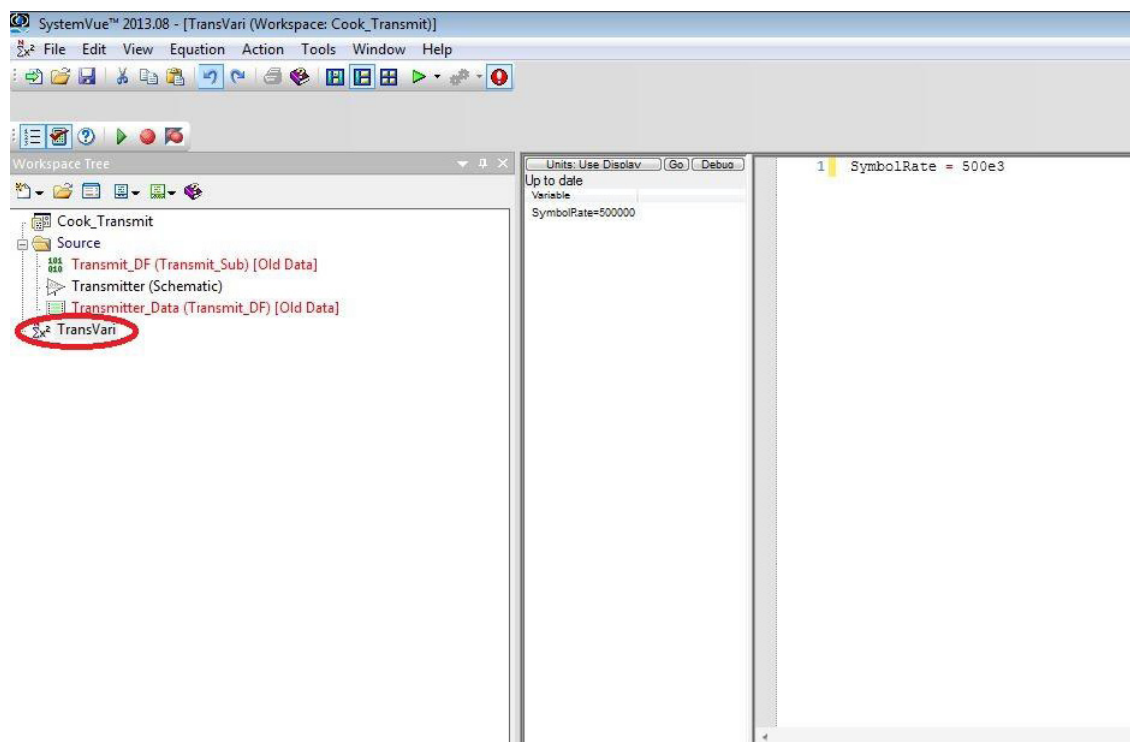


Рис. 5.10. Страница уравнений и объявление переменной SymbolRate.

4. Вернитесь к схеме и отредактируйте значение SampleRate блока SetSampleRate так, чтобы оно равнялось "SymbolRate".

Теперь пришло время реализовать фильтрацию формы импульсов для передатчика.

5. Найдите и установите фильтр "FIR" (фильтр с конечной импульсной характеристикой).
6. Отредактируйте фильтр FIR, для чего щёлкните на кнопку "Filter Designer" (конструктор фильтра) в нижней части страницы свойств фильтра FIR.
7. На вкладке Specifications (характеристики) выберите значение Low pass (ФНЧ) в выпадающем списке Filter Response (тип фильтра). Меню должно измениться и отобразить список конструкций фильтра. В разделе FIR выберите Raised Cosine (приподнятый косинус) и, щёлкнув OK, закройте окно.
8. Снова дважды щёлкните на фильтре FIR. Открывшееся окно будет выглядеть почти таким же. Измените модель (в пункте "Description" (описание) над "Model Help" (справка по модели)) в значение "LPF_RaisedCosine@Data Flow Models". Альтернативно можно просто напечатать текст "LPF_RaisedCosine@Data Flow Models" в поле ввода Model; эти шаги демонстрируют разные варианты, которые можно использовать для конструирования блока фильтра.

Здесь мы видим много параметров. Большинству из них мы дадим имена.

Перейдите в редактор уравнений, как показано на рис. 5.11, и создайте следующие переменные со следующими значениями.

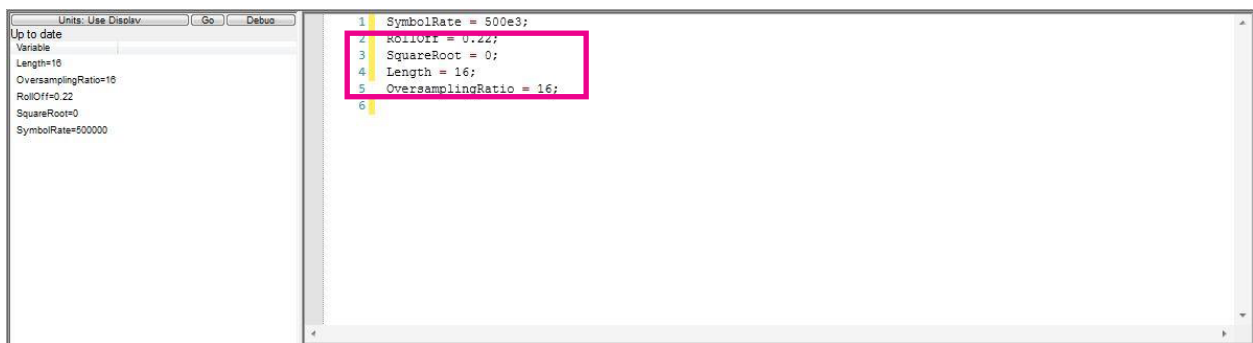


Рис. 5.11. Красный прямоугольник содержит переменные, добавленные на странице TransVari Equation.

После настройки переменных можно поместить их в фильтры с характеристикой приподнятого косинуса, которые мы планируем использовать.

9. Измените SymbolRate (символьная скорость) на "SymbolRate", RollOff (спад) на "RollOff" и SquareRoot (квадратный корень) на "SquareRoot". Измените LengthOption (вариант длины) на "2: Number of Symbols" (2:число символов), Length (длина) на "Length" и Interpolation (интерполяция) на "OverSamplingRatio".

Изменения показаны на рис. 5.12.

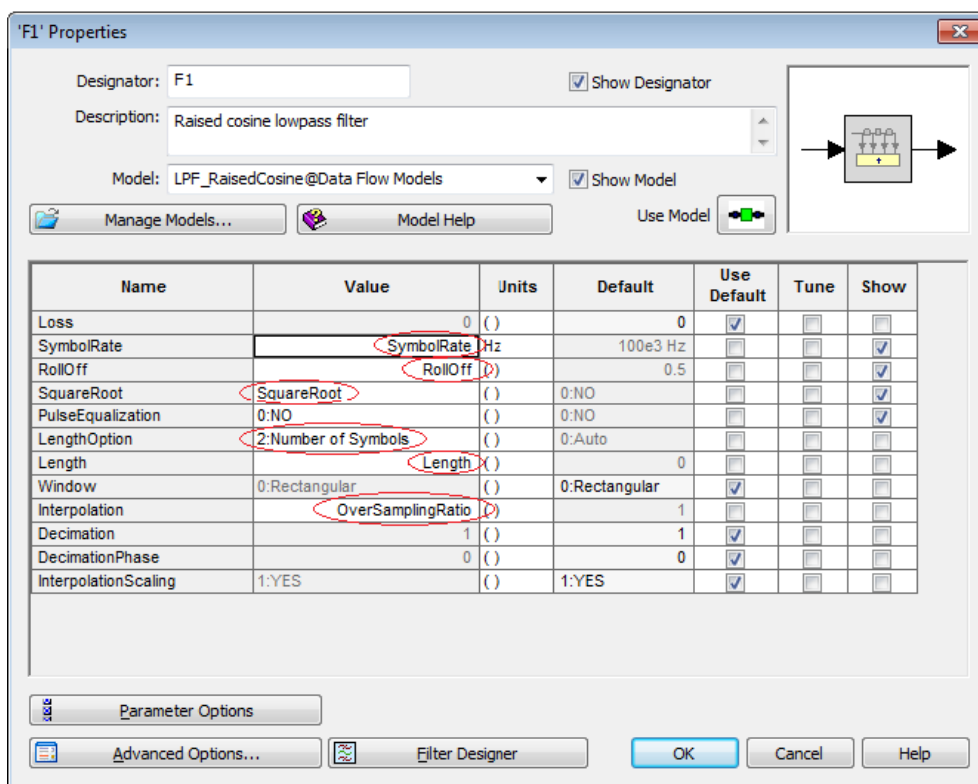


Рис. 5.12. Окно свойств ФНЧ с выделенными изменениями.

- Щёлкните на ОК и сделайте копию фильтра ФНЧ (щёлкните правой кнопкой и выберите Копировать/Ctrl+C). Вставьте копию в свою схему. Затем подключите отдельно каждый фильтр к выходным линиям Im и Re блока CxToRect.

После этого ваша схема должна выглядеть так, как на рис. 5.13.

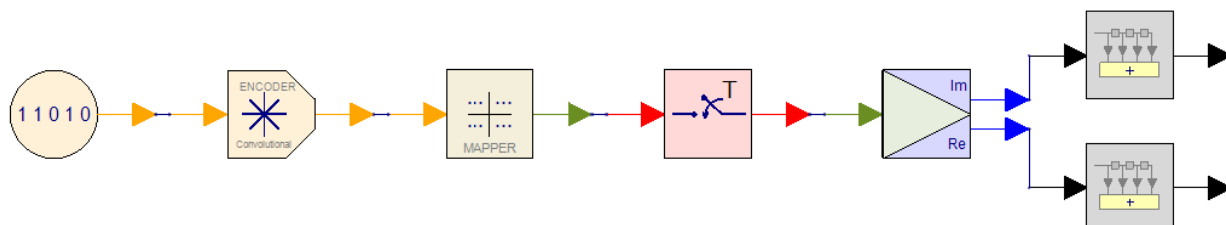
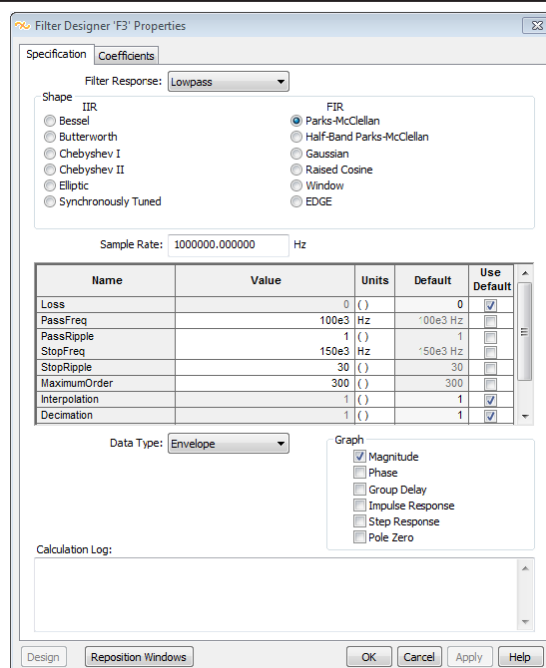
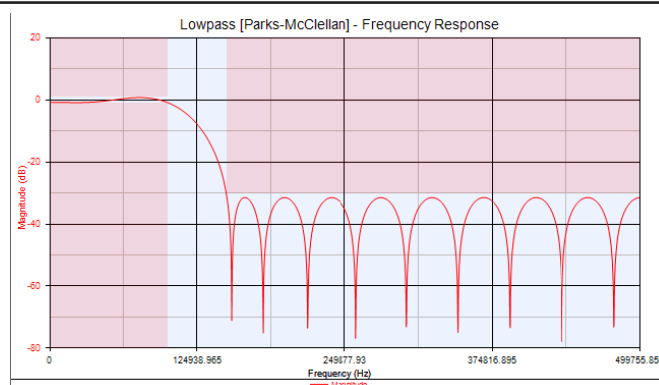


Рис. 5.13. Частично завершённый передатчик с фильтрами.

Проектирование фильтров. Проектирование фильтров в SystemVue даёт пользователю достаточно широкую свободу творчества. В комплект SystemVue входит универсальная опция Filter Designer (конструктор фильтров), которая открывается при изменении блока "Filter" (фильтр). Кнопка вызова конструктора фильтров расположена в нижней части окна. Само окно показано на рис. 5.14.



(a)



(б)

Рис. 5.14. Окно свойств конструктора фильтров (а), и график характеристик конструктора фильтров (б).

Слева от основного окна мы видим полный набор характеристик, форм и параметров фильтров. Имеются следующие стандартные характеристики фильтров: ФНЧ, ФВЧ, режекторный фильтр и полосовой фильтр, а также предусмотрена возможность создания собственных характеристик. Эта опция открывает функцию формы, которая позволяет создать необходимую характеристику фильтра. Под строкой Sample rate можно изменять такие параметры, как символьная скорость, наклон и передискретизация (интерполяция). Затем вы можете увидеть частотную характеристику вновь созданного фильтра, нажав кнопку Design (рис. 5.14б).

5.6 I/Q-модуляция

Теперь настроим I/Q-модуляцию.

1. Найдите блок "Modulator" (модулятор) и добавьте его в схему.
2. Подключите выходы фильтров к двум входам модулятора. Подключив модулятор, проверьте, что линия Im подключена к входу Q (фаза), а линия Re – к входу I (амплитуда).

Обратите внимание, что в верхней части блока модулятора есть ещё один вход. Этот вход используется для подачи опционального сигнала несущей, который можно использовать в процессе моделирования. Если на этот вход ничего не подано, то модулятор будет использовать стандартный сигнал несущей. Вы можете изменять частоту, меняя параметр FCarrier, и/или изменяя начальную фазу.

- Найдите в списке компонентов Oscillator (генератор) и установите его в схему, подключив к верхнему входу блока модулятора.

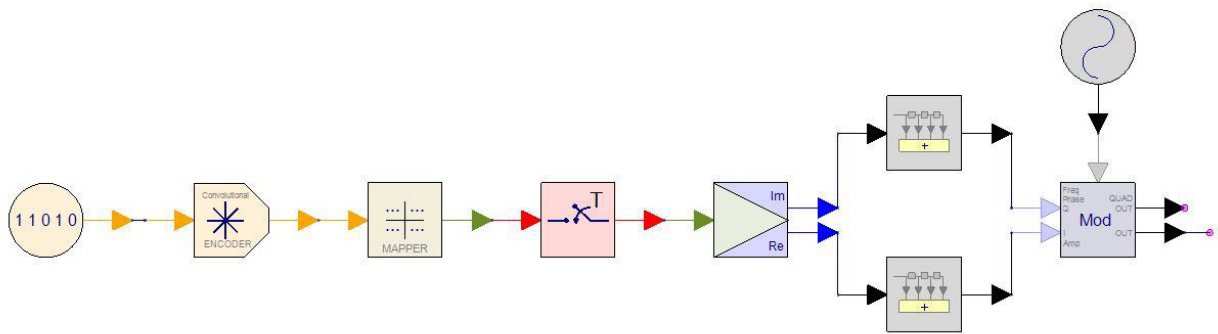


Рис. 5.15. Схема Cook_Transmit с добавленными блоками модулятора и генератора.

Ваш пример должен выглядеть так, как показано на рис. 5.15.

- Откройте окно TransVari Equations (уравнения TransVari) и введите переменные FCarrier = 800e6, TranPower = -10 и SamplingRate = SymbolRate*OverSamplingRatio, как показано на рис. 5.16.

```

8 FCarrier = 800e6;
9 TranPower = -10;
10 SamplingRate = SymbolRate*OverSamplingRatio;

```

Рис. 5.16. Добавление строк инициализации переменных на странице TransVari Equations.

Теперь, когда переменные определены, можно редактировать блок Oscillator.

- Установите Frequency (частота) на FCarrier, Power (мощность) на TranPower в дБм, Advanced Parameters (дополнительные параметры) на "1:YES" (1:Да), SampleRateOption на "1:Timed from SampleRate" (синхронизация от частоты дискретизации) и SampleRate на "SamplingRate", как показано на рис. 5.17.

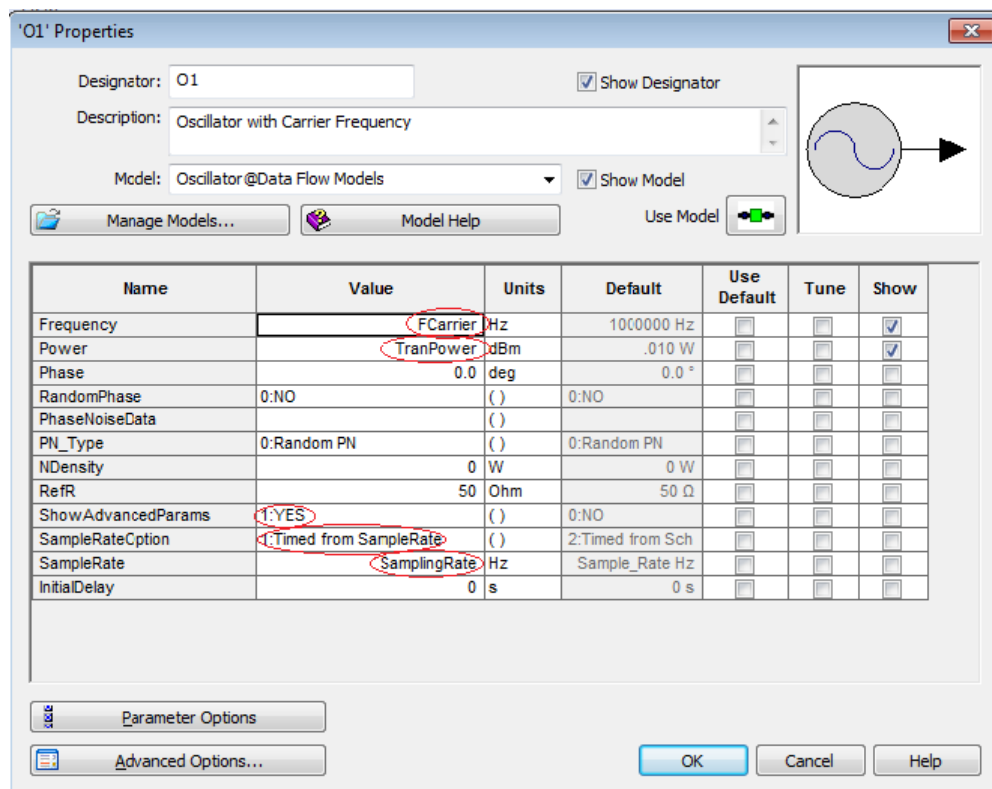


Рис. 5.17. Окно свойств генератора с выделенными изменениями.

I/Q-модуляция. Процесс, использующий модулирующий сигнал для изменения формы сигнала несущей. Модулирующий сигнал представлен данными, которые мы хотим передать. В данном случае мы собрали биты в определённом порядке, диктуемом заранее выбранной схемой формирования символов. Эта схема организована так, что точки сигнального созвездия позволяют легко создавать модулирующий сигнал. В данном случае I и Q могут представлять разные оси системы координат, и нанесённые на график точки превращаются в колебания. Затем эта информация используется для создания двух разных ортогональных сигналов, в совокупности образующих конечный сигнал, из которого можно извлечь полезную информацию.

Выбор типа модуляции в SystemVue относительно прост, просто измените сам блок. Имеются следующие возможности выбора: I/Q, амплитуда/фаза и амплитуда/частота.

5.7 Анализ результатов

И, наконец, мы хотим увидеть и измерить выходной сигнал нашего передатчика. Для этого лучше использовать ПО 89600 VSA. Программное обеспечение 89600 VSA представляет собой многофункциональный анализатор сигналов, чрезвычайно полезный для отладки и тестирования нашего передатчика.

1. Подключите стандартную нагрузку и “VSA_89600_Sink” к выходу модулятора “Mod Out” и запустите моделирование.

Подождите некоторое время, чтобы анализатор VSA начал взаимодействовать с САПР SystemVue. Когда VSA начнет работать:

2. Перейдите в “MesaSetup”, затем в “Measurement Type Vector” (вектор типа измерений), после чего щёлкните на “Digital Demod” (цифровая демодуляция).

Вы увидите весь набор символов на неправильно заполненном сигнальном созвездии и анализаторе спектра.

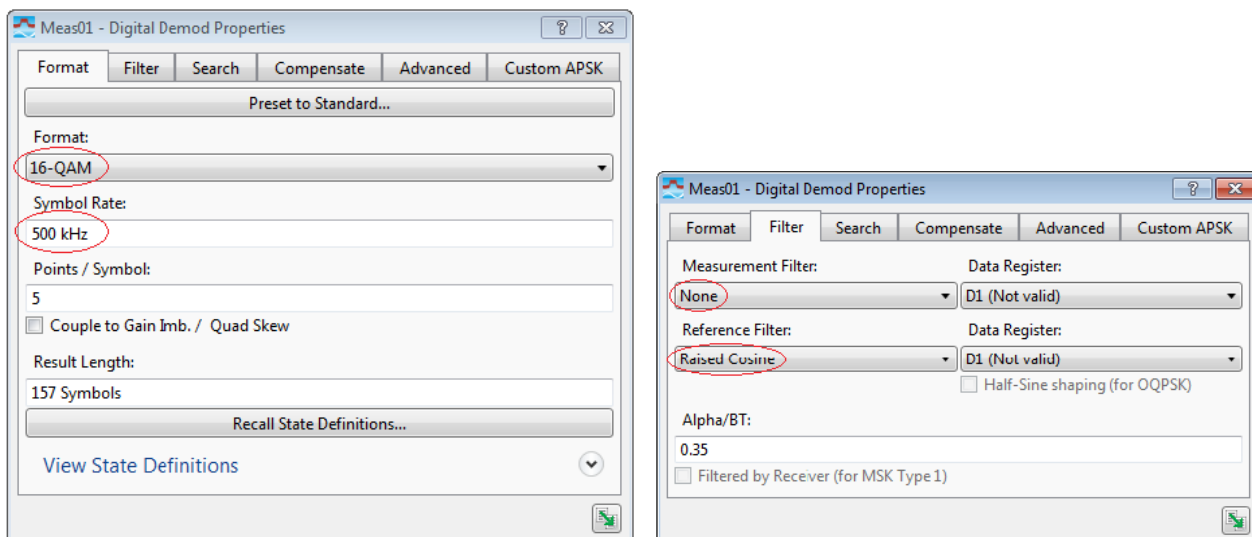


Рис. 5.18. Окно свойств цифровой демодуляции ПО VSA с выделенными параметрами, которые нужно изменить.

3. В окне MesaSetup перейдите в “Digital Demod Properties” (свойства цифровой демодуляции). Измените Format (формат) на 16-QAM, а Symbol Rate (символьную скорость) на 500 кГц. Эти изменения показаны на рис. 5.18.
4. Щелкните на вкладке “Filter” и установите Measurement Filter (измерительный фильтр) на “None” (нет). Убедитесь, что Reference Filter (эталонный фильтр) установлен на “Raised cosine” (приподнятый косинус), а затем остановите моделирование и запустите его снова.

Если ваш сигнал выглядит, как на рис. 5.19 – 5.21, то вы всё сделали правильно.

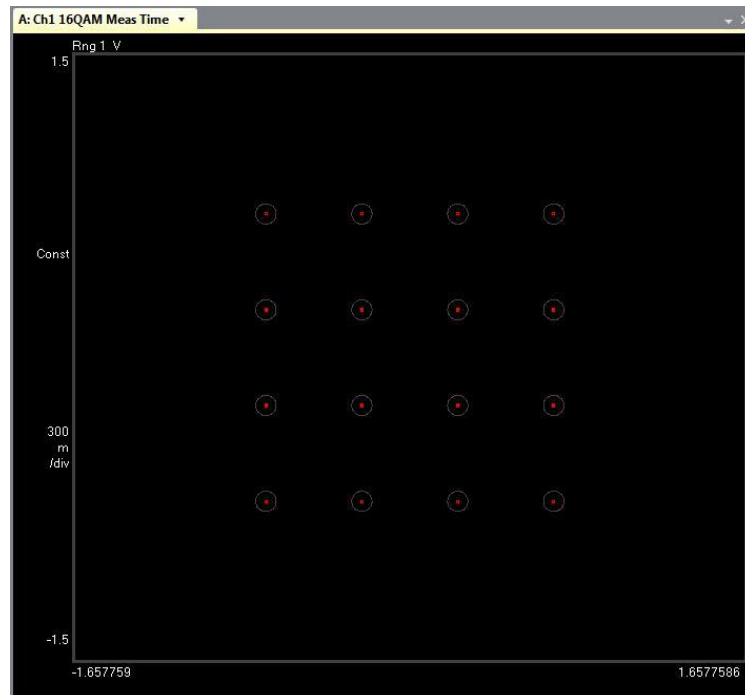


Рис. 5.19. Сигнальное созвездие передатчика 16 QAM в ПО 89600 VSA без жёлтой траектории символьной информации.

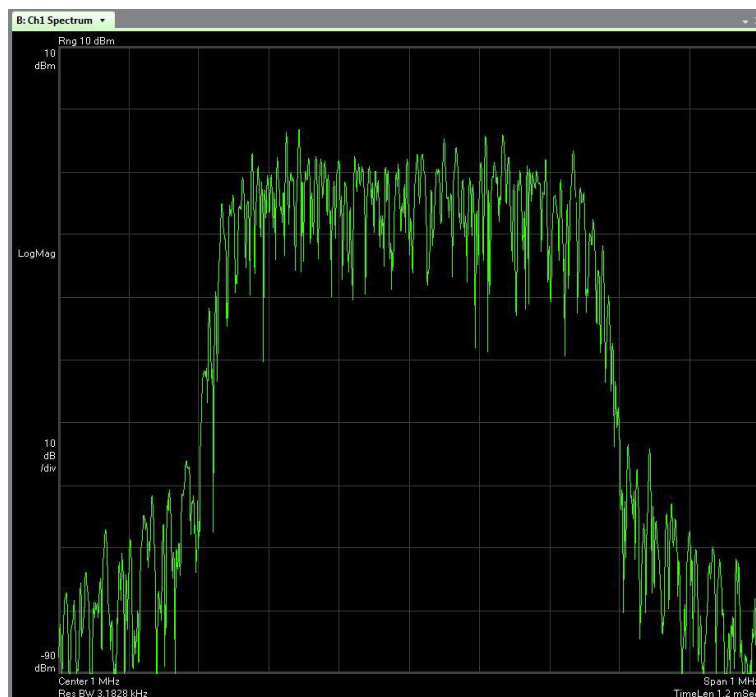


Рис. 5.20. Частотный спектр сигнала 16 QAM в ПО 89600 VSA.

Поскольку теоретически мы создали идеальный выходной сигнал без шумов, убедитесь, что среднеквадратическое значение ошибки не превышает 0,5 %. Амплитуда вектора ошибки (EVM) показана на рис. 5.21.

B: Ch1 16QAM Syms/Errs									
EVM	=	121.85	m%rms	276.61	m%pk	at	sym	235	
Mag Err	=	86.839	m%rms	-258.88	m%pk	at	sym	0	
Phase Err	=	89.415	mdeg	-402.02	mdeg pk	at	sym	213	
Freq Err	=	-2.3455	mHz						
IQ Offset	=	-97.703	dB						
Quad Err	=	10.876	mdeg						
				SNR (MER)	=	55.696	dB		
				Gain lmb	=	0.02	dB		
0	C5597AA4	6B041791	5514E88B	FF874BC8	C34F8CB4				
40	E9A3A792	048CE975	D991AE7F	60C93E40	31380388				
80	5A608FC9	130ECB24	6C8336E8	26543F8E	A5ACBA44				
120	FC334BE7	B1E57D76	95F77B4D	36F82BCE	177B89AB				
160	A365F922	9A007433	5A60A300	28E7CCFA	195CD458				
200	E1093822	BE3C5597	346DC1B6	82288D35	3DBC3E7C				
240	CFC42B09	DD805142	C738E68D	593B7F69	4EAF26F6				
280	853D42FD	C478F5EB	36E34A96	6E38B619	F21B902D				
320	EE4EF63F	16146853	EE4E337C	82A778E8	EDDD92BD				
360	F523B7F6	98F3429B	E3587945	90DA5BB9	46CBF806				
400	B4C42B9E	B9406DE5	D3B62337	DC30D16B	95EA856E				
440	B4835C3E	46F23678	3488309F	3659C9FF	E2FA2A2B				
480	065D75A4	73A324EC	CCF5						

Рис. 5.21. Сводка системных ошибок ПО 89600 VSA.

В результате ваша конечная модель должна выглядеть так, как показано на рис. 5.22.

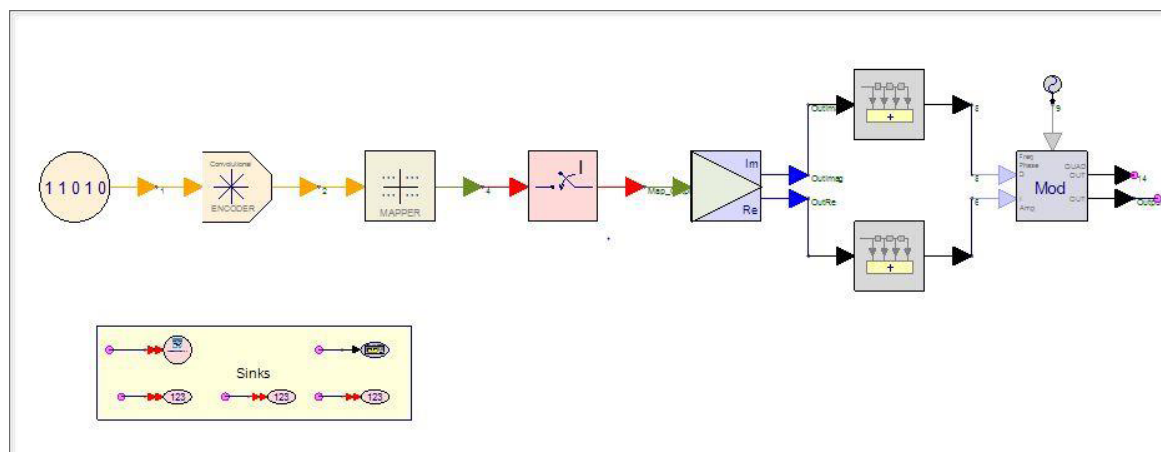


Рис. 5.22. Упрощённая конечная схема.

Глава 6. Обзор приёмника

6.1 Основы конструкции приёмника

Приёмник можно считать второй половиной полной цифровой системы радиосвязи. Он принимает входной сигнал и декодирует его. Сначала входной ВЧ сигнал преобразуется в более низкую промежуточную частоту (ПЧ) и демодулируется. Возможность демодулирования сигнала затрудняется такими факторами, как атмосферные помехи, посторонние сигналы и многолучевое распространение и затухание.

В общем случае обработка принимаемого сигнала состоит из следующих этапов.

1. Восстановление несущей частоты (синхронизация несущей).
2. Восстановление тактовой частоты символов (синхронизация символов).
3. Разложение сигнала на составляющие I и Q.
4. Определение значений I и Q для каждого символа (разделение).
5. Декодирование и устранение чередования.
6. Расширение до исходного битового потока.
7. Цифро-аналоговое преобразование (если нужно).

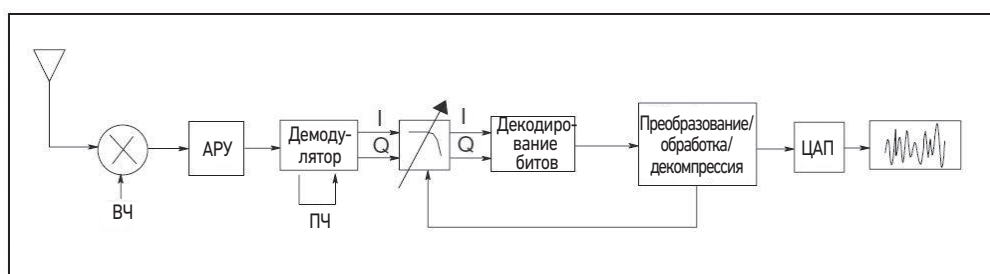


Рис. 6.1. Структурная схема приёмника.

Эти этапы показаны выше на рис. 6.1. Во многих современных системах передаваемый сигнал изначально является цифровым и таковым и остаётся. Он никогда не бывает аналоговым в смысле непрерывного аналогового сигнала, такого как звук. Основное отличие приёмника от передатчика заключается в проблеме восстановления несущей и тактовой (или символьной) частоты.

Для успешной демодуляции битов и восстановления переданной информации в приёмнике нужно корректно восстановить частоту и фазу тактовой частоты символов. Тактовая частота символов может иметь правильную частоту, но неправильную фазу. Если фаза символьной частоты соответствует переходам между символами, а не самим символам, то демодуляция может быть неудачной.

Обычно символьная частота имеет фиксированное значение, и это значение точно известно приёмнику и передатчику. Сложность заключается в согласовании символьных частот приёмника и передатчика по фазе. Для этого существует множество методов, и в большинстве систем применяется два или более таких метода. Если амплитуда сигнала во время модуляции меняется, приёмник может измерить эти изменения. Передатчик может передавать специальный синхросигнал или определённую последовательность битов, например, 101010101010, для «обучения» тактового генератора приёмника. В системах с импульсной несущей символьная частота может синхронизироваться по моментам включения несущей.

В передатчике всегда известна фаза ВЧ несущей и тактовая частота данных, поскольку они генерируются в самом передатчике. Приёмник лишён этого преимущества. Он может примерно оценить фазу несущей, но не имеет никакой информации о фазе символьной частоты. Сложная задача проектирования приёмника заключается в создании алгоритмов восстановления несущей и символьной частоты.

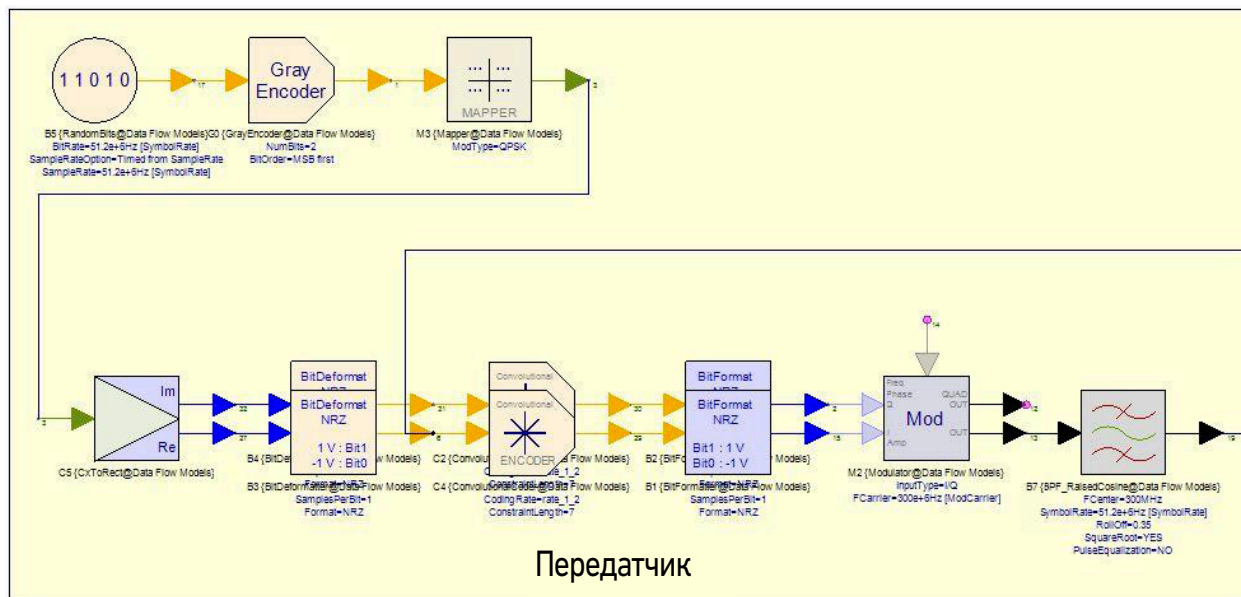
Из рис. 6.1 можно предположить, что для создания приёмника достаточно развернуть передатчик в обратную сторону. Что касается расположения функциональных блоков, передатчик и приёмник имеют до некоторой степени зеркальную конструкцию. В реальности для приёмника характерны более сложные проб-

лемы, в результате чего его блоки отличаются от соответствующих блоков передатчика, но, тем не менее, процесс создания приёмника весьма напоминает процесс создания передатчика. На рис. 6.2 показана схема SystemVue, взятая из примера QPSK_BER_Coded_Viterbi.

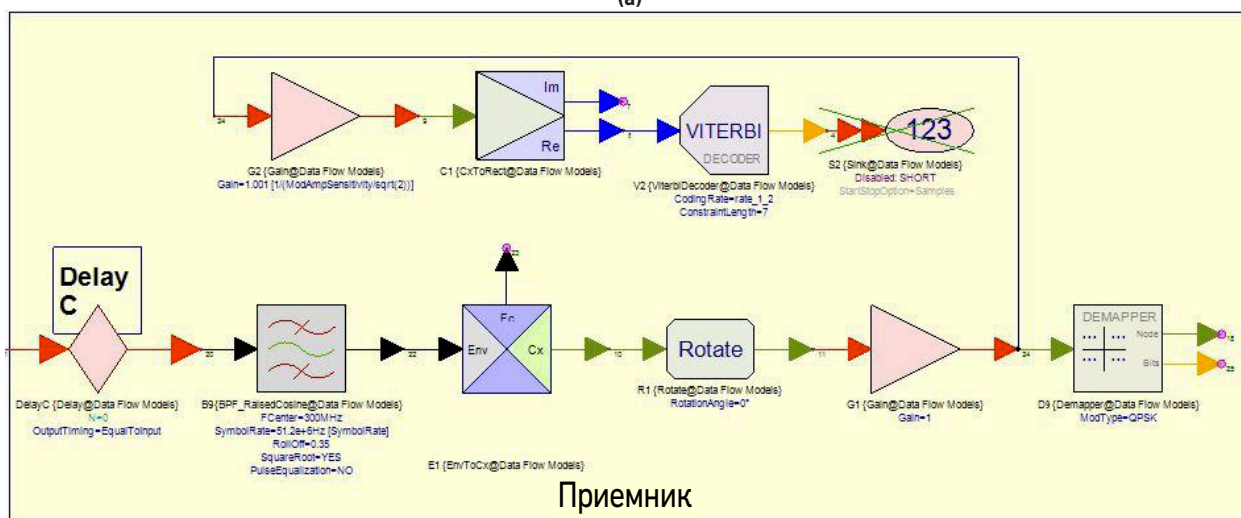
Пример. QPSK_BER_Coded_Viterbi

Этот пример можно найти так:

1. откройте страницу примеров Example;
2. откройте папку Comms (связь);
3. откройте папку BER, здесь и расположен пример.



(a)



(б)

Рис. 6.2. Схема из примера QPSK_Design. Схема передатчика (а), схема приёмника (б).

Как видите, хотя и существуют отличия структурных схем приёмника и передатчика, имеются достаточно заметные параллели, вследствие чего проектирование одного весьма напоминает проектирование другого. В связи с этим данная глава будет отличаться от предыдущей тем, что не будет детально описывать процесс создания приёмника. Вместо этого здесь будет проанализирован полностью функционирующий приёмник. Это поможет пользователям познакомиться с расширенными возможностями SystemVue и в то же время продемонстрирует простой альтернативный способ создания цифровой системы связи с самого начала. Тем не менее, если вы хотите получить больше творческой свободы, то можете использовать в качестве образца схему на рис. 6.2.

Пример. DigMod_Tx_Rx_AWGN

Чтобы найти этот пример, перейдите на страницу Open Example.

1. Откройте папку Baseband Verification (проверка модулирующего сигнала).
2. Откройте папку DigMod (цифровая модуляция).
3. Откройте папку Equalizer (эквалайзер). Здесь и находится искомый пример.

Примечание. Именем DigMod_Tx_Rx_AWGN обозначен пример цифровой модуляции, состоящий из передатчика и приёмника с аддитивным белым гауссовским шумом (известным как AWGN). AWGN представляет собой упрощённый набор условий, влияющих на канал во время передачи. В канал линейно добавляется белый шум с постоянной спектральной плотностью и гауссовским распределением амплитуды.

Основой наших примеров будет эта рабочая среда, иллюстрирующая сам приёмник и его модель в SystemVue. На рис. 6.3 показана система связи, смоделированная в SystemVue.

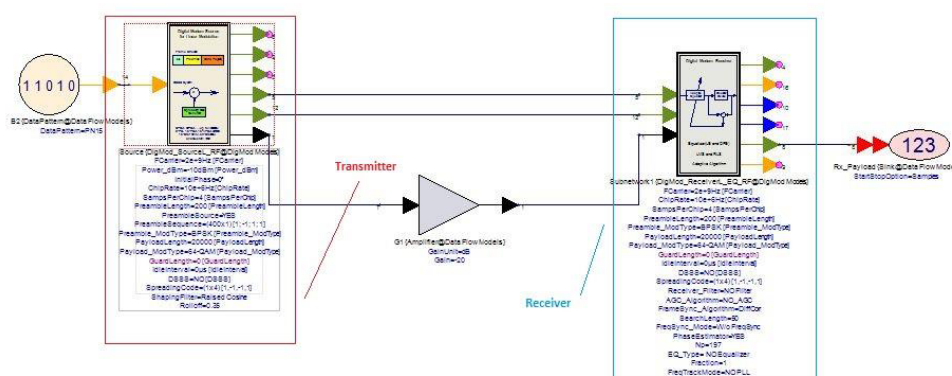


Рис. 6.3. Схема DigMod_TxRx_wo_FrameSync из примера рабочей среды DigMod_Tx_Rx_AWGN.

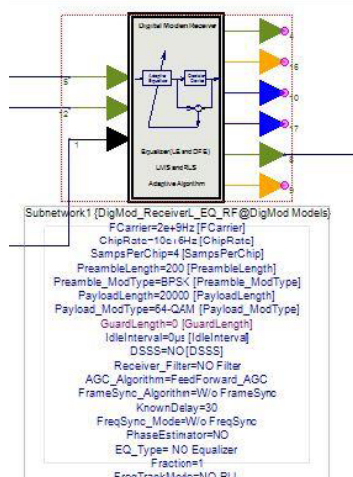
Показанная на рис. 6.3 схема взята из папки “1.1 WoFrameSyncEffect” и начинается со входа случайных битов, выхода на нагрузку и двух больших блоков между ними. Левый блок представляет собой полный передатчик, а правый блок – полный приёмник. Каждый из этих блоков является источником и приёмником ВЧ сигнала с линейной модуляцией. Эти блоки могут полностью настраиваться с помощью набора параметров. Сами по себе они выступают в роли образцовой модели для моделирования цифровой системы связи, которая будет полезна для анализа компонентов приёмника. Позже в примере DigMod_Tx_Rx_AWGN мы увидим немного видоизменённый вариант этой схемы, но функционально эти схемы будут идентичны.

Большие блоки приёмника и передатчика, использованные для создания этой схемы, можно найти в библиотеке компонентов DigMod под именами DigMod_SourceL_EQ_RF и DigMod_ReceiverL_EQ_RF соответственно.

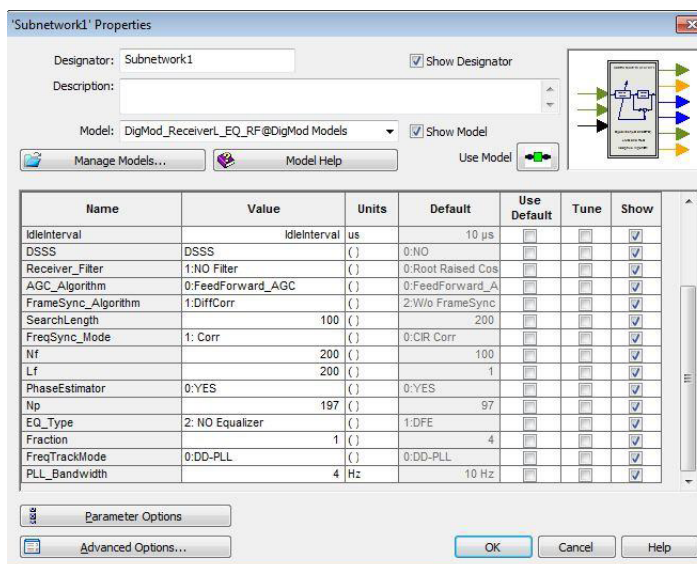
6.2 Обзор схемы приёмника

Основное внимание в этой главе будет уделено модели DigMod_ReceiverL_EQ_RF. Этот блок представляет собой полную модель приёмника с гибко настраиваемыми параметрами. Назначение этого раздела заключается в описании приёмника и методов эффективного измерения параметров и тестирования системы связи в САПР SystemVue. Подход, используемый в этой главе, отличается от предыдущего. Мы будем исходить из предположения, что пользователь обладает достаточным опытом создания моделей из функциональных блоков и уравнений, и сосредоточимся на некоторых расширенных блоках, предлагаемых SystemVue. Кроме того, в этой главе будет показано решение некоторых проблем, с которыми можно столкнуться в процессе создания цифровой системы связи.

Обзор приёмника



(а)



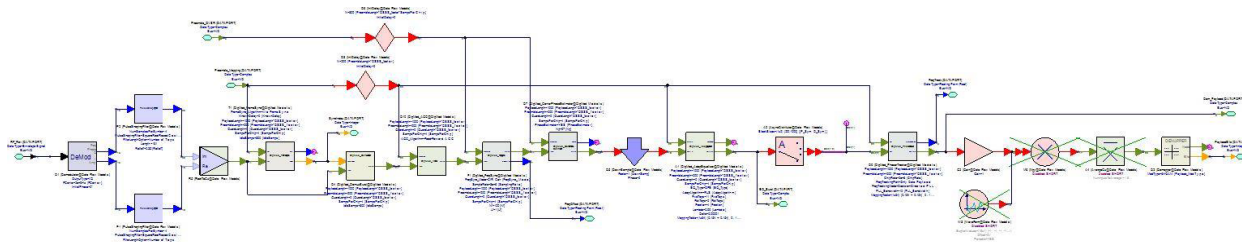
(б)

Рис. 6.4. Структурная схема приёмника (а). Свойства блока приёмника (б).

На рис. 6.4б показаны настраиваемые параметры DigMod_ReceiverL_EQ_RF, такие как типы модуляции (полезного сигнала и преамбулы), алгоритмы синхронизации фреймов, определители фазы и режимы синхронизации частоты. Кроме того, этот блок может выводить разные потоки данных, такие как биты полезных данных, демодулированные полезные данные, смещение частоты и индекс синхронизации. Мы можем пойти ещё дальше и исследовать как это работает. Для этого просто щёлкните на блоке правой кнопкой и выберите следующие пункты:

1. Open (открыть)
2. Model/Subcircuit (модель/подсхема)

Здесь вы увидите подсхему приёмника (рис. 6.5). Заметьте, насколько она соответствует некоторым ключевым идеям и характеристикам идеальной подсхемы.



Демодуляция

Коррекция синхронизации, фазы
и частоты

Расформи-
рование
символов

Согласованная
фильтрация

Рис. 6.5. Подсхема приёмника.

Примечание. Если вы хотите что-нибудь изменить в этом примере или внутри блоков высокоуровневого источника и приёмника, не изменяйте сам пример и не сохраняйте изменения. Вместо этого создайте пустую схему или подсхему, скопируйте содержимое рабочей среды/схемы/файлов в эту пустую схему и затем используйте вновь созданную копию в качестве «испытательного стенда» или нового проекта.

Пример. Папка 1.1, WoFrameSyncEffect

Демодуляция представляет собой процесс, обратный процессу модуляции. Она извлекает вложенную информацию из сигнала несущей. Демодулировать входной сигнал можно разными способами, но самым распространённым является умножение входного сигнала на ортогональные волновые функции.

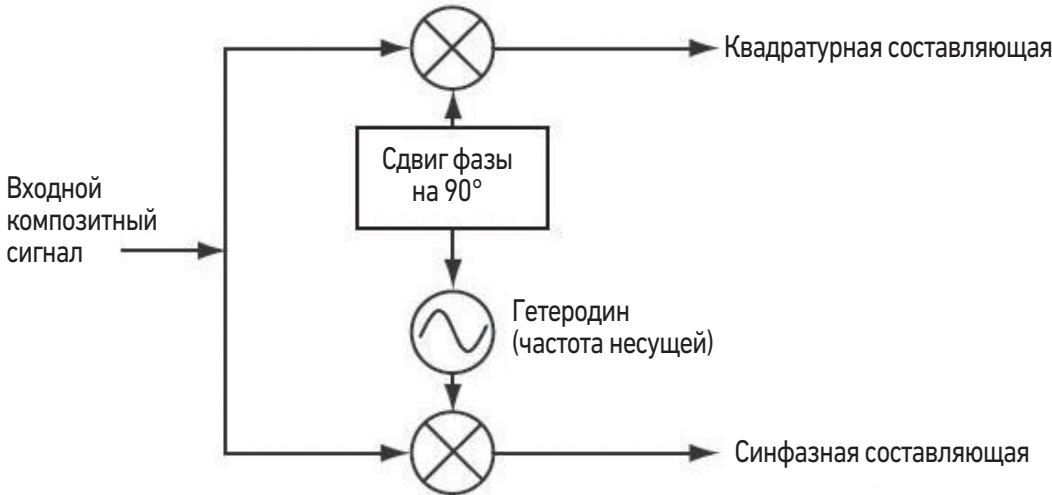


Рис. 6.6. Графическое представление демодуляции сигнала несущей.

Квадратурная амплитудная демодуляция выполняется с помощью двух ортогональных сигналов, как показано на рис. 6.6. Когда сигнал несущей умножается на ортогональный сигнал, каждый из ортогональных сигналов несёт половину необходимой информации, извлекаемой из модулированного сигнала. Из одного сигнала мы можем извлечь значения I и Q, что позволяет восстановить символы, а затем и сами данные. Это значит, что если при демодуляции фаза задающего генератора не соответствует фазе принимаемого сигнала, то уровни выходных сигналов будут искажены. Подробнее это поясняется в разделе 6.6 «Коррекция фазовых ошибок и ошибок синхронизации».

Для настройки процессов модуляции и демодуляции можно отредактировать функциональные блоки, изменить преамбулу и полезные данные. В результате получатся индивидуальные схемы модуляции и демодуляции (рис. 6.7).

PreambleLength	PreambleLength	PreambleLength	PreambleLength
PreambleSource	1:YES	Preamble_ModType	Preamble_ModType
PreambleSequence	Preamble(:)	PayloadLength	PayloadLength
Preamble_ModType	Preamble_ModType	Payload_ModType	Payload_ModType
PayloadLength	PayloadLength		
Payload_ModType	Payload_ModType		

(a)

(б)

Рис. 6.7. Местоположение исходного блока для настройки преамбулы (а). Местоположение блока приёмника для настройки преамбулы (б).

Необходимым условием успешной демодуляции сигнала является соответствие частоты несущей частоте задающего генератора ортогональных сигналов, поэтому, редактируя эти блоки, следите за соблюдением этого условия. Затем две демодулированные составляющие I/Q фильтруются.

6.4 Согласованная фильтрация

Фильтрация в передатчике преследует цель ограничения полосы передаваемого сигнала, чтобы он не подвергался воздействию других сигналов и внешних шумов. Однако даже при передаче узкополосного сигнала, по пути к приёмнику он подвергается воздействию белого шума. Фильтр в приёмнике отфильтровывает

шум, добавленный в процессе передачи сигнала по каналу. Этот фильтр, безусловно, полезен, но нужен не всегда, и должен включаться по мере необходимости. Обычно фильтр приёмника строится на базе фильтра источника и обеспечивает полное согласование, что позволяет исключить ошибки данных и повышенный шум.

В этой модели мы видим, что передатчик использует для формирования выходного сигнала фильтр с характеристикой приподнятого косинуса, а приёмник не использует фильтров вообще. Это связано с тем, что фильтр с характеристикой приподнятого косинуса является полосовым фильтром. Он пропускает только нужные сигналы на определённой частоте. В связи с этим согласованный фильтр на стороне приёмника не нужен. Однако если мы захотим, то сможем использовать фильтр (см. рис. 6.8).

DSSS	0:NO
Receiver_Filter	0:Root Raised Cosine
Rolloff	0.35

Рис. 6.8. Настройки параметров фильтра.

6.5 Синхронизация

При передаче сигнала по эфиру на большие расстояния неизбежно возникает некоторая задержка. Эта задержка может создавать определённые трудности, поскольку она изменяет время прихода символов и фазу несущей. Чтобы решить эту проблему, мы можем реализовать метод минимизации смещения тактовой частоты. Однако, измеряя задержку, вызванную средой и расстоянием, мы должны отдельно измерять время прихода символов и фазу несущей. Это связано с разной амплитудой частот этих сигналов. Сначала давайте рассмотрим ошибки времени поступления символов.

В нашем примере будет использоваться базовая схема приёмника и передатчика, показанная во введении. Сначала она будет настроена без синхронизации фреймов (рис. 6.9).

AGC_Algorithm	0:FeedForward_AGC
FrameSync_Algorithm	2:W/o FrameSync
KnownDelay	80

Рис. 6.9. Здесь показаны параметры для настройки синхронизации фреймов. В настоящее время синхронизация фреймов отключена.

Давайте рассмотрим, как проявляются ошибки времени прихода символов. Для подробного изучения этой проблемы нужно сначала исследовать входной сигнал. Для этого мы используем сигнальное созвездие, которое в данном случае представляет собой 64 QAM. Сигнальное созвездие без синхронизации фреймов показано на рис. 6.10.

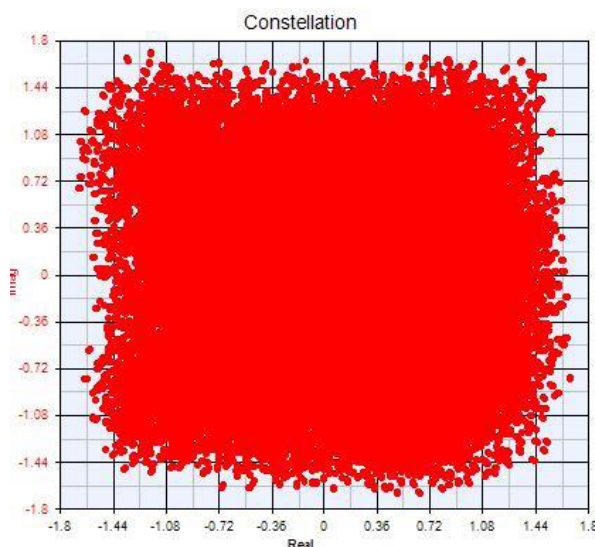


Рис. 6.10. Сигнальное созвездие без синхронизации фреймов.

На рис. 6.10 вы видите целую кучу точек. В данном случае это говорит о нарушении синхронизации фреймов. Обычно, если бы сигнал обрабатывался корректно, мы бы увидели группы точек вокруг центров сигнального созвездия, представляющих точки данных. Однако для этого приёмник должен получать символы в нужные моменты времени, синхронно с символьной частотой. Если частота дискретизации не соответствует частоте структуры фреймов, мы считываем точки из сигнала несущей, получая значения в соответствующих точках и между ними.

Примечание. В данном случае такое размытие точек означает рассинхронизацию символьной частоты, но в других случаях причина может оказаться другой. Подобная проблема может быть вызвана одной из трёх причин (или их комбинацией): рассогласованием скорости сигнала, несоответствием фильтров приёмника и передатчика и рассинхронизацией сигнала несущей.

Для устранения этой проблемы можно просто включить алгоритм синхронизации фреймов. Чтобы сделать это в SystemVue, просто отредактируйте и измените значение FrameSync_Algorithm на "1:DiffCorr", как показано на рис. 6.11.

AGC_Algorithm	0:FeedForward_AGC
FrameSync_Algorithm	1:DiffCorr
SearchLength	100

Рис. 6.11. Изменение алгоритма синхронизации фреймов на дифференциальную корреляцию.

Примечание. Другой способ изменения блоков заключается в том, чтобы дважды щелкнуть на синем тексте параметра под именем и описанием в структурной схеме.

Пример. Папка 1.2, WoAGC

Когда мы включаем дифференциальную корреляцию и синхронизация восстанавливается, мы можем наблюдать существенное улучшение. На рис. 6.12а видно, что теперь данные считываются корректно и соответствуют структуре сигнального созвездия 64 QAM. Однако если уменьшить масштаб, то будет видно, что сами точки распределены по амплитуде неправильно, как показано на рис. 6.12б.

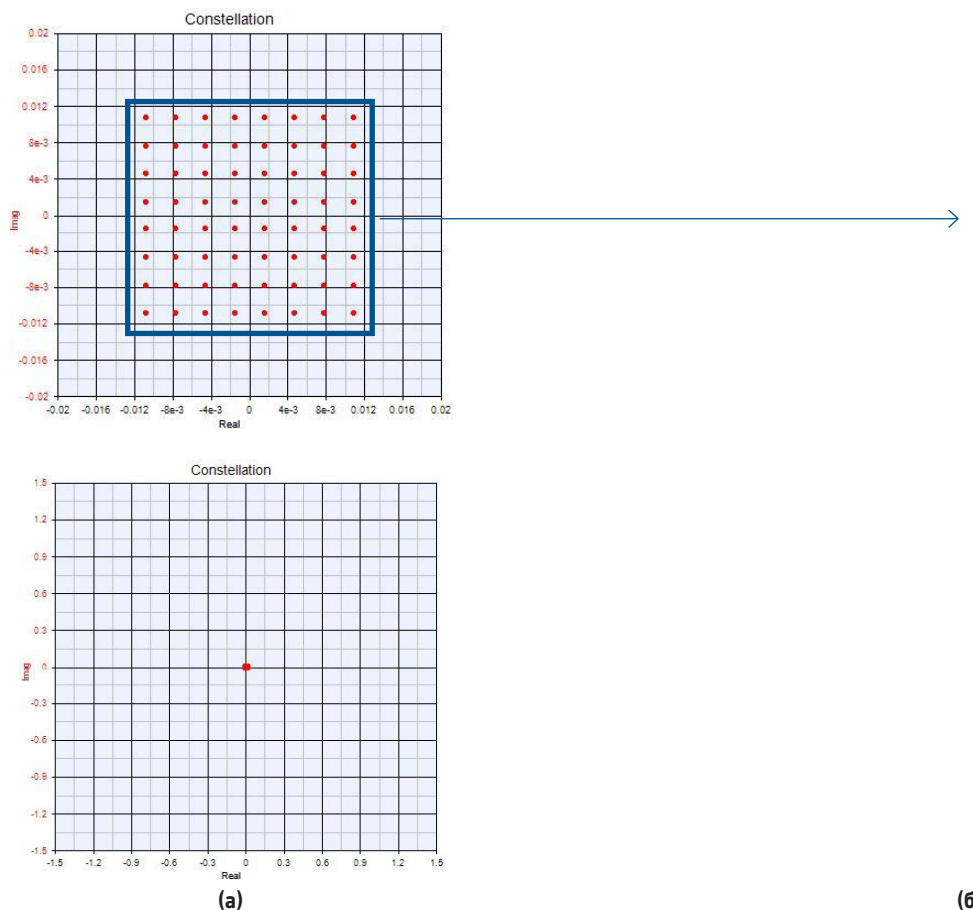


Рис. 6.12. На этом рисунке показаны разные масштабы модели WoAGC. На первой диаграмме показано сигнальное созвездие в процессе моделирования (а). На второй диаграмме выполнено сравнение сигнального созвездия в соответствующем масштабе (б).

Это связано с тем, что при прохождении сигнала через НЧ и ВЧ фильтры в приёмнике и передатчике его мощность ослабляется. Кроме того, сигнал ослабляется при прохождении через среду передачи, т.е. по кабелю или эфиру.

Чтобы решить эту проблему, нужно включить или изменить алгоритм автоматической регулировки усиления (АРУ), который восстанавливает значения мощности до их исходного уровня и позволяет им занять корректные области сигнального созвездия.

Эти изменения показаны на рис. 6.13.

Receiver_Filter	1:NO Filter
AGC_Algorithm	0:FeedForward_AGC
FrameSync_Algorithm	1:DiffCorr

Рис. 6.13. Пример включения алгоритма АРУ.

Результирующий эффект показан на рис. 6.14, который демонстрирует усиленное и нормализованное по амплитуде сигнальное созвездие.

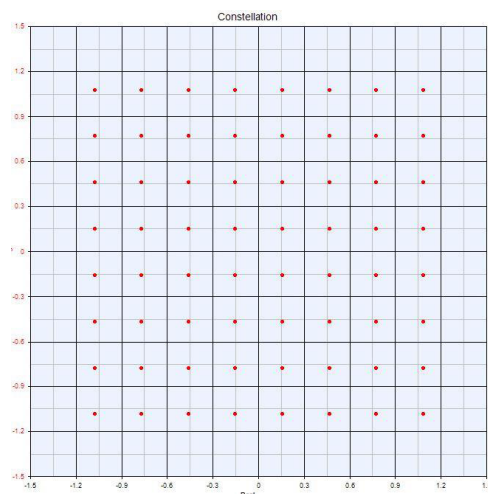


Рис. 6.14. Сигнальное созвездие с включённой АРУ.

6.6 Коррекция фазовых ошибок и ошибок синхронизации

Наиболее широко применяемая техника коррекции называется фазовой автоподстройкой частоты (ФАПЧ). ФАПЧ использует фильтр обратной связи, через который сигнал подаётся на генератор, управляемый напряжением (ГУН). ФАПЧ работает по принципу, описываемому уравнением:

$$\int_{-\infty}^{\infty} \sin[\theta(t)] \cos[\theta(t)] dt = 0 \quad \text{Е.6.1}$$

Из уравнения Е.6.1 следует, что если фазовая ошибка имеет некоторую величину, не ортогональную исходному сигналу, то мы можем оценить эту фазовую ошибку.

Передача сигнала из передатчика в приёмник вызывает фазовый сдвиг сигнала, который сказывается на сигнальном созвездии.

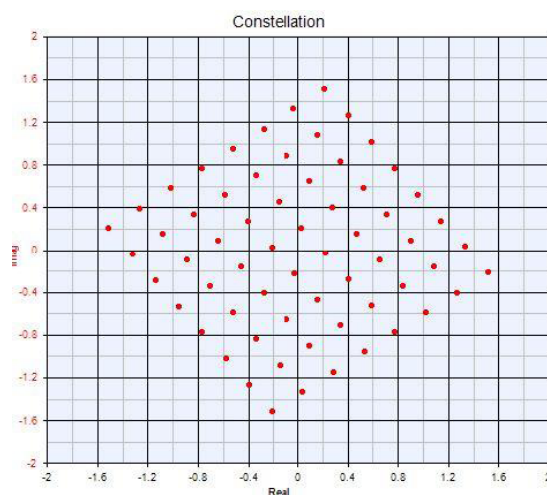


Рис. 6.15. Сигнал 64 QAM, повёрнутый на 135 градусов.

Эта проблема хорошо видна на рис. 6.15. Чтобы решить такую проблему, мы просто устанавливаем значение Phase Estimator (оценка фазы) в Yes (да) (рис. 6.16).

FreqSync_Mode	3:W/o FreqSync
PhaseEstimator	0:YES
Np	197

Рис. 6.16. Изменение режима оценки фазы.

Пример. Папка 2.2, CarrierPhaseEstimator

Теперь мы получаем сигнальное созвездие с исправленным сдвигом фазы, показанное на рис. 6.17.

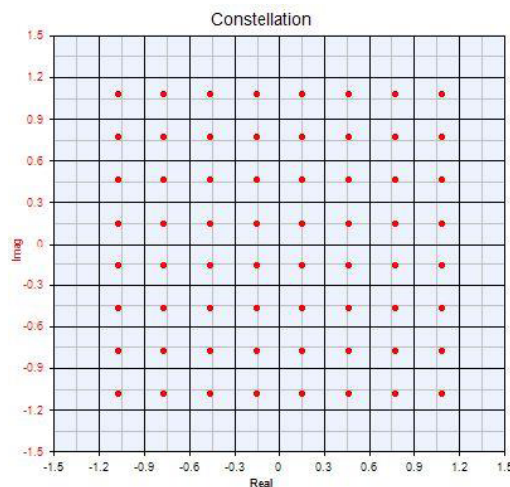


Рис. 6.17. Исправленное сигнальное созвездие, которое до этого имело сдвиг фазы.

6.7 Коррекция частотной ошибки

Пример. Папка 3.1, CarrierFrequencyOffsetEffect

Если выборки сигнала делаются с частотой, не соответствующей частоте несущей, мы называем это смещением частоты. Величина погрешности и уход частоты дискретизации вверх или вниз от частоты несущей обычно определяют скорость перемещения точек и направление, в котором они движутся. Пример смещения частоты показана на рис. 6.18, где частота несущей смещена на 50 Гц.

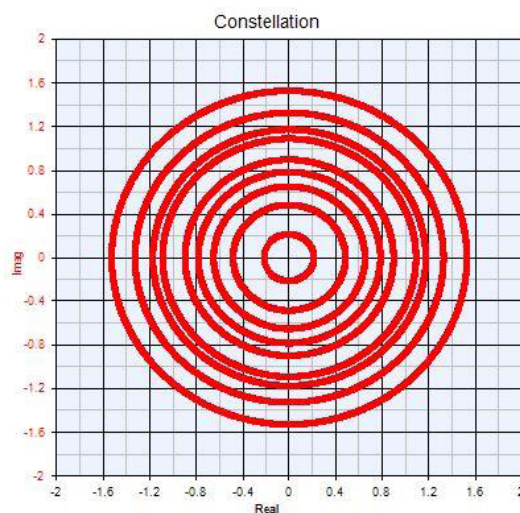


Рис. 6.18. Влияние смещения частоты на сигнальное созвездие.

Чтобы исправить смещение частоты, мы можем включить алгоритм синхронизации частоты несущей (рис. 6.19). Установка значения FreqSync_Mode на Corr (корреляция) исправляет сигнал и даёт диаграмму, напоминающую сигнальное созвездие, но приводит к небольшому сдвигу частоты из-за остаточной ошибки.

Другим возможным вариантом является изменение значения FreqSync_Mode на CIR_Corr (корреляция импульсной характеристики несущей), что устраняет ошибку и даёт лучшую диаграмму.

Пример. Папка 3.2, CarrierFrequencySync

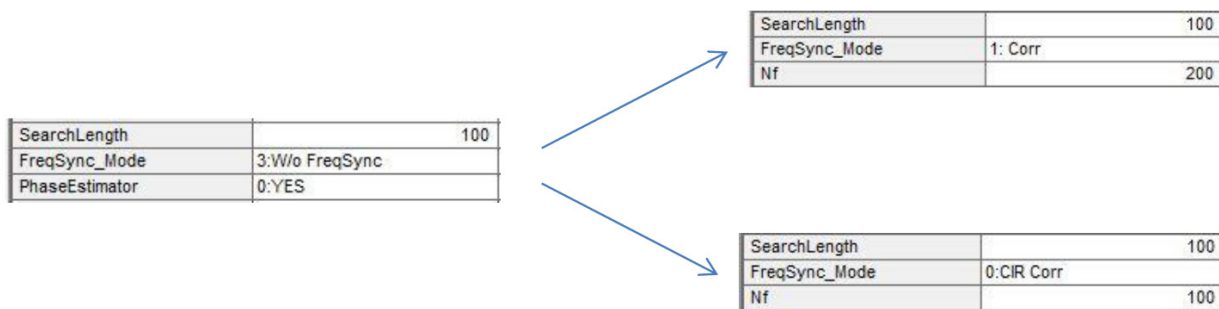


Рис. 6.19. Различные варианты режима синхронизации частоты.

Это приводит к следующему результату:

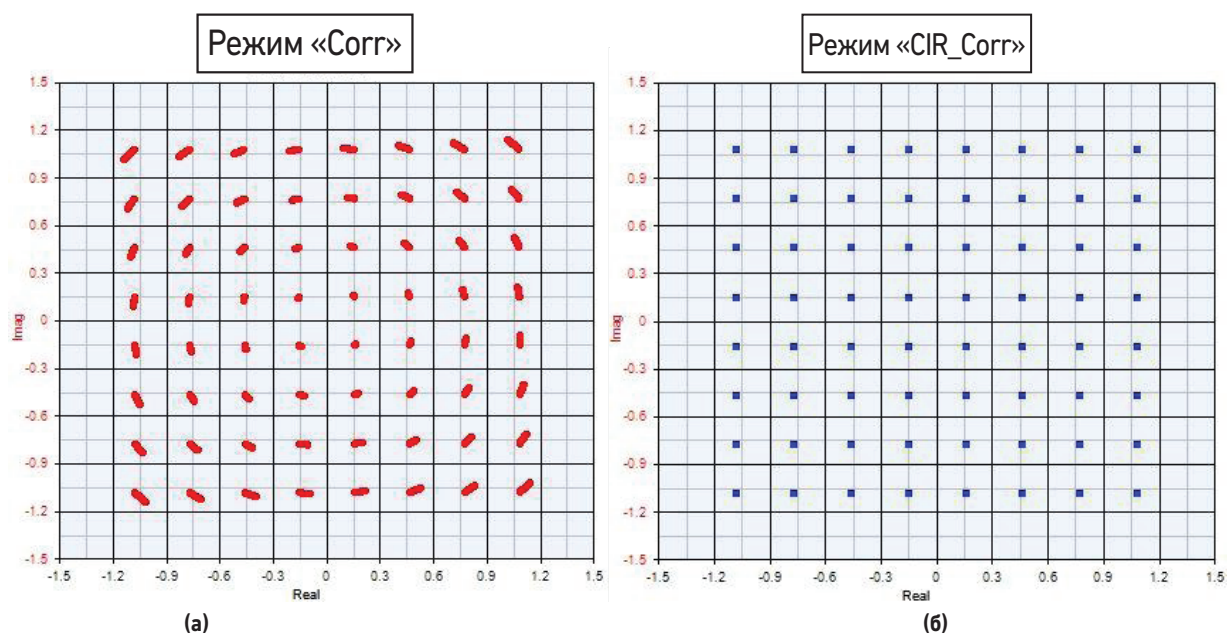


Рис. 6.20. Результаты применения различных методов коррекции смещения частоты. Сигнальное созвездие в режиме «Correlation» (a) и в режиме «CIR Correlation» (б).

Взглянув на рис. 6.20, обратите внимание на то, как смещение для корреляции (a) отличается от смещения для импульсной характеристики несущей (CIR) (б). В первом случае сигнальное созвездие медленно поворачивается по часовой стрелке, пока не достигнет положения CIR_Corr. Это связано с тем, что методу Corr необходимо время на реакцию обратной связи, как показано на рис. 6.21.

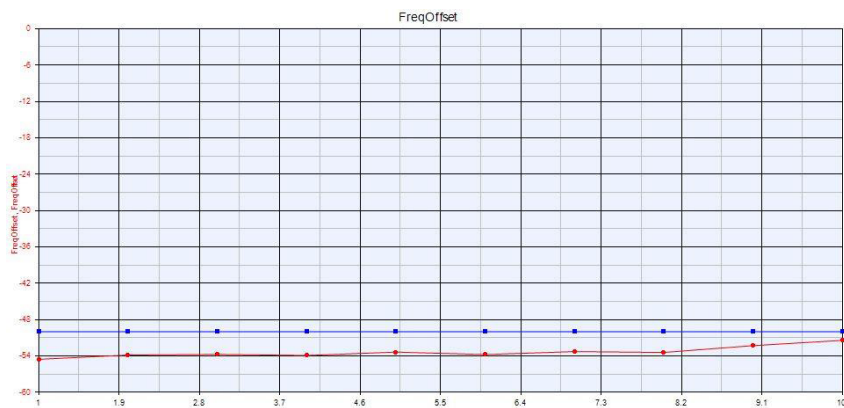


Рис. 6.21. Смещение, выполненное приёмником для компенсации смещения частоты передатчика.

Пример. Папка 3.3, CarrierFrequencyTracking_PLL

Теперь мы покажем, что происходит при малом смещении частоты. Для этого можно просто включить смещение ФАПЧ и рассчитать смещение частоты (рис. 6.22).

Fraction	1
FreqTrackMode	0:DD-PLL
PLL_Bandwidth	60

Рис. 6.22. Здесь показано, как выбрать начальную частоту и опцию PLL_Bandwidth.

Смещение ФАПЧ от синхронизации до нужной частоты может занять разное время, в зависимости от того, насколько смещается сигнал от демодулируемого сигнала. Кроме того, этот метод устраняет смещение несущей, как показано на рис. 6.23.

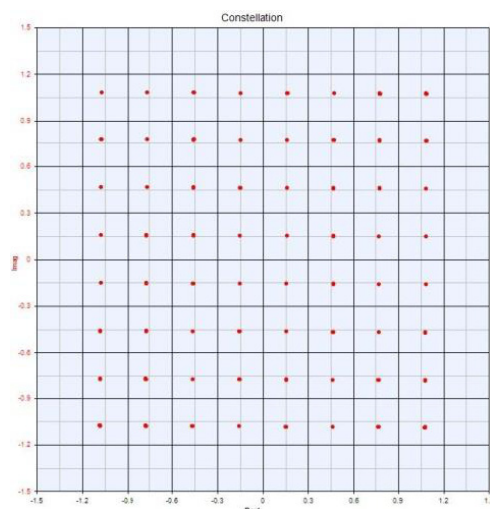


Рис. 6.23. Влияние ФАПЧ на сигнал со смещением частоты.

Процесс коррекции ошибки показан на рис. 6.24.

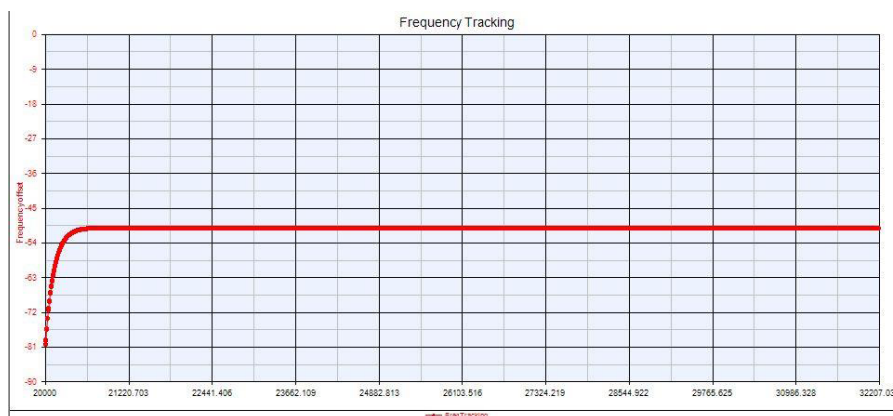


Рис. 6.24. График изменения частоты ФАПЧ, меняющей смещение частоты во времени.

На рис. 6.24 видно, что фаза постоянно подстраивается системой обратной связи в течение нескольких периодов дискретизации, пока не найдёт нужную частоту.

Пример. Папка 3.4, CarrierFreqSync_and_Tracking

А что произойдёт при большом смещении? Смещение ФАПЧ работает хорошо, если известна частота, но если частота неизвестна, вы получите небольшие следы смещения частоты. Это связано с тем, что алгоритм не

может подойти к смещению частоты достаточно быстро. Это справедливо и для корреляции. Чтобы учесть это явление, нужно использовать комбинацию смещения ФАПЧ с CIR_Corr/Corr. Например, большое смещение частоты 24990,9 может нарушить работу любого из этих методов, но если методы объединить, то смещение ФАПЧ и корреляция помогут приёмнику подойти к значению смещения. Метод синхронизации частоты корреляции выступает в роли грубого детектора, устанавливая смещение частоты близко к реальному значению. Смещение ФАПЧ, использующее отслеживание частоты, является более тонким методом синхронизации, который устраняет смещение частоты, когда оно уже близко к нужному значению (рис. 6.25).

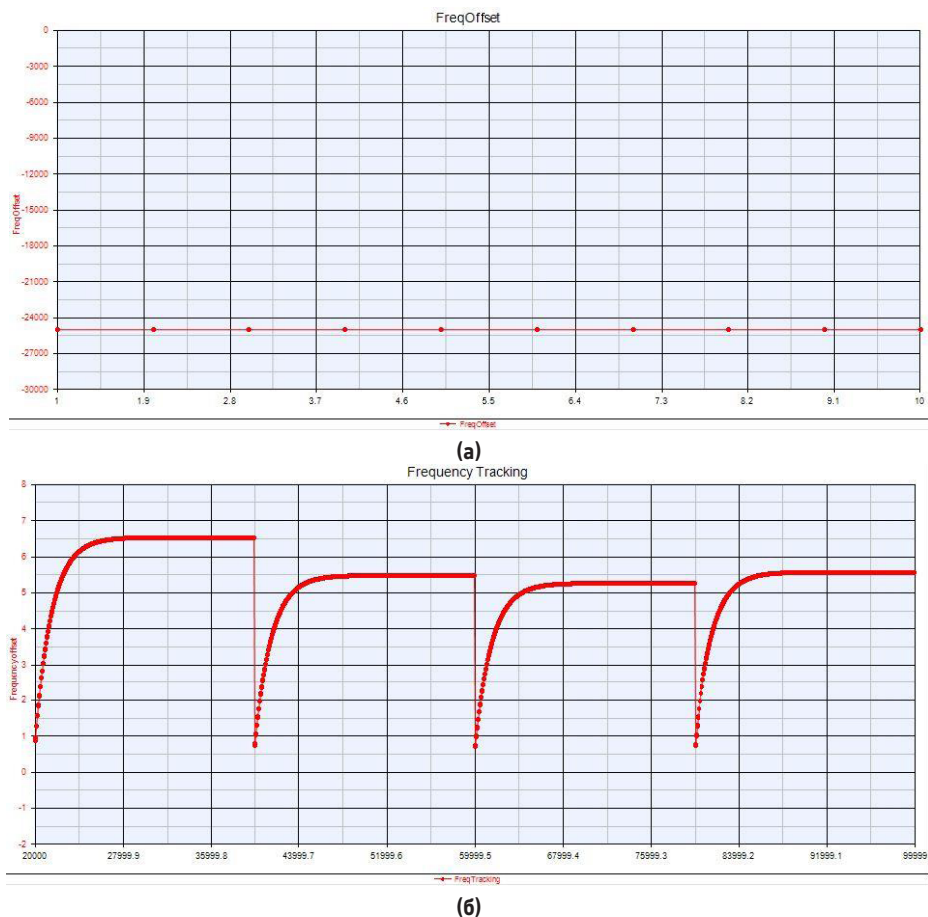


Рис. 6.25. Значение смещения частоты, полученное методом корреляции (а). Одновременное отслеживание частоты системой ФАПЧ уменьшает ошибку (б).

Результат комбинации синхронизации частоты несущей с отслеживанием частоты показан на рис. 6.26.

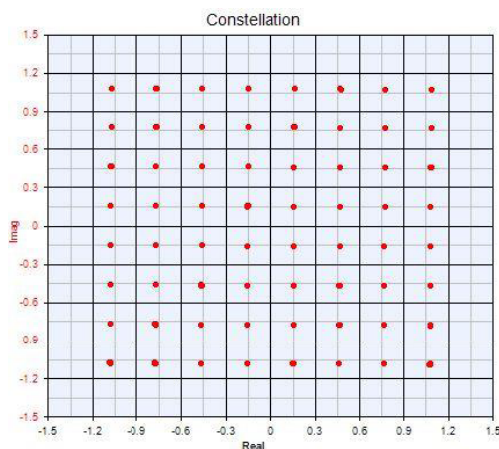


Рис. 6.26. Результирующее сигнальное созвездие после применения комбинации ФАПЧ с корреляционной коррекцией.

6.8 Оценка канала

Оценка канала опирается на сбор информации об ошибках при прохождении сигнала через среду передачи. Эти эффекты включают рассеяние, затухание и снижение мощности в зависимости от расстояния. Чтобы выполнить оценку канала надо знать его характеристики. Процесс начинается с начальной оценки канала и отслеживания изменений его свойств.

Для оценки канала используются разные методы: слепой, полуслепой и метод обучения.

6.9 Расформирование символов

Остальная часть приёмника аналогична конструкции передатчика. В большинстве случаев расформирование символов определяется демодуляцией при выборе типа модуляции. Однако существуют и другие способы расформирования. Простейший из них называется разделением. Вы выбираете пороговое значение уровня или определённые уровни напряжения и присваиваете каждому из порогов определённые биты или символы. Затем при считывании вы определяете граничные значения двоичных данных передаваемого сигнала.

6.10 Коррекция кода

Подобно свёрточному кодеру, представленному в главе с передатчиком, кодирование Рида-Соломона тоже использует серию регистров. Оно работает подобно свёрточному коду, но вместо считывания из одного полинома, считывает из двух. Коды Рида-Соломона являются недвоичными, циклическими кодами для коррекции ошибок. Они позволяют систематически создавать коды, способные обнаруживать случайные символьные ошибки.

Кроме того, эти блоки могут изменять структуру фрейма путём редактирования Payload_Length и Preamble_Length (рис. 6.27).

PreambleLength	PreambleLength
Preamble_ModType	Preamble_ModType
PayloadLength	PayloadLength
Payload_ModType	Payload_ModType

Рис. 6.27. Изменение длины полезных данных и преамбулы.

Глава 7. Моделирование системного уровня

7.1 Коэффициент битовых ошибок (BER)

Теперь, когда мы научились создавать модели цифровых систем связи, пришло время узнать, насколько точно САПР SystemVue может моделировать и тестировать созданные модели. Начнём с коэффициента битовых ошибок (BER) – безразмерного параметра, показывающего насколько подвержена ошибкам наша система связи.

Пример. DigMod_Tx_Rx_AWGN, папка 4. DigMod_TxRx_NoISI_BER

В САПР SystemVue BER можно рассчитывать двумя способами. Один заключается в использовании уравнений. Для тестирования теоретической модели ошибок на основе уровня частоты (в dBs) **нужно записать несколько уравнений**. Первое общее уравнение BER – уравнение E.7.1, описывает модуляцию QPSK с шумом AWGN.

$$BER = \frac{1}{2} \operatorname{erfc}(\sqrt{E_b/N_0}) \quad \text{E.7.1}$$

Однако в большинстве случаев это уравнение описывает лишь базовую составляющую общего расчёта BER. Поэтому, если кто-то использует другие коммуникационные схемы, ему нужно добавить в это уравнение дополнительные члены или изменить его. При работе в SystemVue об этом можно не беспокоиться. Мы можем рассчитать BER для всей схемы и использовать блоки для построения графика BER. Эти наборы данных создаются с помощью блока BER_FER (коэффициент битовых и фреймовых ошибок), показанного на рис. 7.1.

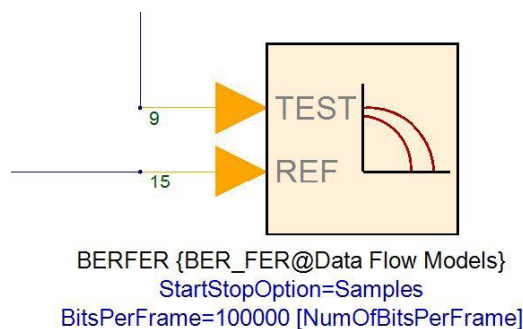


Рис. 7.1. Линия 9 блока BER представляет собой выход приёмника, а линия 15 – вход передатчика.

Следующий пример из “DigMod_TxRx” берёт BER и сравнивает его с теоретическими пределами для данного типа модуляции. Он создаёт систему уравнений в поле страницы уравнений рабочей среды Theoretical_QAM_BER (рис. 7.2).

```

1 EbNodB_1024=-6:2:30
2 EbNo=10.^(EbNodB_1024/10);
3 %1024-QAM BER
4 k=10;
5 M=2^k;
6 x=sqrt(3*k*EbNo/(M-1));
7 QAM_1024_Pb=(4/k)*(1-1/sqrt(M))*(1/2)*erfc(x/sqrt(2));
8
9 EbNodB=-6:2:24
10 EbNo=10.^(EbNodB/10);
11
12 %256-QAM BER
13 k=8;
14 M=2^k;
15 x=sqrt(3*k*EbNo/(M-1));
16 QAM_256_Pb=(4/k)*(1-1/sqrt(M))*(1/2)*erfc(x/sqrt(2));
17
18
19 %64-QAM BER
20 k=6;
21 M=2^k;
22 x=sqrt(3*k*EbNo/(M-1));
23 QAM_64_Pb=(4/k)*(1-1/sqrt(M))*(1/2)*erfc(x/sqrt(2));
24
25 %16-QAM BER
26 k=4;
27 M=2^k;
28 x=sqrt(3*k*EbNo/(M-1));
29 QAM_16_Pb=(4/k)*(1-1/sqrt(M))*(1/2)*erfc(x/sqrt(2));
30
31 %QPSK (4-QAM) BER
32 k=2;
33 M=2^k;
34 x=sqrt(3*k*EbNo/(M-1));
35 QAM_4_Pb=(4/k)*(1-1/sqrt(M))*(1/2)*erfc(x/sqrt(2));
36

```

Рис. 7.2. Страница уравнений Theoretical_QAM_BER, показывающая расчёты теоретического значения BER.

По этим уравнениям можно построить график зависимости BER от E_b/N_0 и понять, насколько близко моделируемая система связи QAM приближается к оптимизированной системе. В самом примере этот график обозначен как QAM_BER_vs_Ebn0.

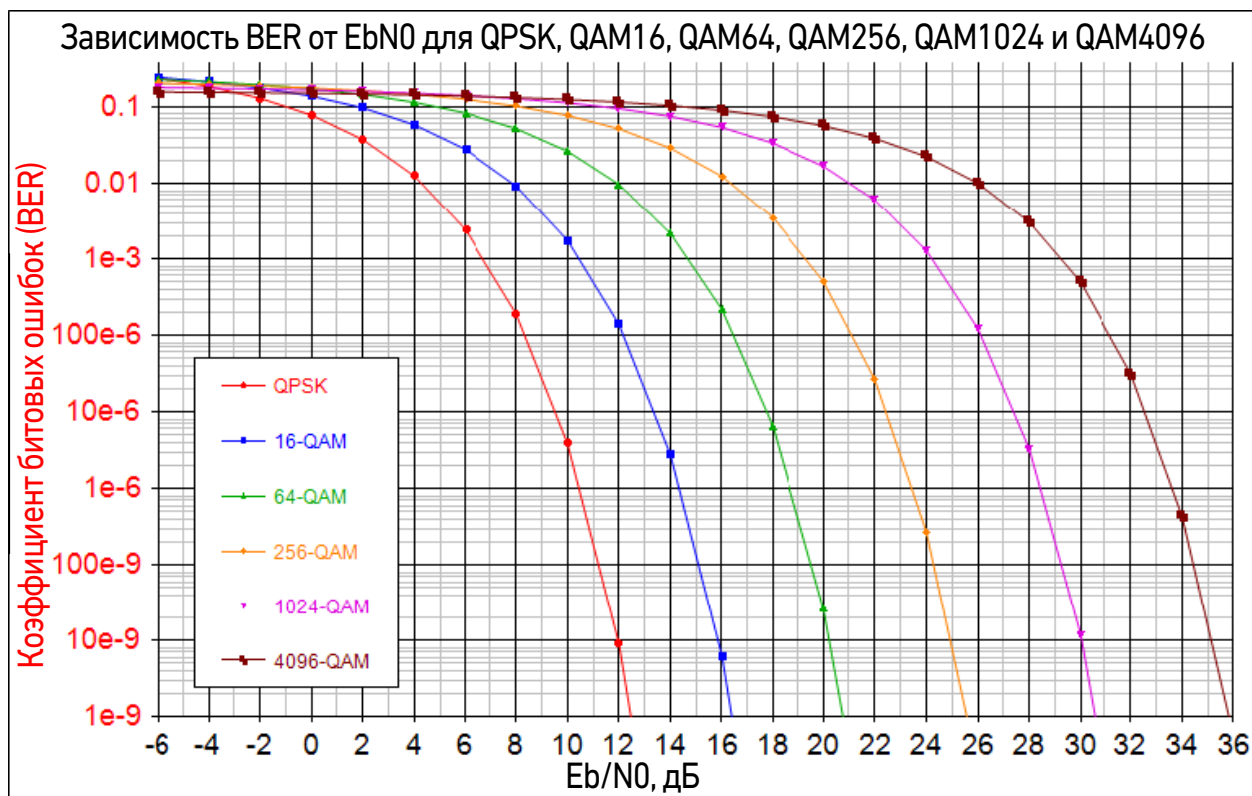


Рис. 7.3. Графики теоретических и измеренных значений BER.

На рис. 7.3 показано сравнение BER, которое демонстрирует очень близкое соответствие теоретических пределов и результатов моделирования. Это говорит о том, что блок приёмника, предлагаемый САПР SystemVue, даёт результаты, близкие к оптимальному приёмнику и, следовательно, может использоваться в качестве эталона. Учитывая, что параметры приёмника уже достаточно близки к теоретическому пределу, мы покажем, что в большинстве случаев не нужно использовать уравнения для измерения теоретических значений BER, и что наш эталонный приёмник будет дублировать результаты этих уравнений.

Примечание. Если вы хотите реализовать анализ BER в примере SystemVue, то это можно сделать следующим образом.

1. Перед построением графика данных создайте переменную, которую можно анализировать. Она называется переменной настройки и позволяет использовать так называемое уравнение свёртывания для измерения BER при разном усилении. Перейдите на страницу схемы и создайте страницу уравнений. На этой странице введите следующий код.

```

EbN0 = ?0; %Переменная настройки
Power_dBm = x; %Мощность передачи в дБм
BitsPerSymbol = x;
SamplesPerChip = x; %Число выборок на символ
Chiprate = x; %Символьная скорость
NDensity = Power_dBm - 10*log10( ChipRate) - 10*log10(SampsPerChip) -
10*log10(BitsPerSymbol) - EbN0;

```

Буквами x обозначены места, куда можно ввести значения той системы, которую вы пытаетесь создать. E_b/N_0 является настраиваемой переменной и имеет знак вопроса перед нулевым значением.

2. Теперь можно создать уравнение свипирования. Для этого перейдите в дерево рабочей среды и откройте меню New Item -> Evaluations -> Add Sweep (Создать объект -> Уравнения -> Добавить свипирование).
3. Откройте страницу Sweep (свипирование). Здесь нужно изменить пару областей. Во-первых, установите Analysis (анализ) в значение Sweep (тип анализа, который вы хотите выполнить), перейдите в Parameter to Sweep (свипируемый параметр) и выберите в качестве этого параметра переменную настройки. Измените начало и конец диапазона на те значения, между которыми вы хотите измерить BER. Выберите режим свипирования (или, другими словами, величину шага). В данном примере мы выбираем Linear: Step Size (Линейное: размер шага) и затем нажимаем на кнопку "Calculate Now" (рассчитать).
4. Перейдите в набор данных, созданный в процессе свипирования. Щёлкните правой кнопкой на BERFER_BER -> Add to Graph (добавить к графику) -> New Graph Series Wizard (мастер нового графика) -> Y vs. X Graph (график зависимости Y от X) -> "Parameters_EbN0_Swp_ERFER_ER_Index" установлен в значение X, а ERFER_BER – в значение Y.

7.2 Моделирование в нескольких областях

Примечание. Для выполнения анализа в нескольких областях вам понадобится библиотека для ВЧ проектирования W1719. В примере, приведённом в этом разделе, используются лицензии на библиотеку для ВЧ проектирования W1719 и библиотеку W1910 для проверки модулирующего сигнала 3GPP LTE.

Для создания любой части системы связи нужны знания в области проектирования схем ВЧ и модулирующего сигнала. Однако эти дисциплины сильно отличаются друг от друга и традиционно считаются разными областями электроники. ВЧ сигналы обычно рассматриваются в частотной области и обрабатываются аналоговыми фильтрами и усилителями. Цифровые схемы ориентированы на временную область, ключевыми элементами которой являются частоты дискретизации и типы модуляции.

Обычно одновременная разработка схем ВЧ и модулирующего сигнала занимает много времени и денег. Создавая прототип приёмника, инженеры часто вынуждены разрабатывать изделие «с нуля». Обычно на это уходят месяцы, с учётом того, что технические характеристики системы могут в любой момент измениться. Эти изменения заставляют инженеров тратить время и деньги на создание новых версий исходной конструкции. А поскольку система связи состоит из ВЧ и цифровых узлов, даже небольшое изменение любого узла может отразиться на работе других узлов. Во многих САПР (в том числе и в SystemVue) можно использовать уравнения Mathlang/MATLAB для моделирования поведения устройств, но часто эти возможности ограничены и не позволяют корректно работать с ВЧ компонентами.

Вот здесь-то и выходит на сцену САПР SystemVue. Она хорошо работает в роли платформы проектирования верхнего уровня, благодаря её способности объединять область ВЧ с областью цифровых сигналов в процессе, который называется моделированием в нескольких областях. И хотя SystemVue изначально ориентирована на работу с модулирующими сигналами, она может моделировать ВЧ схемы и создавать всеобъемлющие модели, способные имитировать работу системы в аналоговой и цифровой областях.

Возможность анализировать результаты в цифровой и аналоговой областях позволяет инженерам начинать создание прототипа системы связи, не беспокоясь о построении схем с самого начала. Обладая подобным пакетом, вы можете легко сократить время и стоимость разработки. Например, при изменении спецификаций цифровой части системы, вы можете быстро изменить спецификации ВЧ узлов, не разрабатывая заново печатную плату. Кроме того, это означает, что вы получаете точные модели, которые можно показать заказчикам, поставщикам или конкурентам. Моделирование в нескольких областях мы рассмотрим в следующем примере.

Пример. 3GPP_LTE_DL_SISO_BER_RF_Link

Чтобы открыть этот пример, выполните следующие шаги.

1. Перейдите в Open Example (открыть пример).
2. Откройте папку RF Architecture Design (проектирование ВЧ архитектуры).

Теперь откройте схему "LTE_FDD_DL_AWGN_BER". Она должна выглядеть, как на рис. 7.4.

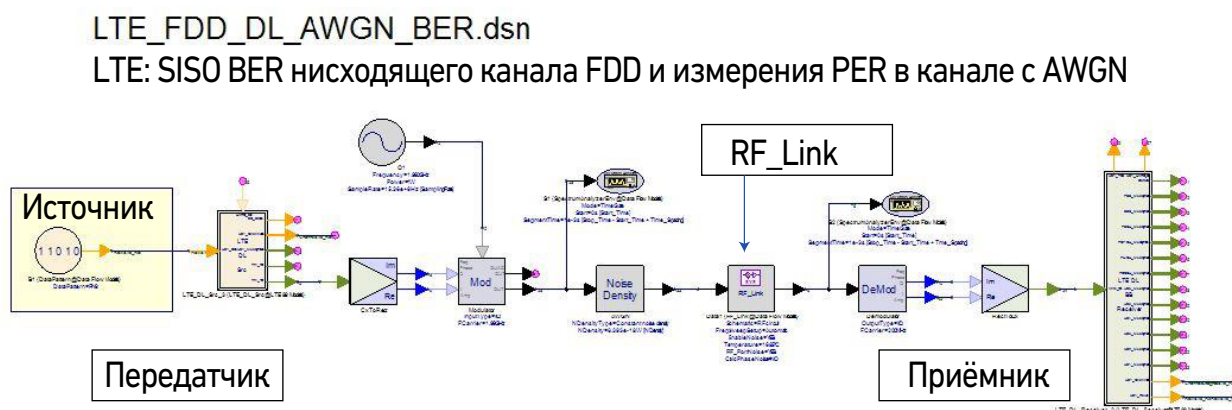


Рис. 7.4. Схема LTE_FDD_DL_AWGN_BER.

Схема на рис. 7.4 подобна ранее описанным передатчикам и приёмникам. Блоки LTE (блоки с числом выходов больше двух) можно считать блоками формирования/расформирования символов и канального кодирования/декодирования соответственно. Всё остальное выглядит относительно стандартно. Узлы передатчика и приёмника мы отделили на схеме двумя вертикальными линиями. Между ними располагается блок плотности шума, который имитирует среду передачи сигнала. Интересующий нас блок называется RF_Link (самый первый блок приёмника).

ВЧ интерфейс приёмника нисходящего канала LTE (в Spectrasys)

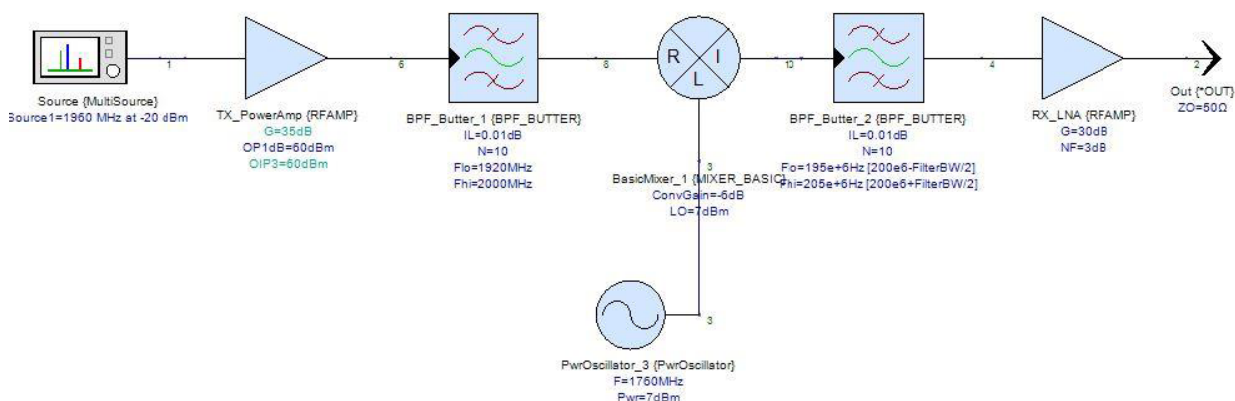


Рис. 7.5. Пример проекта Spectrasys. В примере 3GPP_LTE_DL_SISO_BER_RF_Link эта схема обозначена, как блок RF_Link.

Начнём мы с создания проекта Spectrasys (рис. 7.5). Проект Spectrasys представляет собой схему с параметризованными ВЧ блоками. Модуль RF_Link ссылается на подсхему. Примером такой подсхемы является схема с именем Rfcircuit, которую можно найти в папке "RFBLOCKS". После создания схемы блок RF_Link можно настроить так, чтобы в области свойств он ссылался на проект Spectrasys. В зависимости от версии SystemVue вам может понадобиться указать, какой порт является входом, а какой порт – выходом. На этом настройка ВЧ системы закончена.

Описанный процесс позволяет настроить анализ ВЧ системы, реализующий её поведение. Здесь вы можете привязать анализ к проекту Spectrasys, что позволяет создать набор данных для ВЧ системы в частотной области.

И, наконец, блок RF_Link, расположенный на главной схеме высокого уровня, можно проанализировать в обычном симуляторе Data Flow. Это позволяет моделировать схемы нижнего уровня в поведенческих моделях верхнего уровня. Благодаря такой возможности вы можете анализировать топологию устройства и интегрировать другие устройства, не создавая ВЧ схему с самого начала.

7.3 Генерация кода HDL

Примечание. Для работы с программируемыми логическими интегральными схемами (ПЛИС) вам понадобится библиотека для проектирования оборудования W1717.

Создавая всеобъемлющую систему автоматизированного проектирования, было бы опрометчиво не включить в неё поддержку языка описания аппаратного обеспечения (HDL) для ПЛИС. В комплект поставки SystemVue входит библиотека для проектирования оборудования W1717, позволяющая создавать код HDL для моделирования прототипов схем. Но прежде чем мы сможем создать файл HDL, нам придётся преобразовать высокоуровневые модели системы в более дружественный для сигнальных процессоров формат. Чтобы понять, зачем это надо, нужно пояснить разницу между числами с фиксированной и плавающей запятой.

Эти два формата отличаются способом двоичного представления численных значений. Обычно формат с фиксированной запятой является традиционным, он представляет точное значение числа. Формат с плавающей запятой работает по-другому. Большая часть битов двоичного слова используется для обозначения целого значения или основания числа, тогда как другие биты обозначают степень. Это позволяет расширить диапазон чисел и обеспечивает более высокую точность выражения любого числа.

Но хотя формат с плавающей запятой более эффективен, он имеет некоторые ограничения, которые не позволяют использовать его в качестве основного формата для сигнальных процессоров или ПЛИС. Его способность передавать большой диапазон численных значений повышает сложность описания схем. Большой размер чисел требует большего объёма памяти, больше потребляемой мощности и физического размера. Оборудование, рассчитанное на работу с числами с фиксированной запятой, значительно проще и потребляет меньшую мощность. Поскольку схемы становятся всё сложнее и потребляют всё больше мощности, стоимость изготовления аппаратного обеспечения, поддерживающего числа с плавающей запятой, растёт. Все эти факторы делают оборудование для чисел с фиксированной запятой значительно более приемлемым для мобильных устройств, которые должны быть небольшими, лёгкими и сравнительно дешёвыми.

Так почему же для моделирования системы связи мы решили использовать инструменты, работающие с плавающей запятой? Проверка формата с фиксированной запятой печально известна своей сложностью. Проектирование в формате с фиксированной запятой более подходит для работы на нижнем уровне. Это значит, что большая часть телекоммуникационного оборудования разрабатывается на верхнем уровне с применением формата с плавающей запятой, но на нижнем уровне создаётся с применением формата с фиксированной запятой. А поскольку эти форматы используют в корне отличающиеся представления чисел, они используют разные алгоритмы и, следовательно, генерация кода HDL для схем с фиксированной запятой требует применения специальной программы HDL.

Пример. QPSK_transceiver_with_symbol_timing_recovery

Чтобы открыть этот пример, выполните следующие действия.

1. Перейдите в Open Example (открыть пример).
2. Откройте папку Hardware Design (проектирование оборудования).
3. Откройте папку CodeGeneration (генерация кода).
4. Откройте папку QPSK_transceiver_with_symbol_timing_recovery (приёмник QPSK с восстановлением синхронизации символов).

Работа с языком HDL в САПР SystemVu заключается в простом обращении к библиотеке проектирования оборудования. Эта библиотека содержит копии множества алгоритмических блоков, использующих формат с фиксированной запятой. Они обозначены розовым цветом и в их имени присутствует обозначение "Fxp". Чтобы начать проектирование, просто создайте дополнительную модель схемы или подсхемы и начните размещать блоки Fxp. SystemVu сильно упрощает процесс генерации HDL и ускоряет кодирование, преобразуя высокоуровневые функции и схемы в оптимизированный и эффективный код HDL. Это, в свою очередь, сокращает время разработки за счёт применения графического интерфейса.

Пример модели с фиксированной запятой приведён на рис. 7.6.

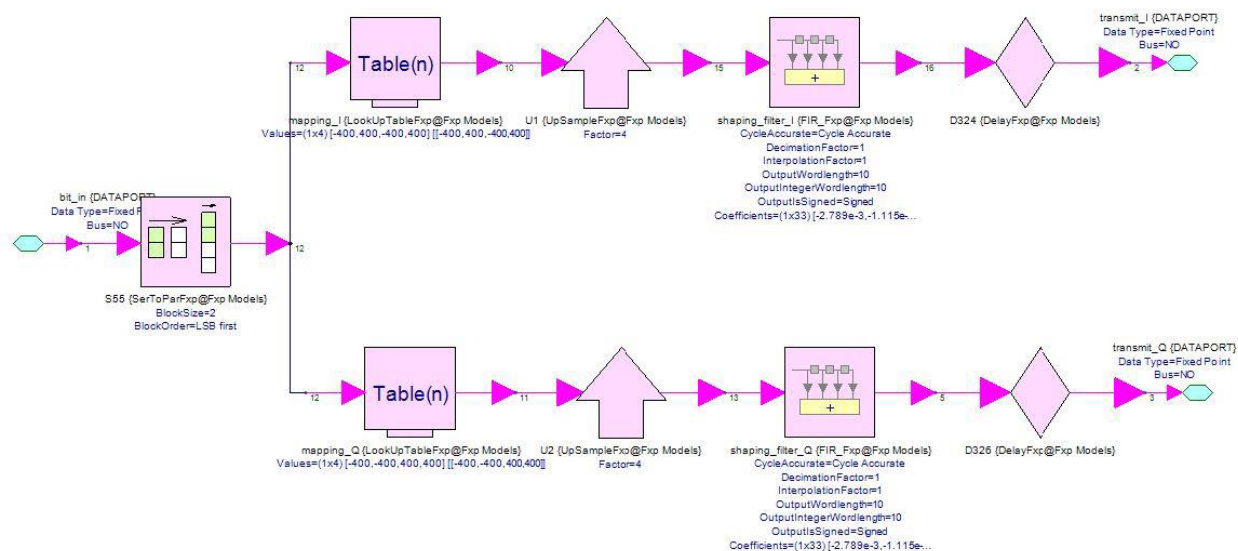


Рис. 7.6. Подсхема qpsk_mod в папке Tx из примера QPSK_transceiver_with_symbol_timing_recovery.

Обратите внимание, что разные блоки на рис. 7.6 имеют эквивалентные алгоритмы, упрощающие переход от разработки алгоритма к реализации HDL.

Для создания кода HDL проделайте следующее.

1. Перейдите в дерево рабочей среды, а затем в Code Generators (генераторы кода) (расположенные в меню создания нового документа).
2. Добавьте HDL Code Generator (генератор кода HDL). Откроется окно, подобное тому, что показано на рис. 7.7.

HDL Code Generation Options

Name: QPSK_transceiver_ML402

Top Level Design: Design1

Description: Implementation on Xilinx ML402 FPGA board

Configuration: Default ...

Factory Defaults

Select Subnetwork for HDL Generation

Select Items:

Subnetwork	Model	Generated Module Name
Data1	demod_awgn	demod_awgn

Output Directory

☒ Use Default Directory

C:\Program Files\SystemVue2013.08\Examples\Hardware

Target Configuration

☒ Automatically add generated model to Part model list

Target: Xilinx FPGA

Test Vector Generation: OFF

Language: VHDL

Subnetwork: Data1

Subnetwork Configuration

Project Setup | Clock/Reset | I/O Settings

Generation Mode: ISE Project

Synthesis Tool: XST

FPGA Family: Virtex4

FPGA Device: XC4V5X35

FPGA Package: FF668

FPGA Speed: -10

Generate Global Options... OK Cancel Help

Рис. 7.7. Окно генератора кода HDL.

На рис. 7.7. показано, как преобразовать подсистемы и схемы верхнего уровня в HDL. Код может создаваться в формате VHDL или Verilog для любого числа ПЛИС. Кроме того, SystemVue предлагает способ настройки тактовой частоты и сброса для HDL и создаёт проект ISE или непосредственно генерирует двоичный файл.

Глава 8. OFDM

8.1 Введение

Теперь, когда мы познакомились с основами цифровой связи, пришло время взглянуть на другие варианты реализации коммуникационных систем. При проектировании любых цифровых систем связи приходится решать одни и те же проблемы. Одна из этих проблем заключается в стремлении повысить скорость передачи данных, но при этом возникает ограничение, связанное с ростом BER. При передаче информации по каналу основной проблемой являются искажения сигнала. В свою очередь, различные системы цифровой связи создаются с учётом таких факторов, как многолучевое распространение и ВЧ помехи. Помимо требований к достоверной передаче сигнала, существует ещё и потребность в создании более быстродействующих схем обработки. Система цифровой связи, которая пытается решить все эти проблемы, называется OFDM.

Мультиплексирование с ортогональным делением частот (или OFDM) предусматривает передачу данных в нескольких соседних частотных поддиапазонах, что иногда называется методом модуляции с несколькими несущими. Один поток данных разбивается на несколько параллельных сигналов, как показано на рис. 8.1. Затем эти сигналы передаются ортогонально, то есть так, чтобы их частотные характеристики имели минимальное перекрытие в точках пиковых значений и, следовательно, не влияли друг на друга. Каждый сигнал имеет свою схему модуляции, выбранную в зависимости от отношения сигнала к шуму (С/Ш) на этой конкретной частоте. Если отношение С/Ш мало, то такая частота не используется. Если отношение С/Ш велико, то используется сложная схема модуляции. При среднем значении отношения С/Ш используется простая схема модуляции. Благодаря этому OFDM обеспечивает высокую достоверность передачи, не прибегая к помощи традиционных корректирующих фильтров.

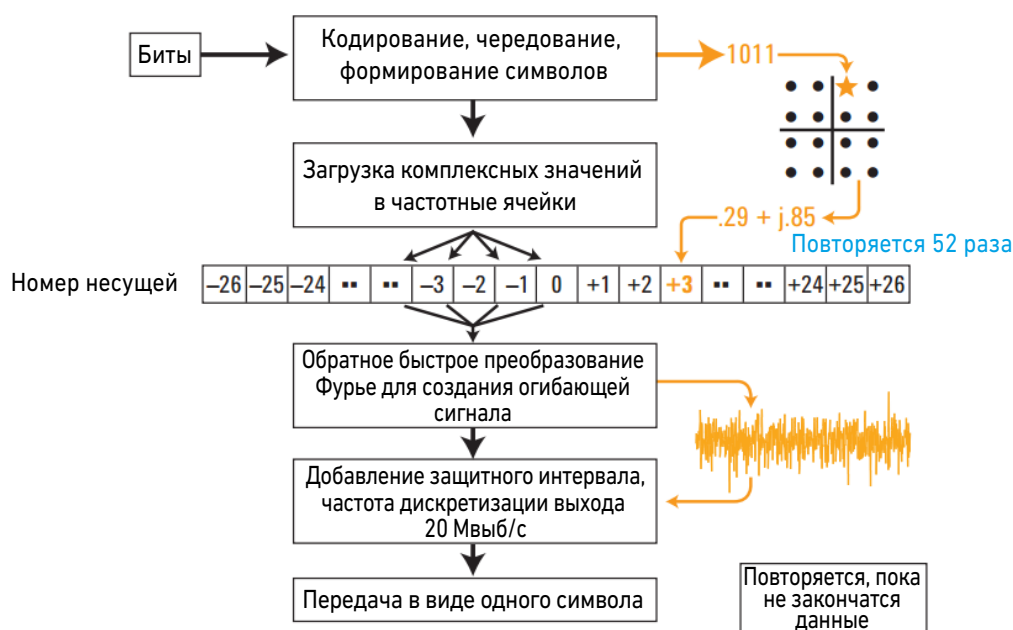


Рис. 8.1. Процесс передачи OFDM.

OFDM обладает целым рядом преимуществ. Её конфигурация повышает эффективность использования спектра, позволяя упаковывать информацию в более плотный частотный спектр. Другое преимущество заключается в том, что благодаря применению нескольких сигналов, она может передавать данные в несколько раз быстрее, чем обычные цифровые коммуникационные схемы с одним входом и одним выходом. Кроме того, она структурирована так, что влияние многолучевого распространения и узкополосных помех существенно ослабляется, поскольку затухание на некоторых частотах приводит к тому, что эти участки спектра просто не используются.

Поскольку OFDM хорошо работает в неблагоприятных условиях распространения радиосигналов, она используется в широком диапазоне коммуникационных протоколов, например в WiMAX™ и 4G LTE. Также OFDM широко применяется в системах Ethernet и оптических линиях связи. В оптических системах связи

OFDM позволяет преодолеть множество ограничений, таких как модовая дисперсия и фазовая автомодуляция⁵⁰. Кроме того, эта схема модуляции используется в спутниковой связи, а также в авиационных, GPS и военных приложениях.

Но это не значит, что OFDM полностью безупречна. Поскольку система использует несколько выходных сигналов на разных частотах, она страдает от большого отношения пиковой мощности к средней (PAR). Также OFDM испытывает проблемы в случае неточной синхронизации самой системы, которая приводит к потере несущими ортогональности. Кроме того, для сохранения низкого уровня собственных шумов внутри и за пределами диапазона приходится снижать порядок модуляции поднесущих.

Пример. WNW_OFDM_TxRx

Чтобы открыть WNW_OFDM_TxRx, сделайте следующее:

1. Перейдите в Open Example (открыть пример).
2. Откройте папку Comms (связь).
3. Откройте расположенную здесь папку OFDM.

Сначала давайте рассмотрим схему WNW_OFDM_TxRx, показанную на рис. 8.2.

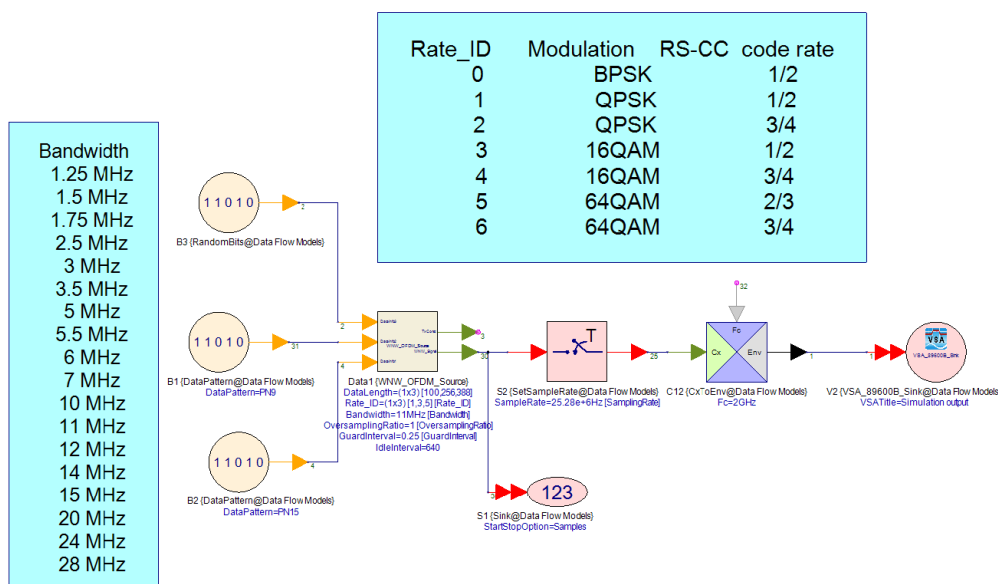


Рис. 8.2. Схема WNW_OFDM_TxRx.

На этой схеме показано высокоуровневое представление передатчика OFDM, причём синие блоки показывают доступные опции для переменных Bandwidth (полоса) и Rate_ID (идентификатор скорости). Эти переменные можно изменить на вкладке Parameters (параметры). Блоки битов на рис. 8.3, представляющие три разных входа данных (например, преамбула, интервал ожидания, полезные данные и обучение), поступают на блок передатчика OFDM. После передатчика OFDM выходной сигнал дискретизируется и направляется в SxToEnv – блок модулятора, который преобразует комплексные сигналы в значения I/Q. Затем эти значения поступают в блок VSA для анализа.

Являясь высокоуровневым представлением, эта схема не показывает подсхемы, составляющие основу OFDM. Поэтому в этой главе мы будем ссылаться также и на модель WNW_OFDM_Source.

8.2 Структура фрейма OFDM

Общая структура фрейма для большинства примеров OFDM показана на рис. 8.4.



Рис. 8.4. Структура данных OFDM.

Как правило, структура фрейма начинается с интервала ожидания, преамбулы и полезных данных. Чаще всего OFDM начинается с нескольких преамбул и полезных данных, как показано на рис. 8.4. Полезные данные и преамбула могут состоять из нескольких секций. В нашем примере WNW_OFDM_TxRx мы имеем одну секцию полезных данных, состоящую из трёх частей, и одну преамбулу. Преамбулы используются в схемах передачи данных, имеющих пакетный (прерывистый) характер. В зависимости от характеристик системы, преамбулы 1 и 2 можно определить в частотной или во временной области. Кроме передачи информации о синхронизации и частоте, преамбулы используются для реализации таких функций, как автоматическая регулировка усиления (АРУ), оценка канала и начальная оценка опорной фазы. Часто преамбулы снабжаются циклическими префиксами и суффиксами на случай возникновения проблем (рис. 8.5). Префиксы и суффиксы помогают устранять межсимвольные помехи, выступая в роли защитных интервалов и позволяя выполнять простую обработку в частотной области, такую как оценка и коррекция канала.

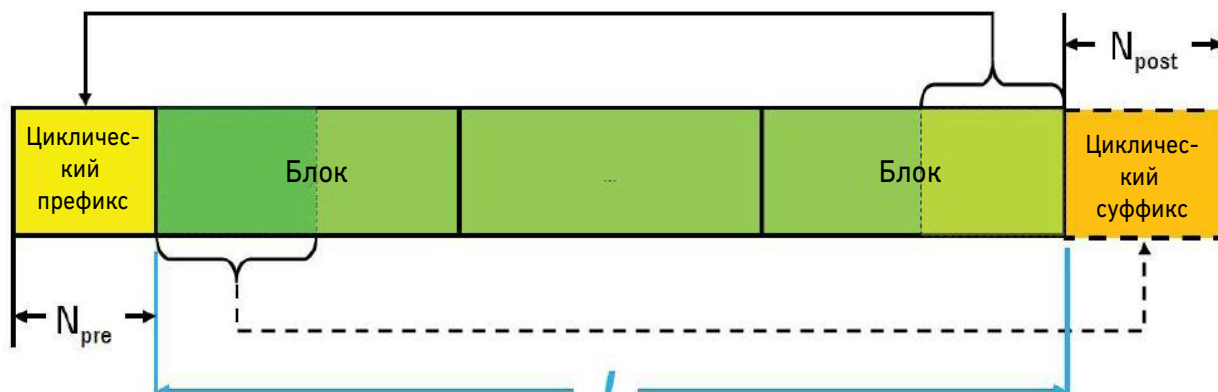


Рис. 8.5. Пример структуры преамбулы системы OFDM.

8.3 Полезные данные OFDM

Сигнал полезных данных OFDM генерируется путём применения обратного быстрого преобразования Фурье (обратное БПФ) к комплексным значениям сигнала. Полезные данные OFDM могут использовать разные форматы модуляции для разных поднесущих.

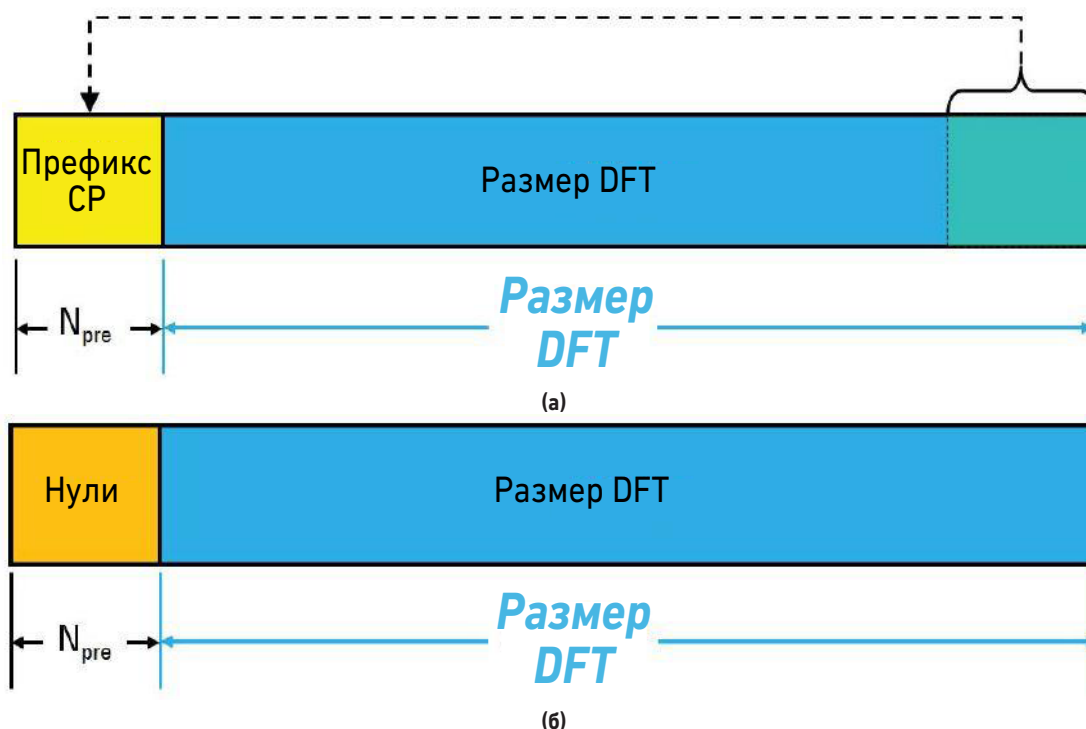


Рис. 8.6. Пример полезных данных OFDM с циклическим префиксом (а). Пример символа полезных данных OFDM с префиксом, состоящим из нулей (б).

Символ OFDM создаётся путём добавления циклического префикса к началу каждого создаваемого блока. На рис. 8.6 показан пример двух префиксов. На рис. 8.6а показан циклический префикс, который используется для того, чтобы групповая задержка в канале не порождала последующих символов OFDM или помех на соседних поднесущих. На рис. 8.6б показаны полезные данные с защитным интервалом из нулей.

8.4 Пилот-сигналы

Обычно пилот-сигнал передаётся на одной частоте и используется для синхронизации, обеспечения целостности, контроля или в качестве опорного сигнала¹¹. В непрерывных системах OFDM пилот-сигналы используются для синхронизации и оценки канала.

В текущих стандартах OFDM используются четыре основные пилотные последовательности (рис. 8.7).

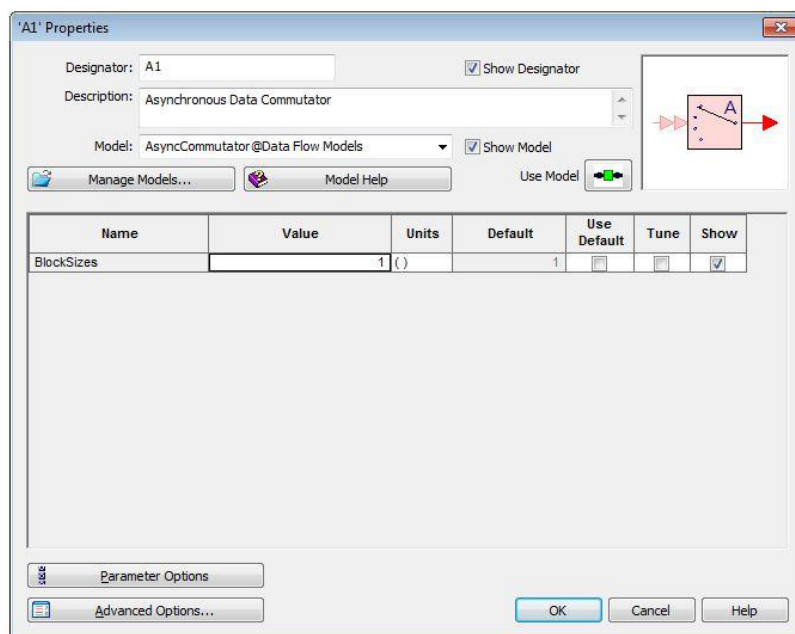
Система OFDM без пилот-сигнала (рис. 8.7а). В некоторых кабельных системах OFDM (например, в системе ERDF G3-PLC) пилот-сигналы не используются из-за применения предварительного корректора.

Система OFDM с непрерывными пилот-сигналами (рис. 8.7б). Например, стандарты беспроводной связи WLAN (802.11a/g/n) используют только непрерывные пилот-сигналы (индексы пилотной поднесущей фиксированы во всех символах OFDM). Такие непрерывные пилот-сигналы используются для отслеживания фазы на рис. 8.4б, поскольку такие системы OFDM используют преамбулу для оценки канала.

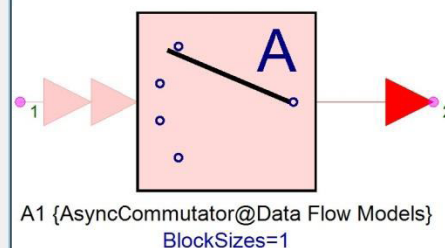
Система OFDM с распределёнными пилот-сигналами (рис. 8.7в). Примером таких систем является Mobile WiMAX™ и стандарты сотовой связи 3GPP LTE (то есть, индексы пилотной поднесущей меняются в каждом символе OFDM). В этом случае распределённые пилот-сигналы используются для оценки канала, поскольку система OFDM использует преамбулы или каналы синхронизации для синхронизации времени и частоты.



отличается от многих других коммуникационных форматов тем, что распределяет передаваемые данные в частотной области по поднесущим, а затем преобразует их с помощью обратного быстрого преобразования Фурье во временную область для передачи в качестве модулирующего сигнала. В SystemVue это делается очень просто.



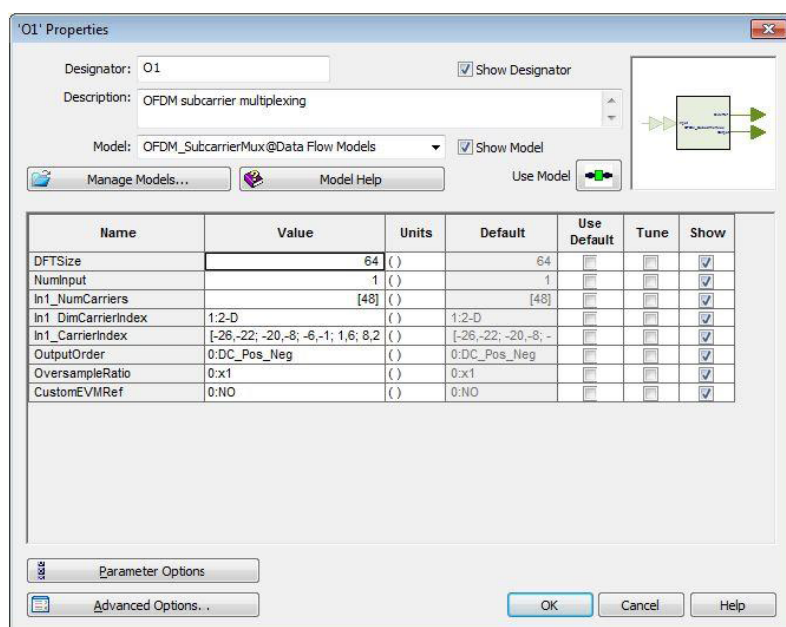
(a)



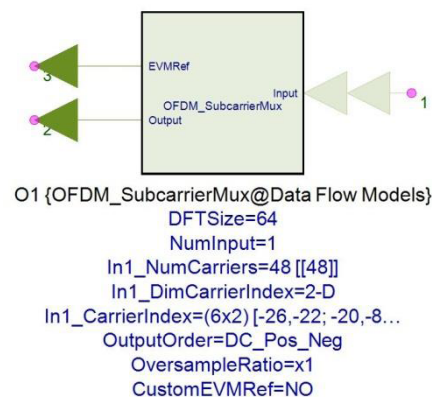
(б)

Рис. 8.8. Окно свойств AsyncCommutator (а). Блок AsyncCommutator (б).

Начнём с выходов формирователя символов, который представляет сигналы и распределяет их по частотному спектру. В данном случае они подаются на блок AsyncCommutator (асинхронный коммутатор). Этот блок мультиплексирует параллельные сигналы и последовательно передаёт их на один выход (рис. 8.8).



(a)



(б)

Рис. 8.9. Окно свойств OFDM_SubcarrierMux (а). Блок OFDM_SubcarrierMux (б).

Затем последовательные данные направляются в специальный блок SystemVue – OFDM_SubcarrierMux, который автоматически преобразует символьную информацию в формат частотной области OFDM (рис. 8.9). Таким образом информация распределяется по ортогональным частотам в частотной области. Перед запуском убедитесь, что переменная NumInput соответствует числу входов.

На рис. 8.10 показано, как расположены поднесущие. Первичные частоты представлены высокими параболой, а расположенные ниже пики представляют остальную часть спектра. Этот спектр пересекает нуль в точках $\pm 1/T$, $\pm 2/T$, $\pm 3/T$ и т.п.

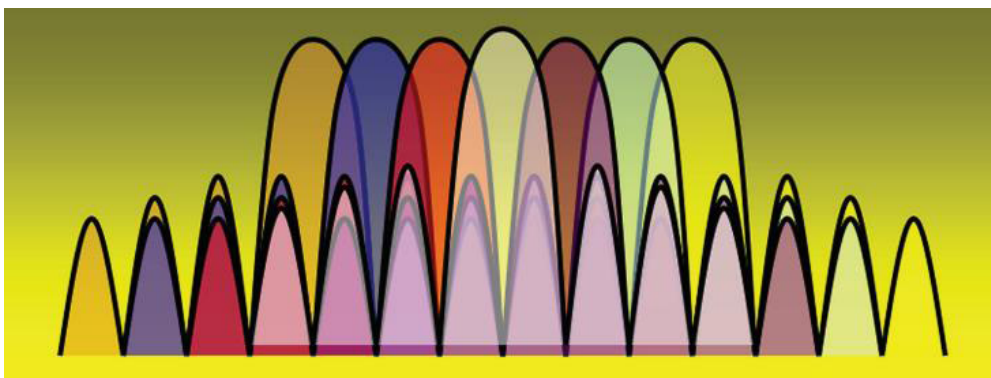
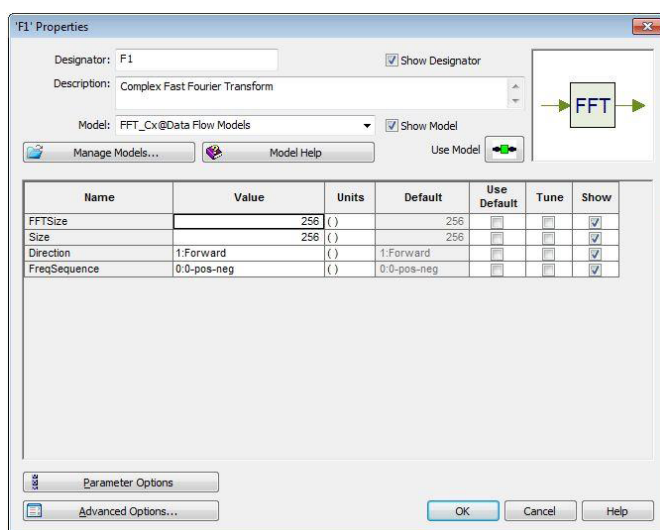
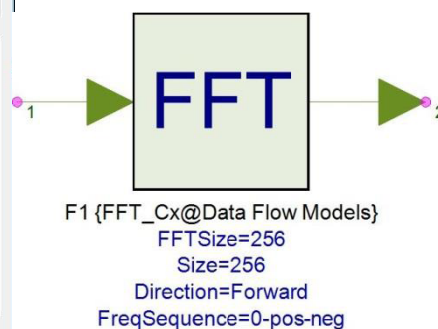


Рис. 5.10. Спектр ортогональных сигналов.

Следовательно, если сигналы разнесены на $1/T$ Гц, то они не мешают друг другу и являются ортогональными.



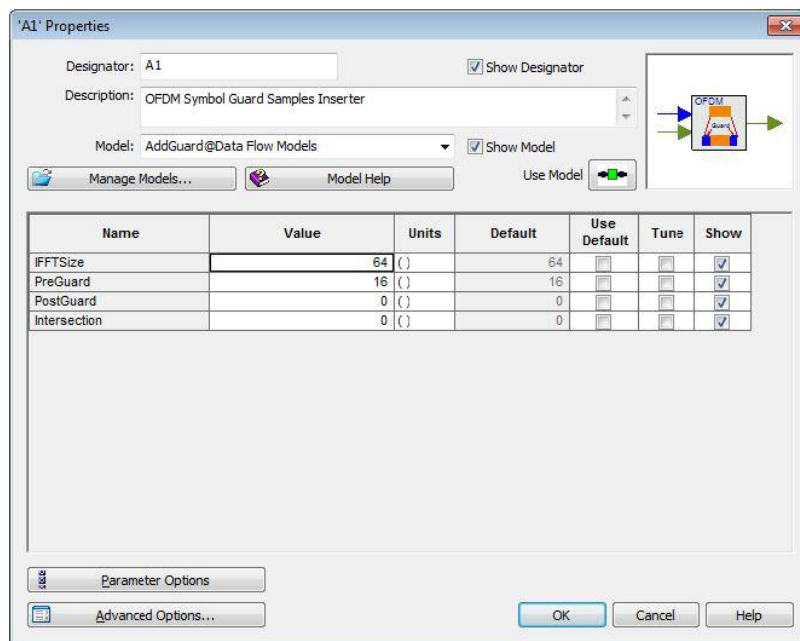
(a)



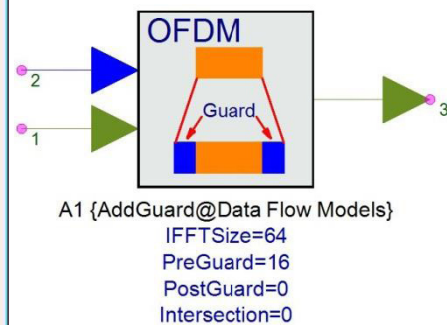
(б)

Рис. 8.11. Окно свойств FFT_Cx (а). Блок FFT_Cx (б).

Когда информация закодирована в такой форме, она готова к выполнению обратного БПФ. Обратное БПФ преобразует данные из частотной области во временную, позволяя подать на выход модулирующий сигнал. Чтобы сделать это в SystemVue, возьмите блок FFT_CX и замените в нём "Forward" (прямое) на "Inverse" (обратное), как показано на рис. 8.11.



(a)



(б)

Рис. 8.12. Окно свойств AddGuard (а). Блок AddGuard (б).

После создания сигнала добавьте блок AddGuard, чтобы добавить с обеих сторон защитные интервалы (рис. 8.12). Значение защитного интервала определяется верхним входом. Конечный результат показан на рис. 8.13.

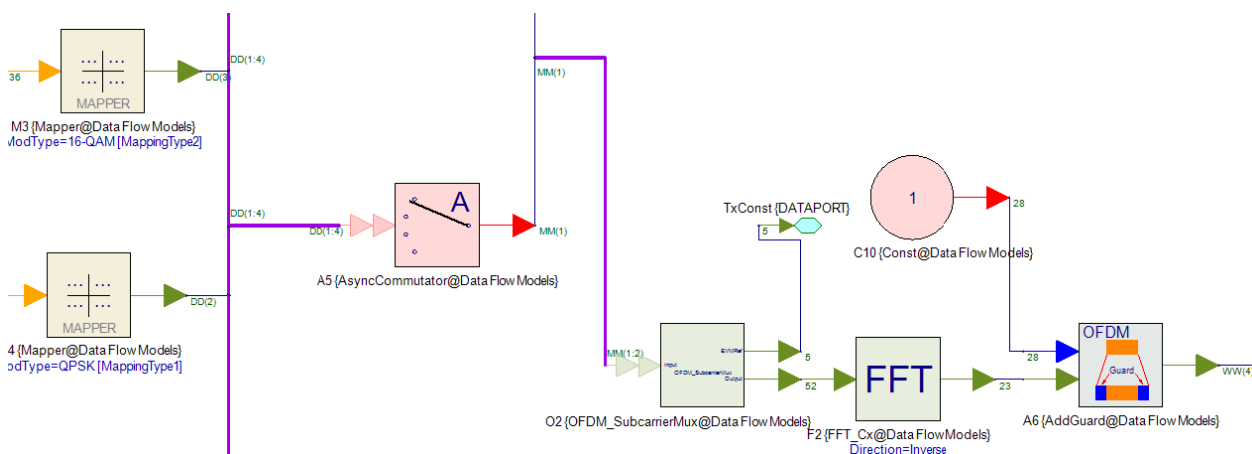


Рис. 8.13. Небольшой фрагмент модели WNW_OFDM_Source, показывающий мультиплексирование поднесущих, обратное БПФ и защитные интервалы.

Глава 9. Основы распределённого спектра CDMA

9.1 Введение

Системы множественного доступа с кодовым разделением каналов (CDMA) позволяют пользователям одновременно передавать данные в одном и том же частотном диапазоне. В этом методе передачи частотный спектр сигнала данных распределяется с применением специальной схемы кодирования, в которой каждому пользователю присваивается свой код. В этой главе мы рассмотрим процесс создания простой системы с распределённым спектром (CDMA) и покажем все инструменты, необходимые для моделирования, отображения и анализа наиболее важных аспектов системы с помощью SystemVue.

CDMA представляет собой форму системы связи с прямым распределением спектра, в которой передаваемые данные кодируются с очень высокой частотой. Для передачи данных в системах с распределённым спектром используется полоса сигнала, значительно превышающая полосу, необходимую для поддержки многопользовательского доступа. Кроме того, большая полоса предотвращает возникновение помех от других пользователей. Многопользовательский доступ достигается за счёт использования псевдослучайного кода. В этой главе псевдослучайный код создаётся с помощью специального генератора, так чтобы обеспечить разделение разных пользователей. Код выглядит случайным, но на самом деле он известен, что позволяет приёмнику его реконструировать. В этой главе псевдослучайный код создаётся генератором кода Уолша из набора ортогональных кодов.

Во время передачи сигнала сигнал данных модулируется псевдослучайным кодом, а результирующий сигнал модулирует несущую. Затем несущая усиливается и передаётся в эфир.

В точке приёма несущая принимается и усиливается. Затем из кодированного сигнала извлекается полезная информация путём сопоставления принятого кода с кодом, генерируемым в приёмнике (тем же кодом Уолша, что и в передатчике). Таким образом приёмник может реконструировать код и извлечь переданные данные.

В этой главе описана лабораторная работа, состоящая из восьми этапов, в результате которой будет создана законченная система цифровой модуляции. Используя BER для оценки характеристик системы с распределённым спектром CDMA, мы можем наблюдать поток битов через предпочтительный канал связи.

9.2 Лабораторная работа 1. Генерируемая последовательность битов

В этой лабораторной работе мы создадим очень простую последовательность битов и проанализируем её выход в форме глазковой диаграммы.

1. Запустите SystemVue (в этом учебном пособии используется версия 2011.10), дважды щёлкнув на соответствующем значке рабочего стола.
2. После загрузки откроется страница приветствия с учебными видеороликами. Пропустите её, щёлкнув кнопку «Закрыть».
3. Следующее окно позволяет открыть имеющиеся рабочие среды или шаблоны. Щёлкнув на «от-меню», создайте пустую рабочую среду.

Откроется окно схемы, в котором и будет выполняться работа по проектированию (рис. 9.1). В верхней части экрана расположена панель инструментов; с левой стороны вы увидите структуру директорий, которая представляет ваше «дерево рабочей среды». Дерево рабочей среды позволяет организовать схемы, представления, уравнения и т.п. в виде иерархической структуры папок. Под деревом рабочей среды расположено «окно настройки», которое является одной из самых мощных функций SystemVue, позволяющих использовать настраиваемые переменные (настройка значений переменных в режиме реального времени) в любом месте проекта. Справа расположена библиотека компонентов (эта часть называется селектором компонентов). В нижней части экрана отображается информация об ошибках. Кроме того, обратите внимание на вкладки под экраном ошибок, показывающие прочую информацию о событиях.

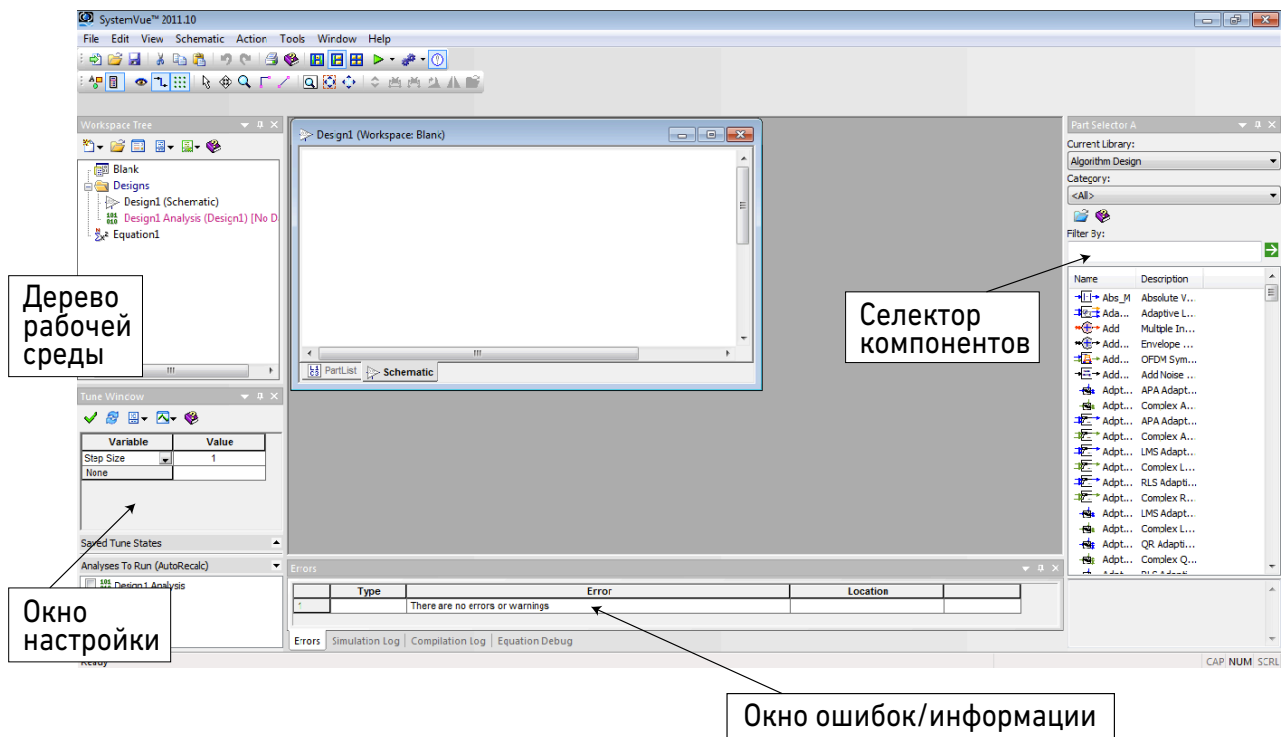


Рис. 9.1. Окно схемы SystemVue.

Итак, начнём.

Откройте меню **File > Save As > My_CDMA_Sys** (Файл > Сохранить как > **My_CDMA_Sys**). В результате вся рабочая среда сохранится под именем **My_CDMA_Sys**. Рекомендуется сохранять рабочую среду после каждой лабораторной работы, выполняя команду **File > Save**.

По умолчанию SystemVue создаёт папку и сохраняет в ней проект и анализ. Переименуйте папку “**Designs**” (Проекты), наведя на неё указатель мыши. Щелкните правой кнопкой и выберите **Rename** (переименовать). Назовите папку “**Step1_Bits**”. Переименуйте схему **Design** в “**Bits**”, а **Design Analysis** в “**DF_Bits**”. Это анализ процесса проектирования, который собирает информацию о наборе значений и запускает нужную программу – примерно так, как это делает компьютер.

Дважды щёлкните на “DF_Bits” и введите значения, показанные на рис. 9.2.

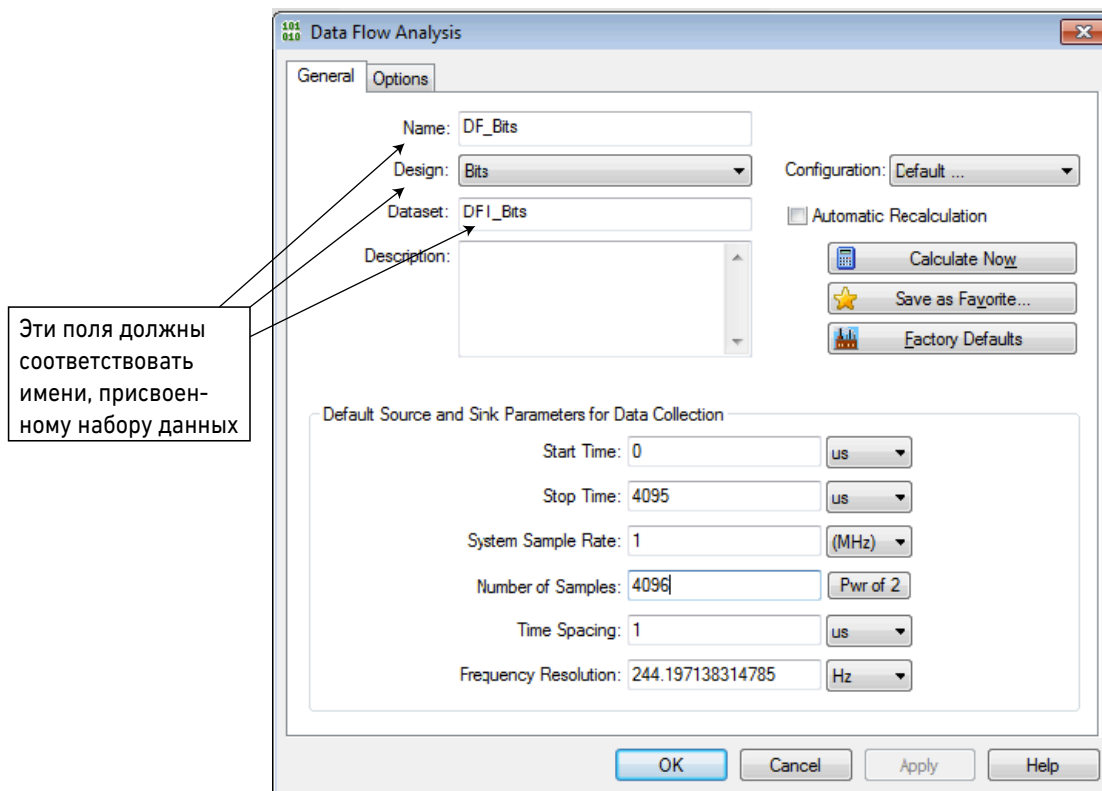


Рис. 9.2. Ввод значений в окно анализа процесса проектирования.

Ключевым параметром является System Sample Rate (системная частота дискретизации). Другим важным параметром является Number of Samples (число выборок), который определяет длину моделирования. Время запуска и останова устанавливается автоматически. Разрешение по частоте рассчитывается по времени останова.

1. Щёлкнув OK, закройте окно.
2. Разверните рабочую среду на весь экран.
3. Из “Part Selector A” (селектор компонента A) введите “Bits” в поле “filter by” (фильтрация по) и нажмите Enter. Вы увидите все компоненты, содержащие слово Bits. Однократно щёлкните на “Bits” и перетащите компонент в пустую схему. Щёлкните снова и отпустите компонент.
4. Дважды щёлкните на компоненте “Bits”. Одна из замечательных и полезных особенностей SystemVue заключается в том, что многие компоненты являются полиморфными моделями.

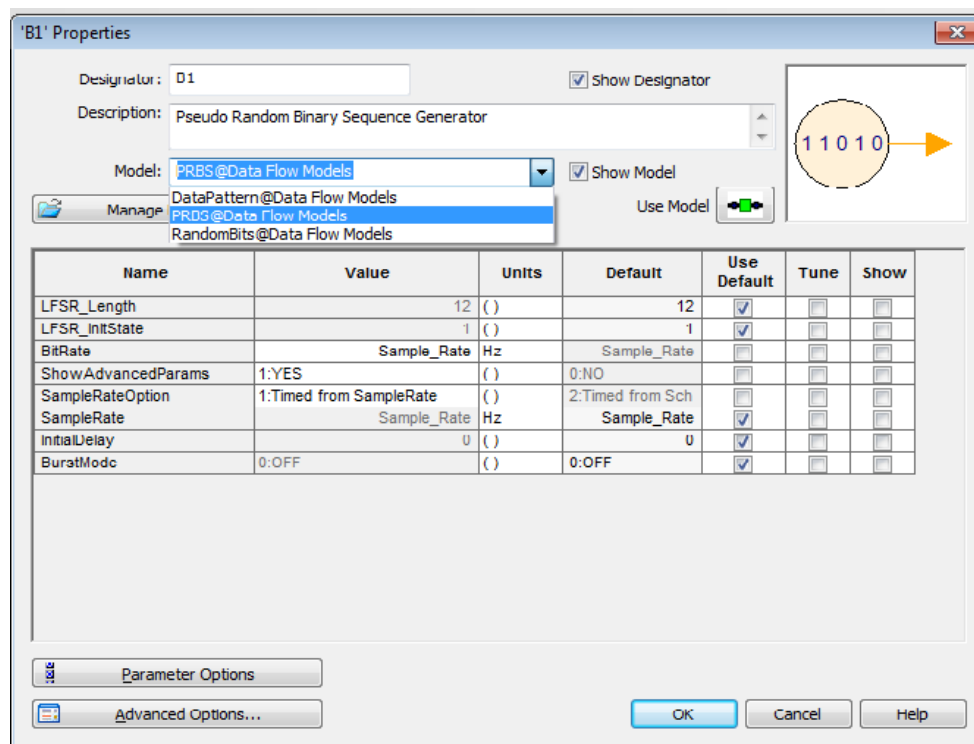


Рис. 9.3. Выпадающее меню свойств B1.

- Щелкните на выпадающем меню возле поля "model" (модель) и выберите "PRBS@Data Flow Models". Эта операция создаст псевдослучайную последовательность.
- Заполните поля, как показано на рис. 9.3. Обратите внимание, что вы можете добавлять в этот список собственные модели, что характеризует SystemVue как гибко настраиваемую систему.

Обратите внимание, что "BitRate" (битовая скорость) установлена в значение "Sample Rate" (частота дискретизации). Благодаря таким настройкам мы получаем одну выборку на бит.

Теперь давайте подключим приёмник к источнику битов и отобразим нашу битовую последовательность. Введите "Sink" (приёмник) в Part Selector. Щёлкните на "Data Sink" (приёмник данных) и установите его рядом с источником битов. Сформируется соединение, как показано на рис. 9.4.

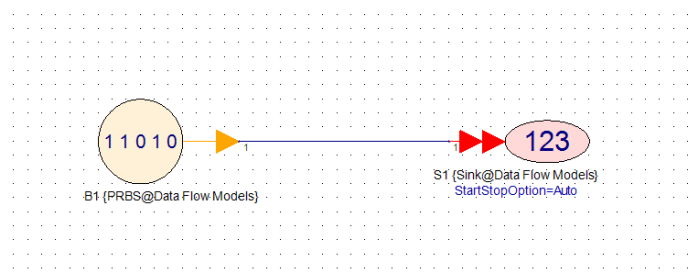


Рис. 9.4. Подключение приёмника к источнику битов.

Полезный совет. Чтобы установить приёмник, можно нажать на клавиатуре "s" и затем щёлкнуть на схеме. Чтобы установить соединение, нажмите "w" (перетащите соединение от одного компонента к другому), чтобы установить фильтр нажмите "f" и т.д.

Щёлкните правой кнопкой на "DF_Bits (Bits)" и выберите "Run (calculate now)" (запустить (рассчитать сразу)). Вы увидите, что к модели добавился набор данных. С помощью этого набора данных можно наблюдать выход.

Чтобы сохранить структуру, щёлкните правой кнопкой на наборе данных и выберите Rename (переименовать). Назовите его “DF1_Bits”.

Поскольку имя набора данных изменилось, нужно убедиться в том, что моделирование выполняется с нужным набором данных.

7. Дважды щёлкните на анализе “DF_Bits” рядом с полем Dataset (набор данных). Напечатайте “DF1_Bits” и щёлкните ОК.

Заметьте, что теперь Data Flow (поток данных) и Dataset окрасились в красный цвет. Это значит, что модель не соответствует актуальной информации, и мы внесли изменения в наш проект.

8. Щёлкните правой кнопкой на “DF_Bits” и выберите Run (calculate now).

После безошибочного моделирования построим диаграмму битов.

9. Дважды щёлкните на наборе данных “DF1_Bits”, он откроется, и вы увидите переменные, соответствующие приёмнику.
10. Щёлкните правой кнопкой на S1 и выберите Add to Graph > New Graph Series Wizard (Добавить к диаграмме > Мастер создания диаграмм) (рис. 9.5).

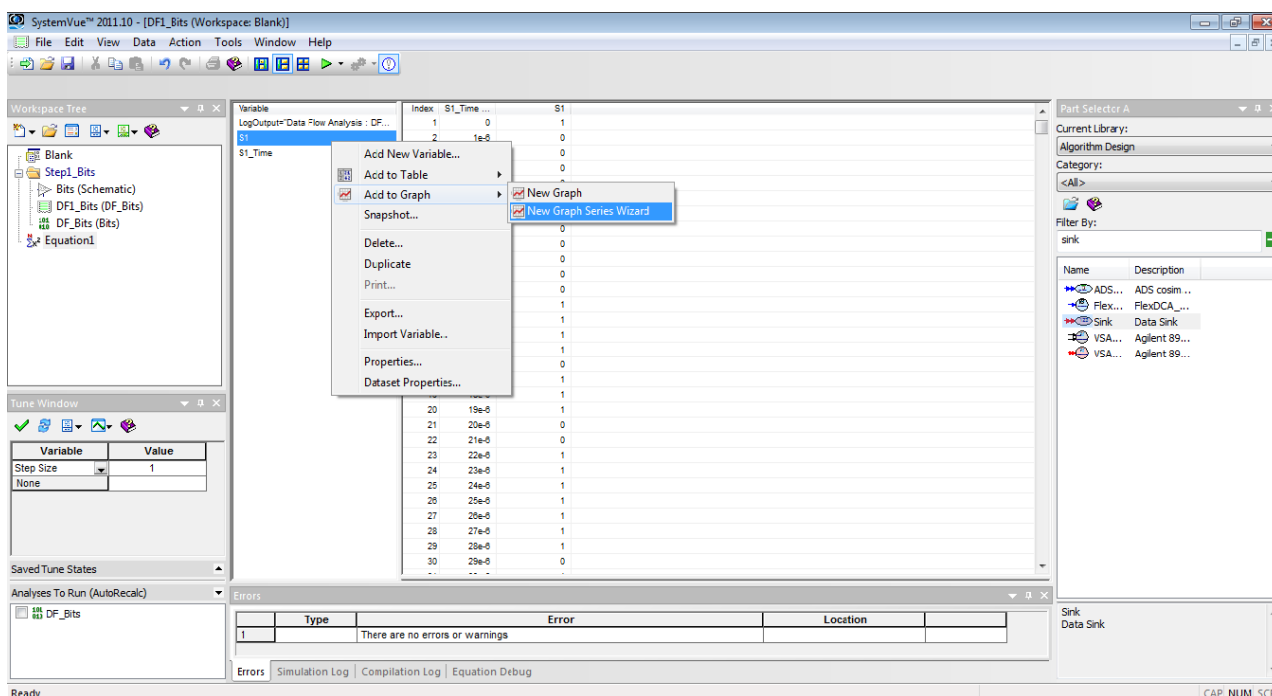


Рис. 9.5. Выбор мастера создания диаграмм.

11. В поле “Type of series selected ” (выбранный тип последовательности) выберите “Time” (время). Убедитесь, что в поле “Data Selected” (выбранные данные) отмечен селектор “S1”, поскольку это и есть выход, диаграмму которого мы хотим построить (рис. 9.6).

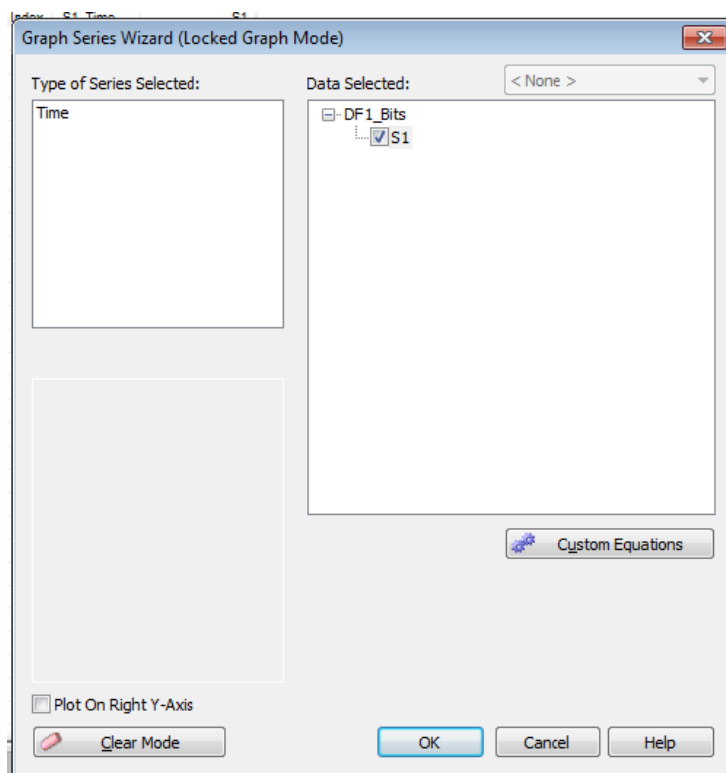


Рис. 9.6. Выберите “S1”, чтобы построить диаграмму этого выхода.

12. Нажмите OK. При этом откроется окно свойств диаграммы, в котором в качестве единственной переменной будет указана “S1” (она будет называться MyVar).
13. Нажмите OK. Откроется график проекта “Bits” во временной области (рис. 9.7).

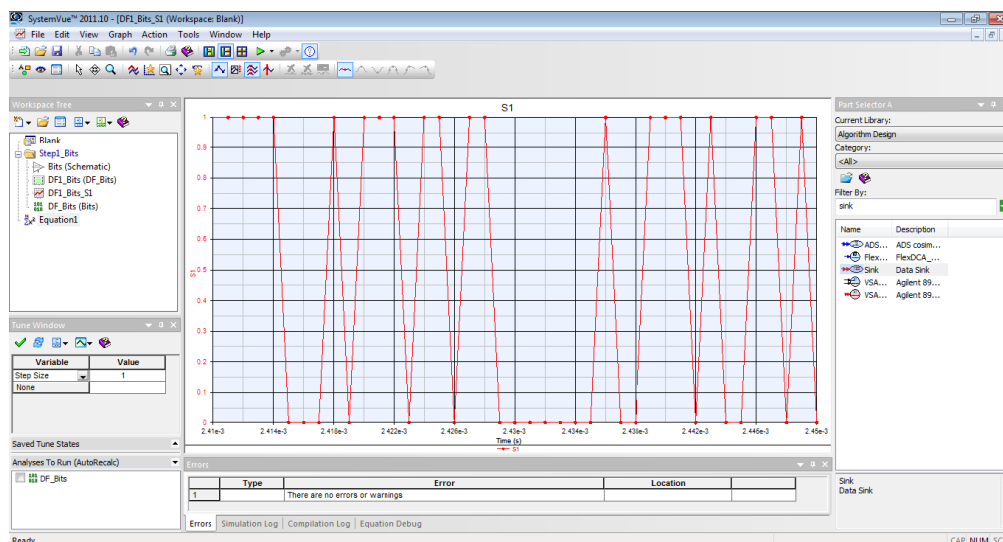


Рис. 9.7. График проекта “Bits”.

14. Увеличьте масштаб изображения по оси X колёсиком мыши, как показано на рис. 9.7. Прокручивая колёсико вперёд, вы более чётко увидите случайные единицы и нули.

Часто для анализа систем связи используются глазковые диаграммы, которые позволяют глубже понять характеристики сигнала.

1. Измените схему так, чтобы она соответствовала рис. 9.8, для чего скопируйте созданную ранее схему "Bits" (убедитесь, что используемые компоненты имеют те же переменные, что и раньше).

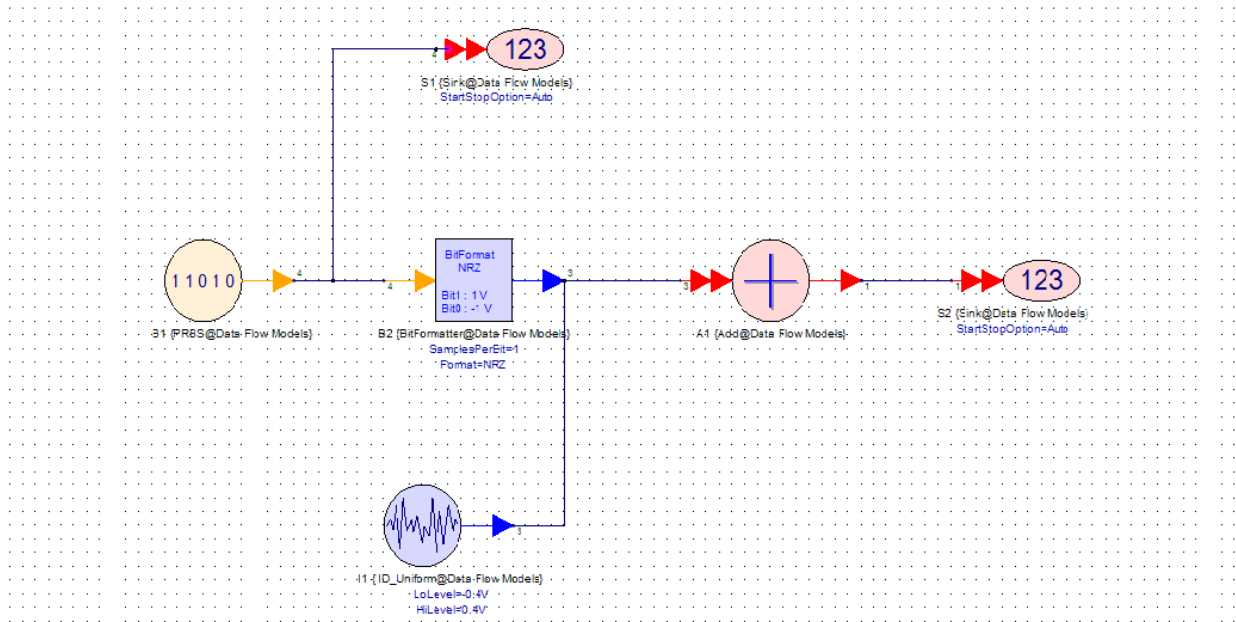


Рис. 9.8. Для схемы можно построить глазковую диаграмму, чтобы лучше понять характеристики сигнала.

2. Напечатав "NRZ" в поле поиска селектора компонентов, можно найти BitFormat. Аналогичным образом для поиска идентификатора источника однородного шума напечатайте "Noise", а чтобы найти в библиотеке компонентов сумматор, выполните поиск по слову "Add". Компонент NRZ преобразует цифровые разряды в заданное напряжение. Для того, чтобы сделать глазковую диаграмму более интересной и реалистичной, мы добавили немного шума.
3. Дважды щёлкните на источнике шума и установите напряжение LoLevel (нижний уровень) на -0,4 В, а напряжение HiLevel (верхний уровень) на 0,4 В. Для этого щёлкните в соответствующих полях и введите значения.
4. Запустите моделирование, щёлкнув правой кнопкой на "DF_Bits" и выбрав Run.
5. Дважды щёлкните на наборе данных, щёлкните правой кнопкой на S2 и выберите Add to Graph > New Graph Series Wizard > Eye > Custom Equations (Добавить к диаграмме > Мастер создания диаграмм > Глазковая > Специальные уравнения). Измените уравнения на:

SymbolRate = 1e6;	% в Гц
NumCycles = 3;	% Число периодов, которые надо построить до сворачивания
StartupDelay = 2;	% Число удаляемых начальных выборок

6. Щёлкнув три раза ОК, закройте все окна. Теперь вы увидите глазковую диаграмму, как на рис. 9.9.

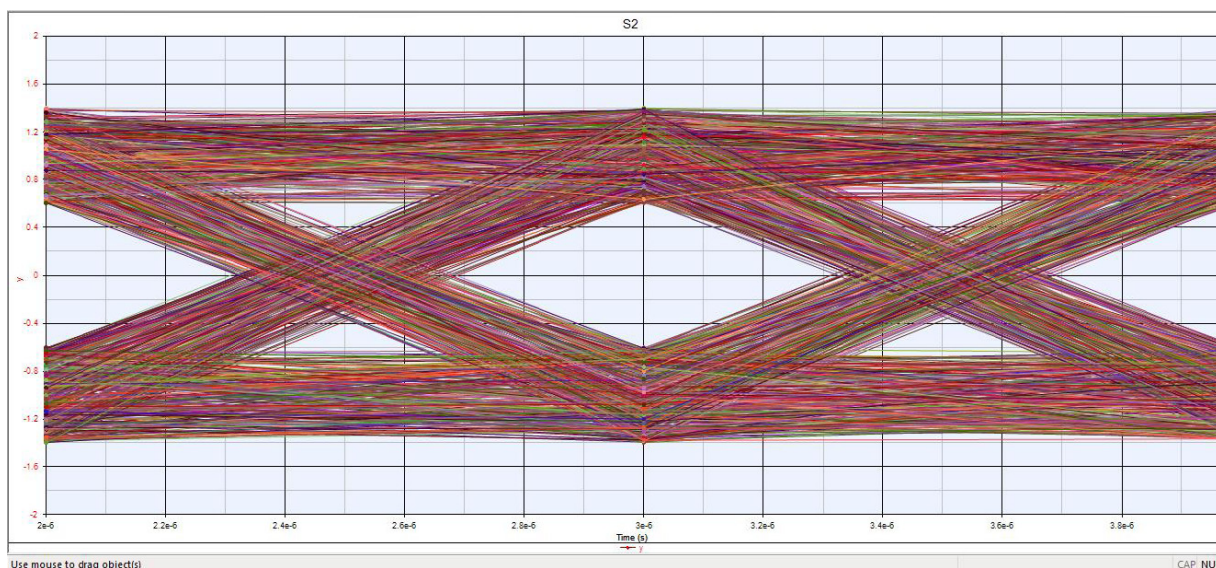


Рис. 9.9. Глазковая диаграмма схемы, показанной на рис. 9.8.

Чтобы изменить диаграмму, дважды щёлкните на ней. При этом откроется диалоговое окно свойств диаграммы. Обратите внимание, что переменная “у” получила название “<продолжение обработки>”. Это значит, что вы можете продолжить обработку данных во временной области для построения глазковой диаграммы. Нажав “edit” (изменить) в диалоговом окне, вы вернётесь в мастер создания диаграмм, где можете изменить уравнения диаграммы.

Лабораторная работа 1 завершена. Не забудьте сохранить рабочую среду.

9.3 Лабораторная работа 2. Формирование символов

В этой лабораторной работе мы преобразуем данные в формат распространённой цифровой модуляции. Затем мы рассмотрим эти данные на сигнальном созвездии.

Создайте новую папку “Step2_Mapping”, для чего выполните следующие шаги.

1. Щёлкните правой кнопкой на ветви “My_CDMA_Sys”, которая расположена в верхней части дерева рабочей среды, и выберите Add > Add Folder (Добавить > Добавить папку). Назовите папку “Step2_Mapping”.
2. Щёлкните правой кнопкой на папке “Step2_Mapping” и выберите Add > Designs > Add Schematic (Добавить > Проекты > Добавить схему). Назовите проект “Mapping” (формирование символов).
3. Добавьте набор данных, для чего щёлкните правой кнопкой на папке “Step2_Mapping” и выберите Add > Add Data (Добавить > Добавить данные).
4. Щёлкните правой кнопкой на папке “Step2_Mapping” и выберите Add > Analysis > Add DataFlowAnalysis (Добавить > Анализ > Добавить анализ потока данных). Назовите его “DF_Mapping”. Проверьте, что информация в всплывающем окне “DataFlow Analysis” соответствует показанной на рис 9.10.

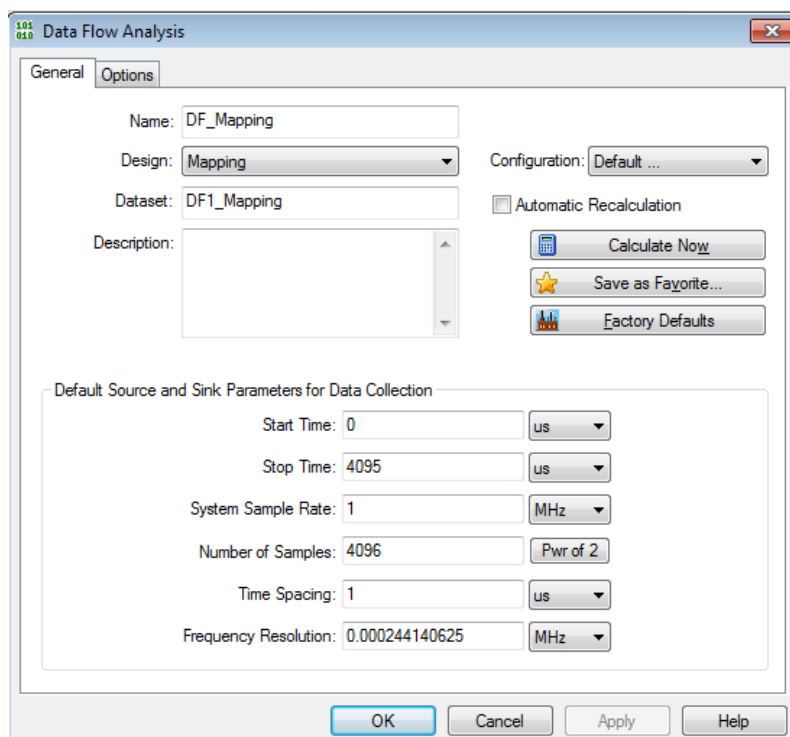


Рис. 9.10. Поля проекта и данных должны соответствовать модели.

Давайте построим распределение битов модуляции QPSK, как показано на рис. 9.11. Для компонента Bits установите параметры так же, как вы делали это в лабораторной работе 1, в пункте 4.

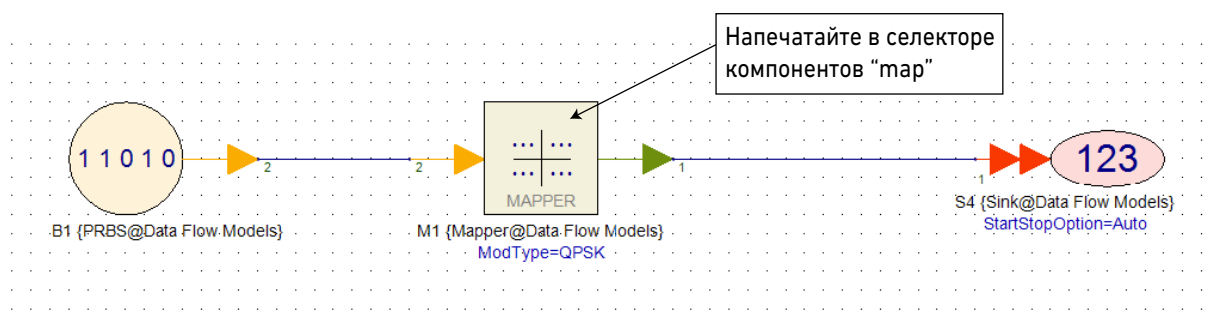


Рис. 9.11. Распределение битов.

Запустите моделирование.

5. Щёлкните правой кнопкой на "DF_Mapping" и выберите Run(Calculate Now).
6. Дважды щёлкните на наборе данных, щёлкните правой кнопкой на приёмнике (S1) и выберите Add to Graph > New Graph Series Wizard > Constellation > Ok > Ok (Добавить к диаграмме > Мастер создания диаграмм > Сигнальное созвездие > OK > OK).

Вы должны увидеть идеальное сигнальное созвездие (рис. 9.12).

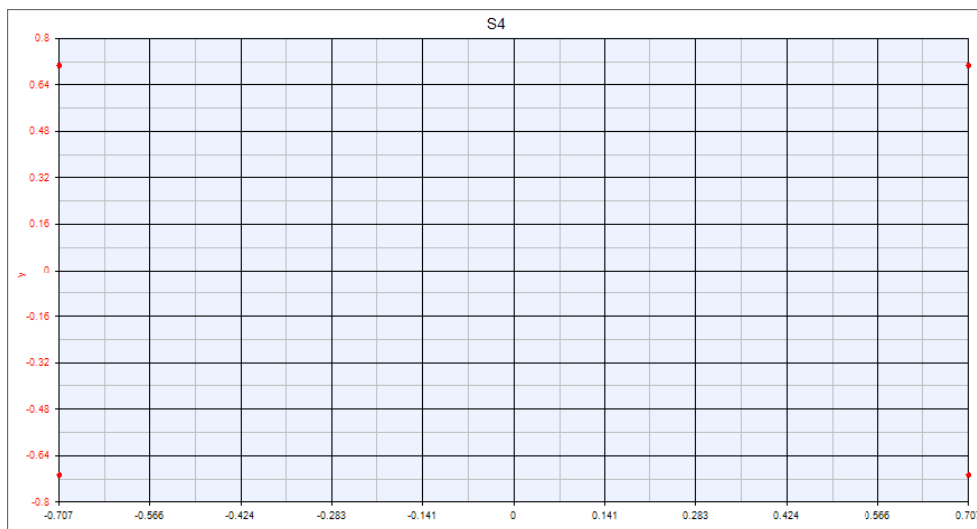


Рис. 9.12. Идеальное сигнальное созвездие.

7. Измените проект, как показано на рис. 9.13.

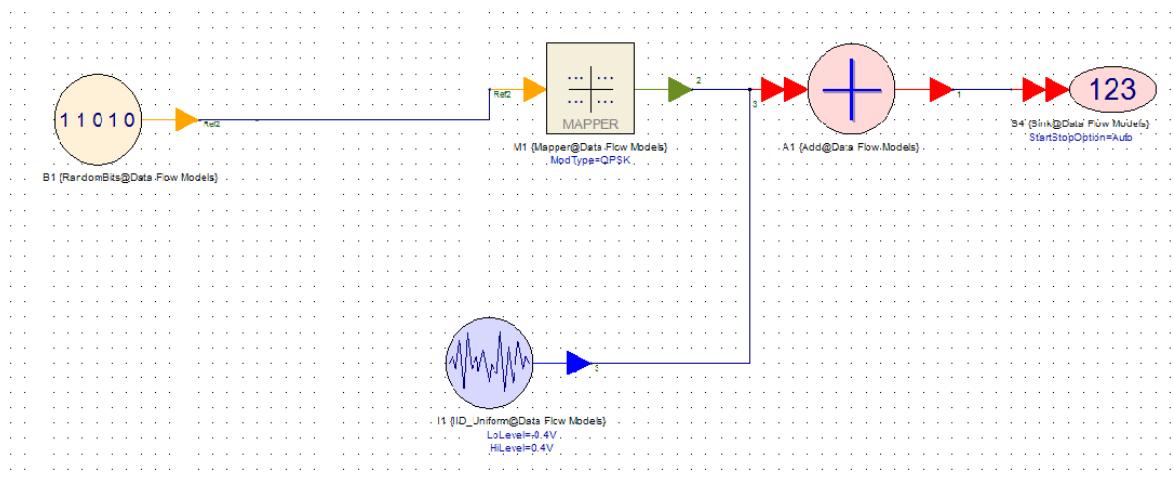


Рис. 9.13. Изменение сигнального созвездия.

Рекомендуется дать соединениям имена. Для этого щёлкните правой кнопкой на соединении и выберите Net > Edit Net name (Цепь > Изменить имя цепи). В данном случае мы выбрали имя Ref.

8. Теперь запустите моделирование, щёлкнув правой кнопкой на "DF_Mapping" и выбрав Run.
9. Вернитесь к предыдущей диаграмме, и вы увидите, что к сигнальному созвездию добавился шум (рис. 9.14).

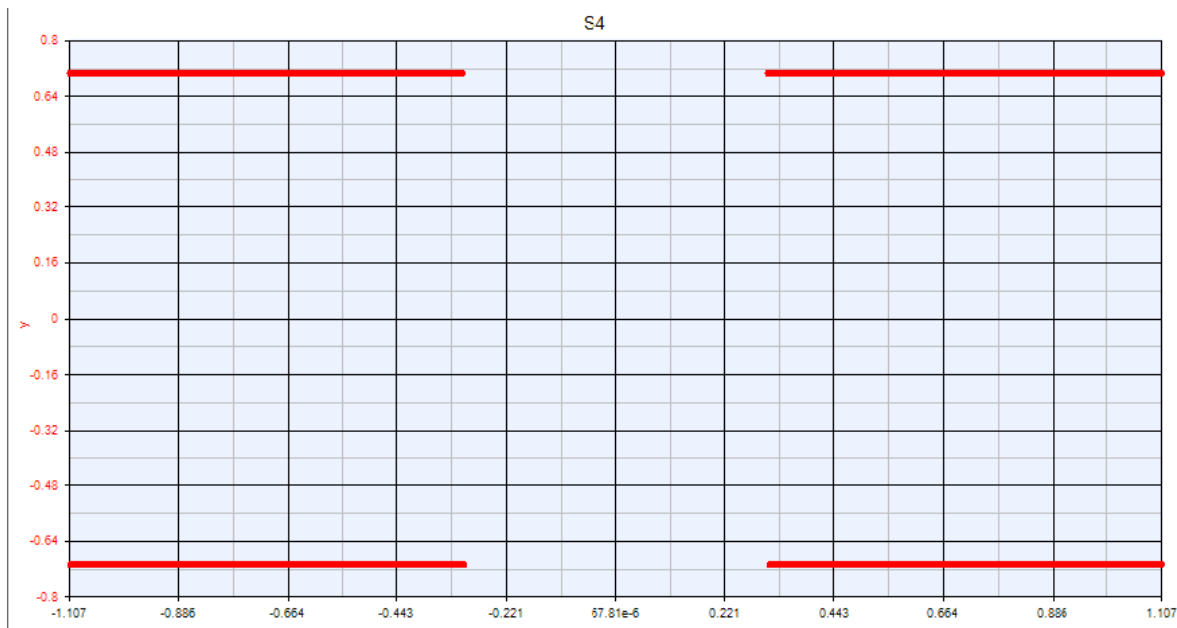


Рис. 9.14. Сигнальное созвездие с добавленным шумом.

Конец лабораторной работы 2.

9.4 Лабораторная работа 3. Код Уолша

В этой лабораторной работе мы создадим код Уолша для наших данных.

1. Щёлкните правой кнопкой на "My_CDMA_Sys" и выберите Add > Add folder. Назовите папку "Step3_WalshCode".
2. Щёлкните правой кнопкой на папке "Step3_WalshCode" и выберите Add > Designs > Add Schematic. Назовите схему "WalshCode".
3. Щёлкните правой кнопкой на папке "Step3_WalshCode" и выберите Add > Analysis > Data Flow Analysis (Добавить > Анализ > Анализ потока данных). Назовите его "DF_WalshCode".
4. Щёлкните правой кнопкой на папке "Step3_WalshCode" и выберите Add > Add Data. Назовите данные "DF1_WalshCode".
5. Щёлкните правой кнопкой на Data Flow Analysis и измените параметры так, чтобы проект и набор данных соответствовали друг другу, то есть, чтобы в поле design было написано "WalshCode", а в поле dataset было написано "DF1_WalshCode". Все остальные параметры должны соответствовать ранее настроенному анализу.

Скопируйте схему из “Step2_Mapping”, и измените проект, как показано на рис. 9.15.

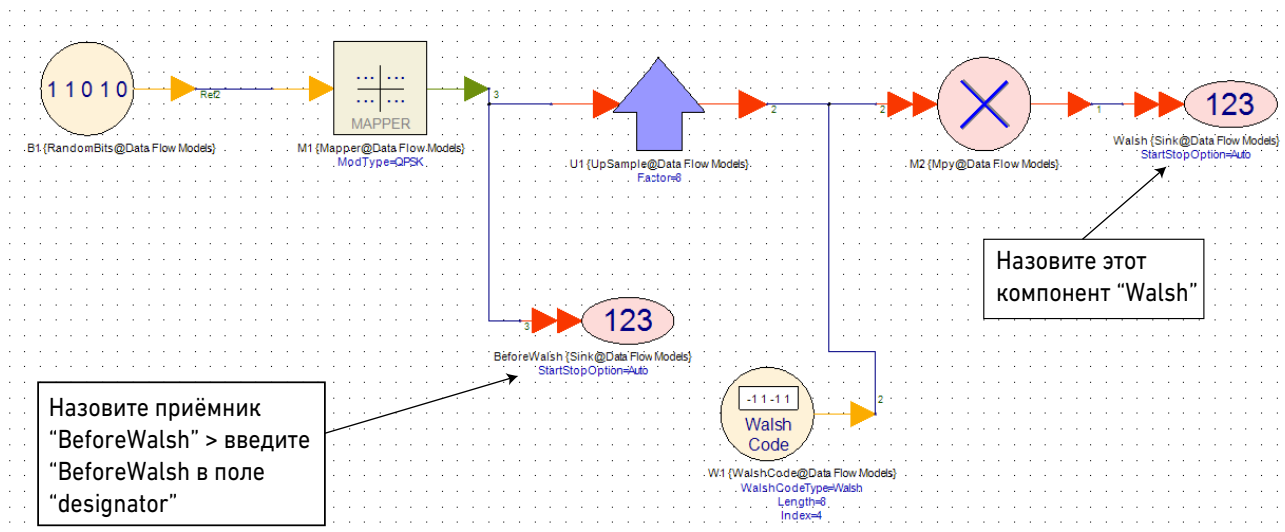


Рис. 9.15. Видоизменённая схема.

После построения графика данных для формирования символов, символьная скорость снижается в 4 раза из-за двойной поляризации и QPSK. Передискретизатор до некоторой степени повышает частоту дискретизации. В данном случае мы повышаем частоту дискретизации в 8 раз, и символьная скорость становится равной 4 МГц, что создаёт достаточную полосу для передачи на несущей по каналу связи.

6. Дважды щёлкните на компоненте Up Sample (повышающий передискретизатор) и установите множитель 8. Выберите режим “HoldSample”. Входная выборка повторится на выходе 8 раз.
7. Дважды щёлкните на компоненте WalshCode Generator (генератор кода Уолша). Установите значения, показанные на рис. 9.16.

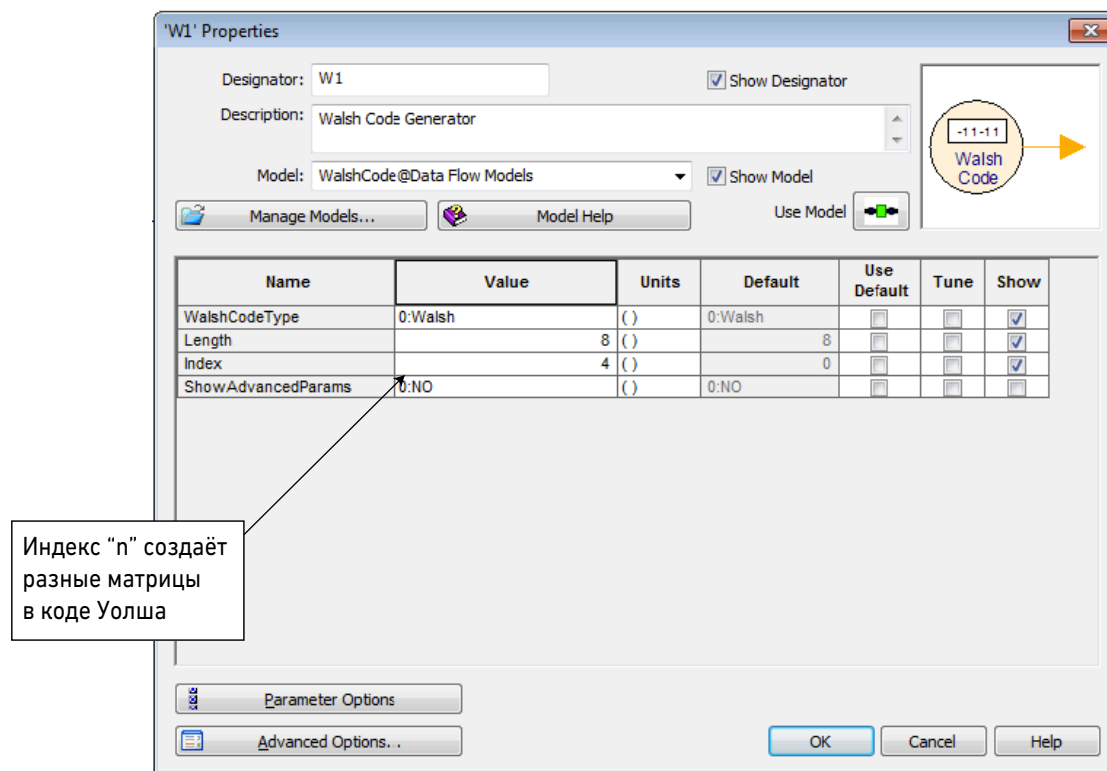
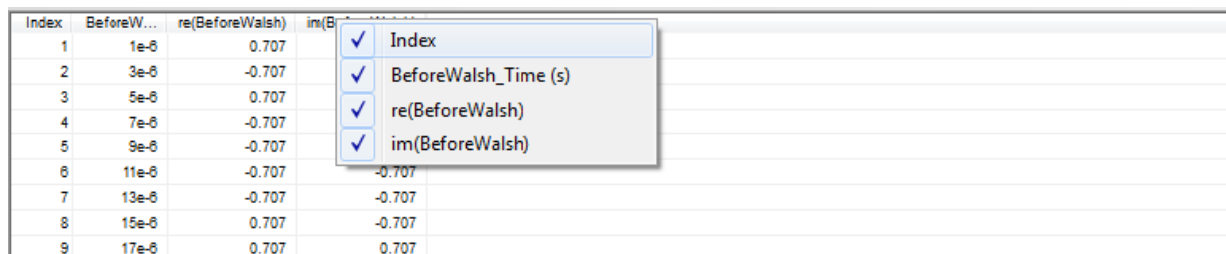


Рис. 9.16. Установка параметров генератора кода Уолша.

Давайте рассмотрим значения кода до и после генерации кода Уолша.

8. Запустите моделирование.
9. Дважды щёлкните на наборе данных, щёлкните правой кнопкой на "BeforeWalsh" (до Уолша) и выберите Add to Table > New Table (Добавить в таблицу > Создать таблицу). Щёлкните «Close» (заккрыть).
10. Щёлкните правой кнопкой на верхней части таблицы рядом с именами значений (рис. 9.17).



Index	BeforeWalsh...	re(BeforeWalsh)	im(B...
1	1e-8	0.707	
2	3e-8	-0.707	
3	5e-8	0.707	
4	7e-8	-0.707	
5	9e-8	-0.707	
6	11e-8	-0.707	-0.707
7	13e-8	-0.707	-0.707
8	15e-8	0.707	-0.707
9	17e-8	0.707	0.707

The context menu is open over the 'Index' column header, showing the following items with checkmarks:

- Index
- BeforeWalsh_Time (s)
- re(BeforeWalsh)
- im(BeforeWalsh)

Рис. 9.17. В таблице щёлкните на "Index" (индекс).

11. Щёлкните на "Index" и на "BeforeWalsh_Time(s)". Вы увидите, что в таблице пропали соответствующие столбцы.
12. Дважды щёлкните на наборе данных Walsh Code (код Уолша), щёлкните правой кнопкой на Walsh и выберите Add to Table > Add to "DF_WalshCode_BeforeWalsh".
13. Разверните окно на весь экран. Столбец "Walsh_Time(s)" можно удалить, потому что он не нужен.

Рассмотрите и сравните значения до и после генерации кода Уолша. Обратите внимание, что они отличаются. Это значит, что наш код надёжно защищён кодом Уолша. Теперь мы должны сделать так, чтобы такой же код Уолша применялся в приёмнике для декодирования данных и извлечения исходной информации. В последней лабораторной работе мы используем некоторые примеры и посмотрим, что происходит с отдельными битами, когда они сопоставляются с кодом Уолша.

Дополнительное упражнение

Измените индекс генератора кода Уолша через его свойства. Снова создайте таблицу значений до генерации кода Уолша и после неё. Обратите внимание на отличие значений.

Лабораторная работа 3 завершена.

9.5 Лабораторная работа 4. Проектирование фильтра

В этой лабораторной работе мы создадим папку для расчёта полосы модуляции.

1. Создайте папку и назовите её "Step4_Filter".
2. Добавьте схему с именем "Filter" (фильтр).
3. Добавьте набор данных с именем "DF1_Filter".
4. Добавьте анализ потока данных с именем "DF_Filter". Установите Number of Samples (число выборок) 4096.

Скопируйте схему, созданную на шаге 3, и вставьте её во вновь созданную пустую схему. Измените проект, как показано на рис. 9.18.

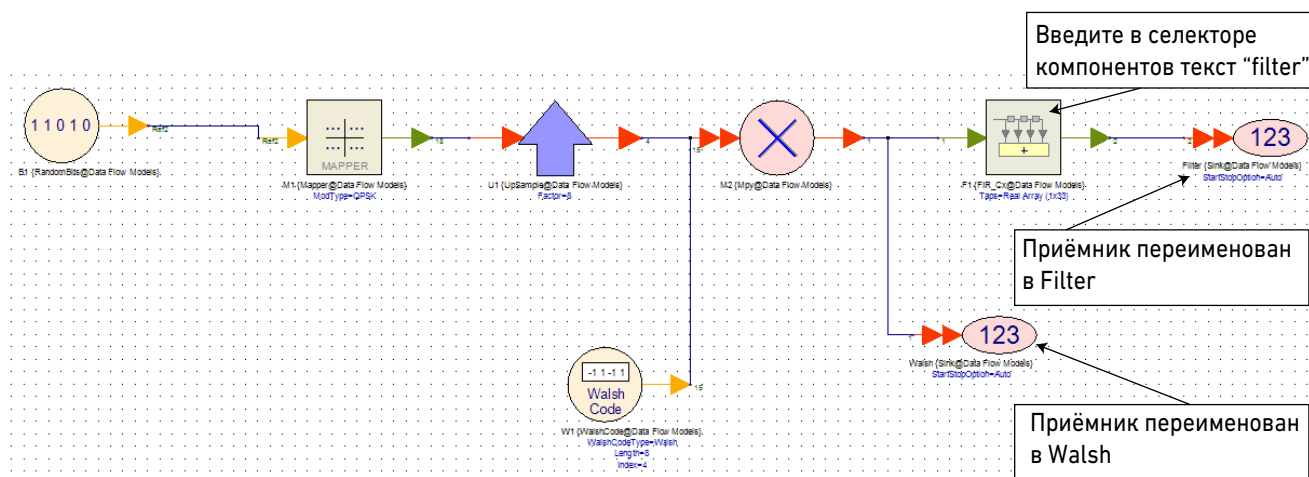


Рис. 9.18. Измените проект, как показано на этом рисунке.

5. Дважды щёлкните на компоненте Filter. Откроются свойства проекта. Нажмите на кнопку Filter Designer (конструктор фильтров). Измените свойства так, чтобы они соответствовали рис. 9.19. Внесите изменения, закройте страницу свойств.

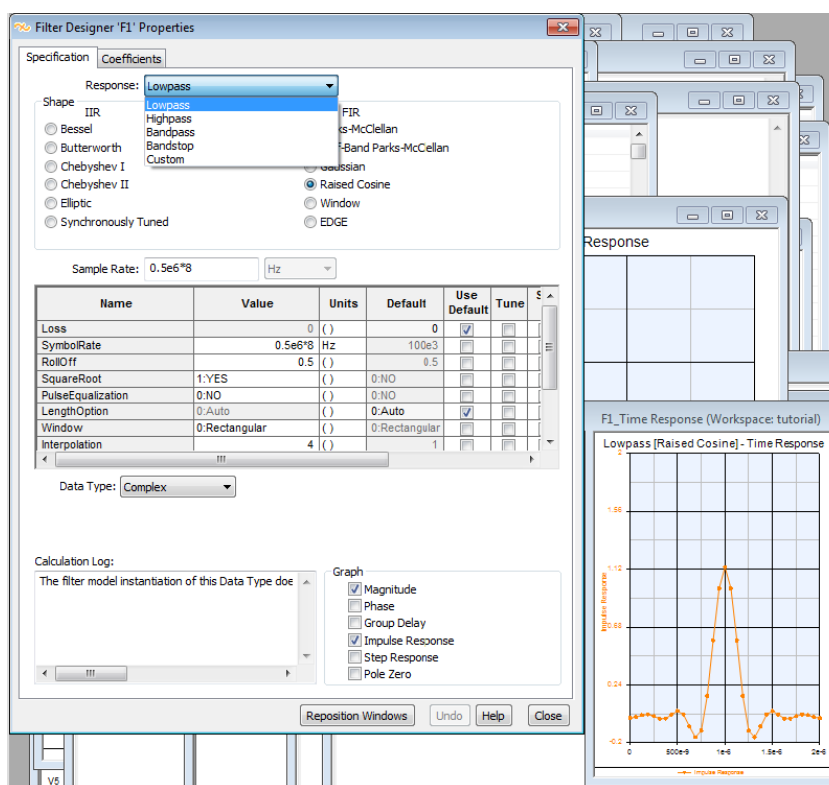


Рис. 9.19. Меню свойств проекта.

6. Запустите анализ.

7. Дважды щёлкните на наборе данных DF1_Filter, щёлкните правой кнопкой на Walsh и выберите Add to Graph > New Graph Wizard > Spectrum (Добавить к диаграмме > Мастер новых диаграмм > Спектр). Дважды нажмите ОК. Появится экран, показанный на рис. 9.20.

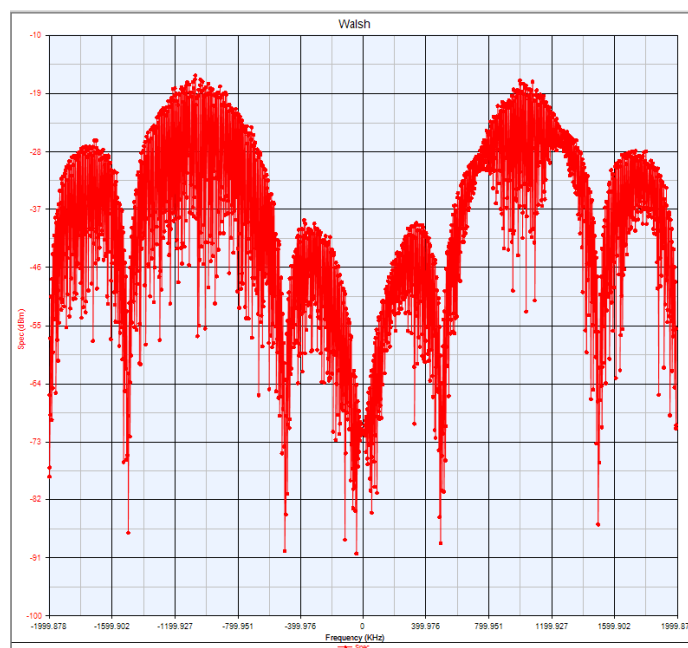


Рис. 9.20. Результирующая диаграмма.

8. Теперь постройте спектр для "Filter". Дважды щёлкните на наборе данных фильтра (DF1_Filter), щёлкните правой кнопкой на Filter и выберите Add to Graph > New Graph Wizard > Spectrum. Дважды нажмите ОК. Вы увидите, как это отразится на фильтре (рис. 9.21).

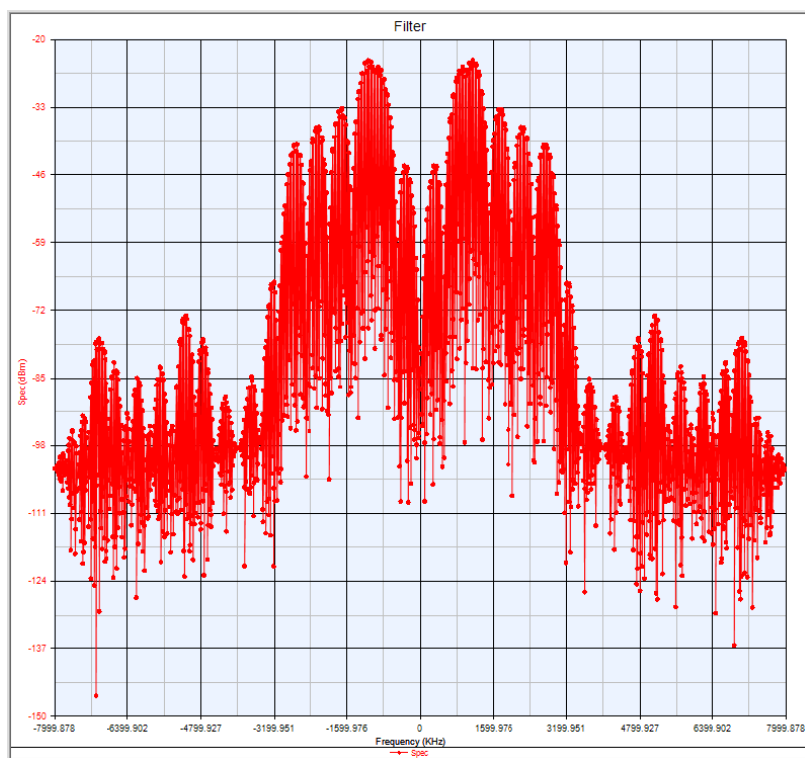


Рис. 9.21. Диаграмма спектра.

Конец лабораторной работы 4.

9.6 Лабораторная работа 5. Модуляция

Создайте новую папку и назовите её “Step5_Modulation”. Добавьте новую схему с именем “Modulation” (модуляция). Добавьте анализ потока данных (“DF_Modulation”) и набор данных (“DF1_Modulation”). Установите число выборок анализа потока данных 4096.

1. Скопируйте схему, созданную на шаге 4, и вставьте её во вновь созданную пустую схему.
2. Удалите выходной приёмник.
3. В селекторе компонентов выполните поиск по ключевому слову “cxt” и выберите компонент “Cxt To Rect”. Он нужен для разделения реальной и мнимой части сигнала.
4. Присоедините этот компонент к выходу фильтра.
5. Добавьте модулятор (введите “modulator” для поиска в библиотеке селектора компонентов).
6. Присоедините его к выходу преобразователя комплексных чисел в реальную и мнимую часть.
7. Установите компонент плотности шума, напечатав в селекторе компонентов “noise density”. Подключите к выходу анализатор спектра.
8. Измените параметры компонентов, как показано на рис. 9.22.

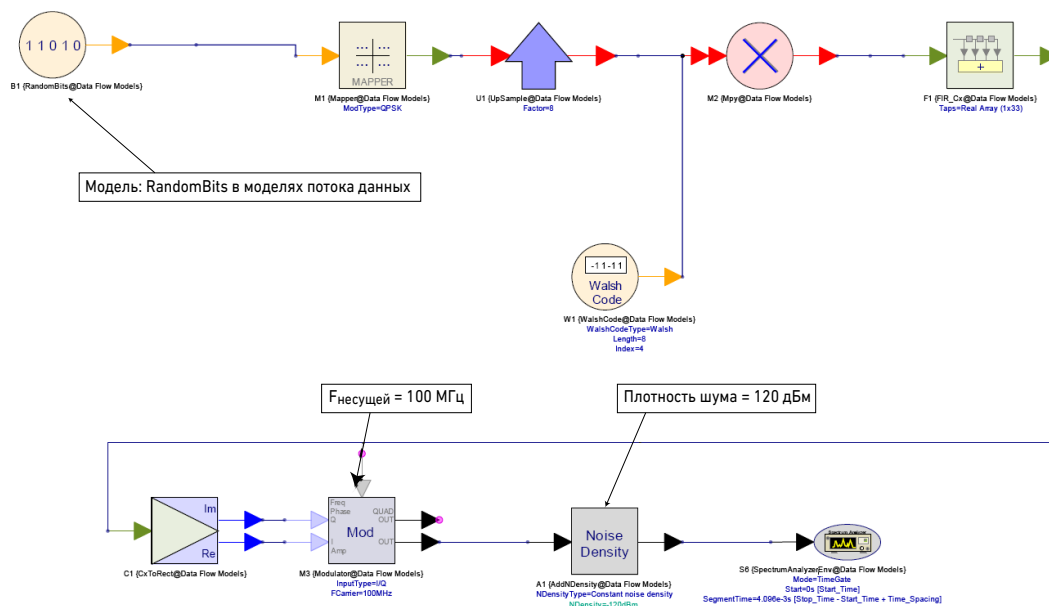


Рис. 9.22. Установите в схему компонент плотности шума.

9. Давайте рассмотрим спектр мощности в SystemVue с помощью анализатора спектра. Дважды щёлкните на наборе данных "DF1_Modulation", щёлкните правой кнопкой на 'S6_Power' и выберите Add to the Graph > New Graph Series Wizard > General (Добавить в диаграмму > Мастер создания диаграмм > Общие). Дважды нажмите ОК (рис. 9.23).

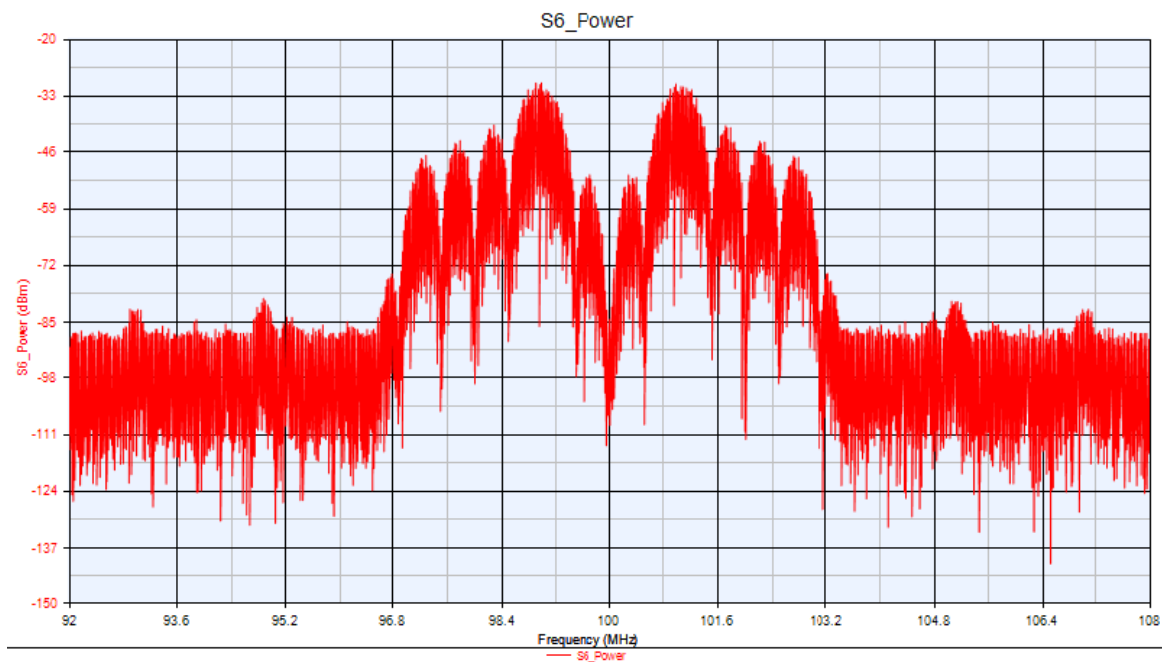


Рис. 9.23. Спектр мощности.

Лабораторная работа 5 завершена.

9.7 Лабораторная работа 6. ВЧ канал

В этой лабораторной работе мы спроектируем ВЧ канал для передачи модулированного сигнала в приёмник.

1. Добавьте новую папку для шага 6. Назовите её "Step6_RFLink".
2. Создайте папку с именем "Step6_RFLink" и вставьте в неё схему (RFLink).
3. В пустой схеме перейдите в селектор компонентов, прокрутите меню "Current Library" (текущая библиотека) и выберите "RF Design" (ВЧ проекты).
4. Оставьте категорию "All", чтобы отображались все компоненты.
5. Создайте ВЧ канала в соответствии с приведённым ниже рисунком. Чтобы выбрать компоненты для схемы, введите в селекторе компонентов показанные на рис. 9.24 сокращения.

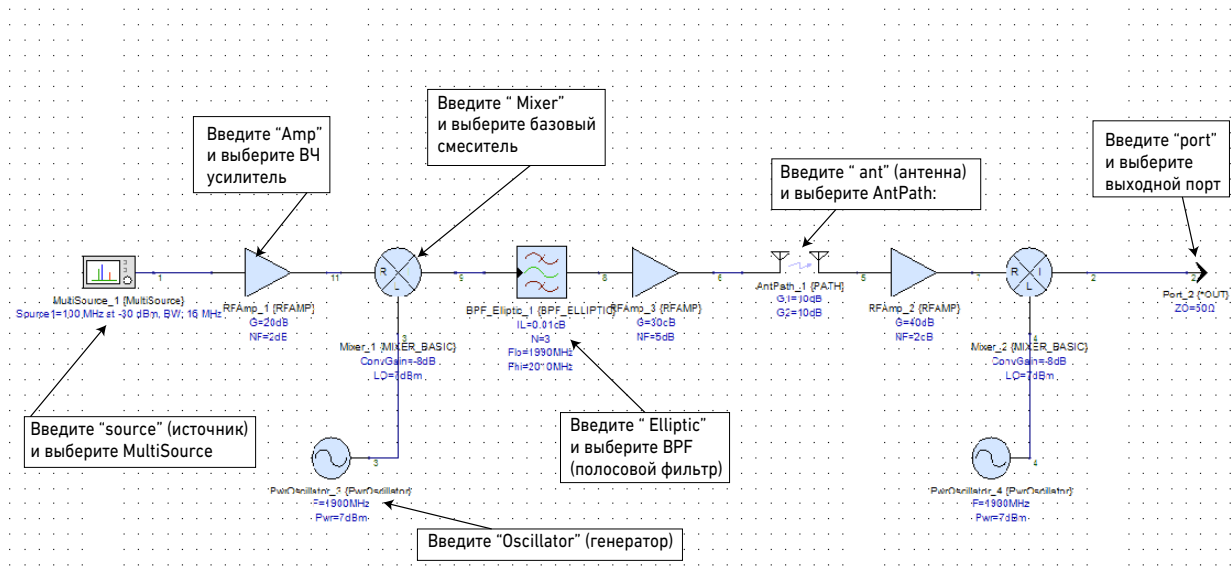


Рис. 9.24. Проект ВЧ канала.

С помощью колёсика мыши увеличьте масштаб страницы и установите параметры. Все параметры подробно описаны ниже.

Параметры:

- MultiSource: дважды щёлкните на компоненте и в окне свойств источника щёлкните на "edit" (изменить). Убедитесь, что селектор установлен на "wideband" (широкая полоса), и полоса равна 16 МГц. Установите центральную частоту 100 МГц и мощность -30 дБм. Оставьте фазу в 0 градусов.
- ВЧ усилитель: дважды щёлкните на компоненте. Установятся значения по умолчанию. Измените коэффициент шума на 2 дБ. Дважды проверьте, что усиление равно 20 дБ.
- Смеситель. По умолчанию должны быть установлены значения: усиление -8 дБм и гетеродин (LO) 7 дБм. Установите параметр SUM в значение 1:Sum.
- Генератор: установите частоту 1900 МГц. Мощность по умолчанию должна равняться 7 дБм.
- Полосовой фильтр: установите нижнюю частоту среза (Flo) 1990 МГц, а верхнюю частоту среза (FHi) 2010 МГц. Вносимые потери уже должны быть установлены на 0,01, а порядок фильтра на 3.
- ВЧ усилитель на выходе полосового фильтра: установите усиление 30 дБ, а коэффициент шума 5 дБ.
- Антенна: установите G1=10 дБ и G2=10 дБ.
- ВЧ усилитель на выходе антенны: установите усиление 40 дБ, а коэффициент шума 2 дБ.
- Смеситель после ВЧ усилителя: установите ConvGain на -8 дБ и LO на 7 дБм. Кроме того, установите параметр SUM в значение 1:Sum.
- Мощный генератор: установите частоту 1900 МГц и мощность 7 дБм.

Добавьте выходной порт и установите его сопротивление в 50 Ом.

6. Щёлкните правой кнопкой на папке Step6_RFLink и выберите Add > Add data (для набора данных). Назовите набор данных "System1_Data".
7. Снова щёлкните правой кнопкой на папке Step6_RFLink и выберите Add > Analysis > Add RF System Analysis (Добавить > Анализ > Анализ ВЧ систем).
8. Дважды щёлкните на анализе (рис. 9.25).

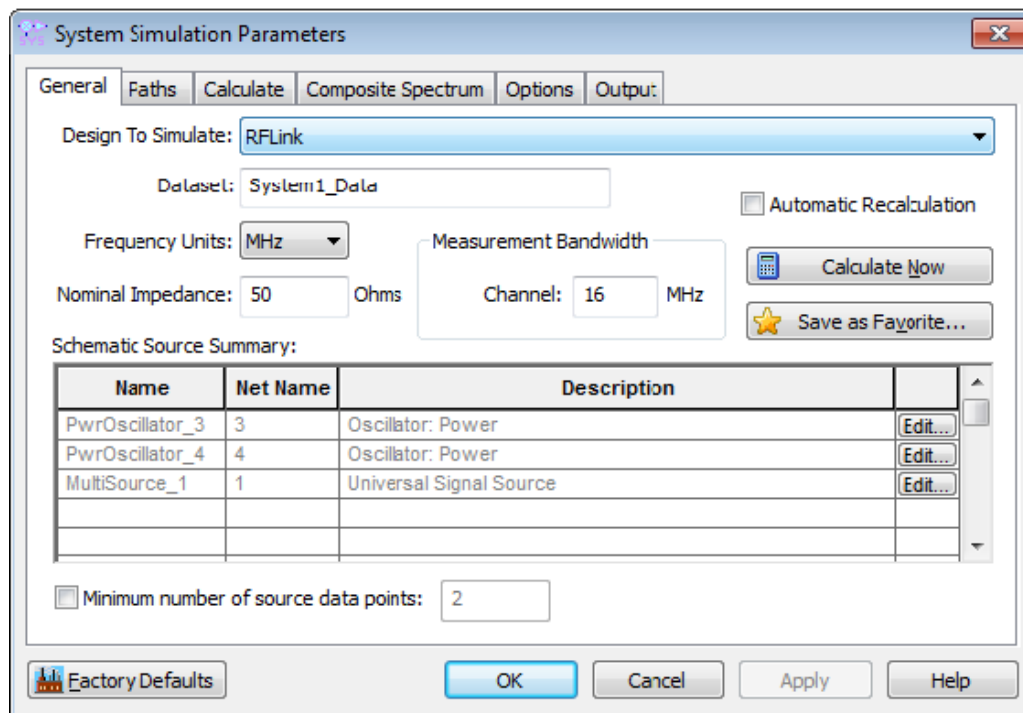


Рис. 9.25. Окно параметров моделирования системы.

На вкладке **General** (общие) убедитесь, что в поле моделируемого проекта (design to simulate) стоит значение RFLink, и что выбран набор данных (dataset) "System1_Data".

9. Перейдите на вкладку "Paths" (маршруты). Выберите "Add all paths from all sources" (добавить все маршруты из всех источников).
10. Нажмите OK.
11. Запустите моделирование, щёлкнув правой кнопкой на System1 (RFLink) и выбрав Run. Вы можете обнаружить, что создана дополнительная папка "System1_Data_Folder".

На схеме наведите курсор на значения. Для начала давайте рассмотрим значение на выходе мультиисточника. Если навести курсор на значение, то рядом со стрелкой появится прямоугольный курсор.

12. Щёлкните правой кнопкой (рис. 9.26).

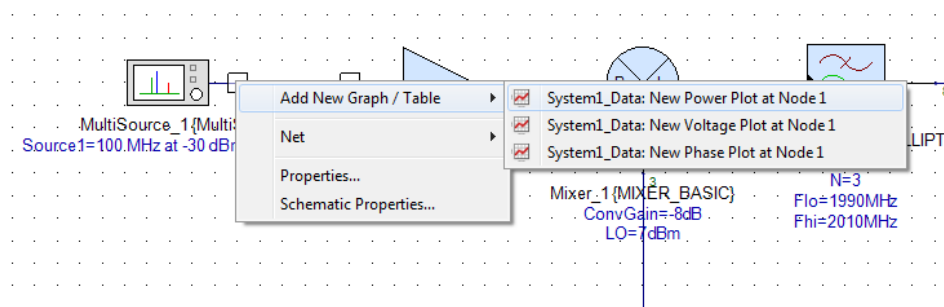


Рис. 9.26. На схеме выберите сначала значение на выходе мультиисточника.

13. Выберите Add New Graph/Table > Select (Добавить новую диаграмму/таблицу > Выбрать), чтобы построить диаграмму мощности (рис. 9.27).

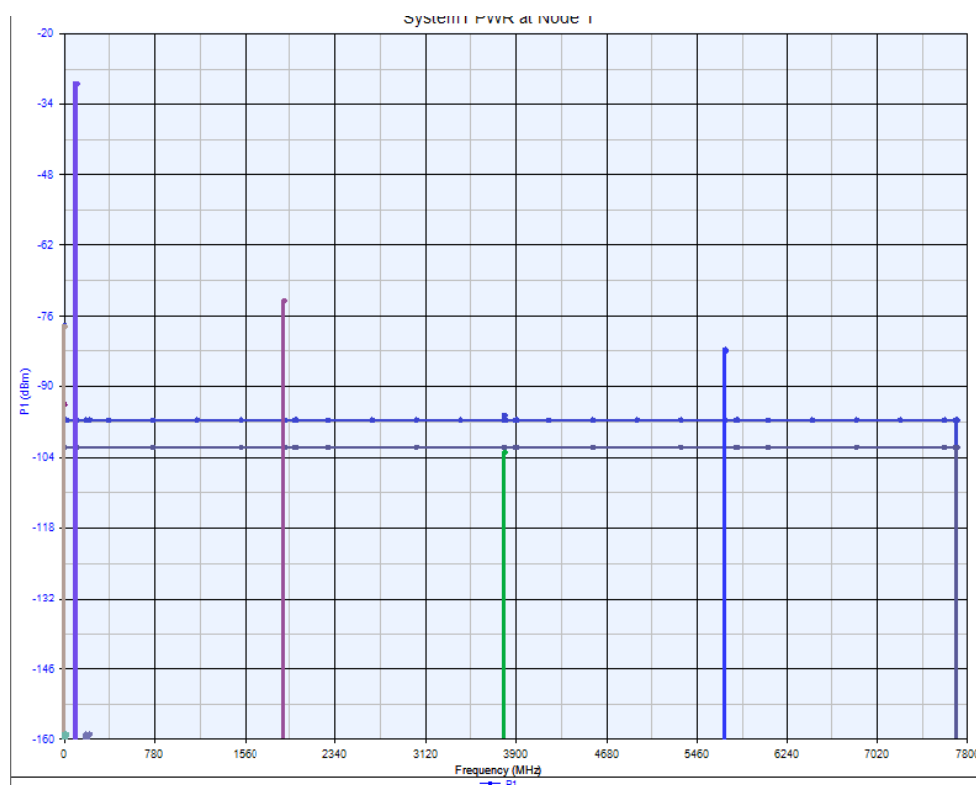


Рис. 9.27. Диаграмма мощности.

Здесь мы можем наблюдать, насколько чист сигнал нашей системы, и присутствуют ли источники помех. Сигнал данных занимает частоту 100 МГц (заданную нашим источником). Это центральная частота, и установлена она достаточно высоко для ВЧ канала связи. Кроме того, мы видим на этой диаграмме другие частотные составляющие и можем подтвердить, что сигналы, создающие помехи для наших данных, отсутствуют.

14. Отобразите мощность на выходном порте (рис. 9.28).

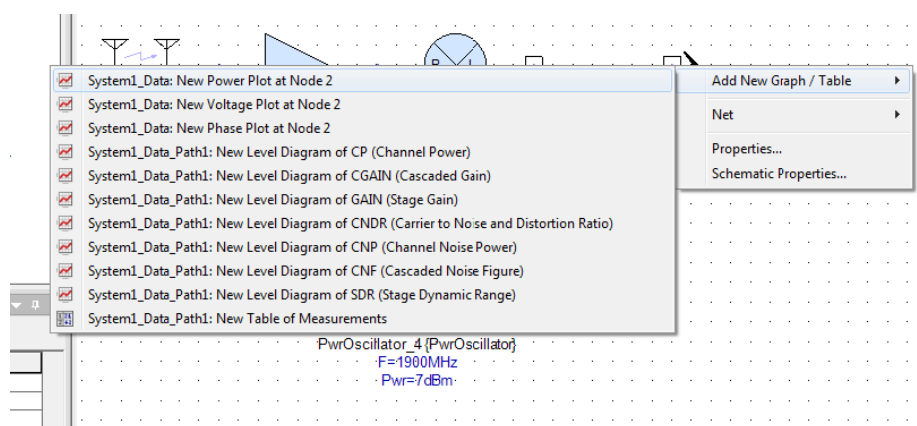


Рис. 9.28. Отображение сигнала на выходном порте.

Постройте новую диаграмму мощности (рис. 9.29).

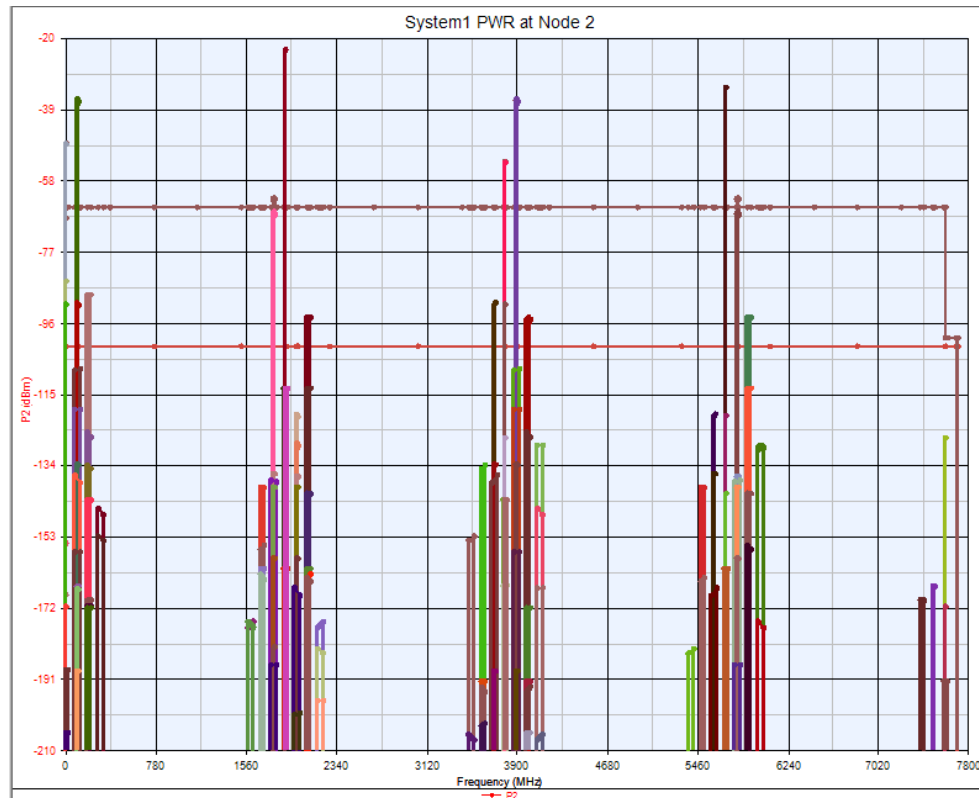


Рис. 9.29. Диаграмма мощности.

Мы можем наблюдать уровень нашего сигнала на частоте 1900 МГц, которая является центральной частотой. Вокруг частоты несущей видны гармоники. Кроме того, мы видим другие частотные составляющие на верхних и нижних частотах, демонстрирующие реальные условия системы передачи. Также мы можем заключить, что после передачи по каналу связи уровень нашего сигнала ещё достаточно высок и, следовательно, мы можем выполнить его демодуляцию и извлечение исходных передаваемых данных без ошибок.

На этом лабораторная работа 6 завершена.

9.8 Лабораторная работа 7. Демодулятор

В этой лабораторной работе мы демодулируем сигнал и спроектируем приёмную часть системы.

1. Создайте новую папку с именем "Step7_Demodulation". Добавьте новую пустую схему и назовите её "Demodulation" (демодуляция). Добавьте анализ потока данных с именем "DF_Demodulation". Создайте набор данных с именем "DF1_Demodulation".
2. Скопируйте проект, созданный на этапе 5 Step5_Modulation, и вставьте его в новую пустую схему "Demodulation".
3. В дереве рабочей среды разверните папку "Step6_RFLink". Щёлкните на схеме RFLink и перетащите её в любое место текущей схемы "Demodulation".

Замечательной функцией SystemVue является возможность изменять имена цепей на те, что вам нужны, и выполнять соединения без физических проводов, не нарушая организации проекта.

4. Щёлкните на соединении между "Noise density" (плотность шума) и "Spectrum Analyzer" (анализатор спектра) (рис. 9.30).

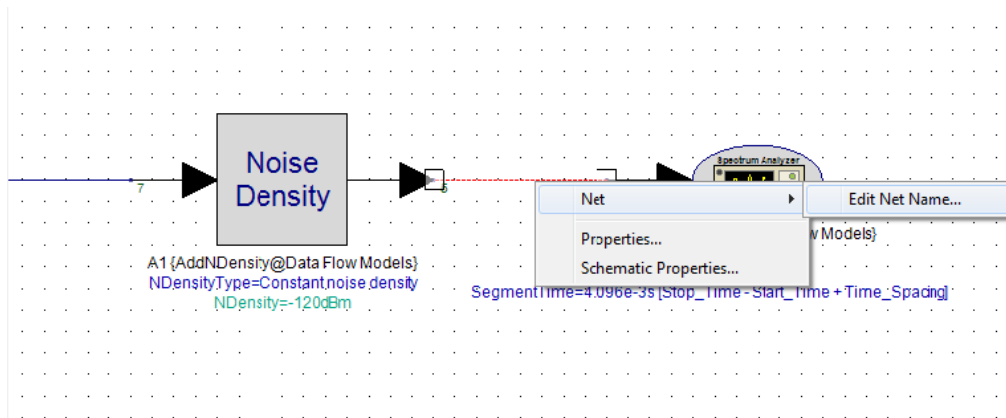


Рис. 9.30. Измените имя цепи.

5. Щёлкните правой кнопкой на соединении и выберите Net > Edit Net Name (Цепь > Изменить имя цепи). Введите новое имя "IF_in".

Слева от RF_Link создайте соединение, позволяющее нам выделить эту цепь, и дайте ей имя (рис. 9.31).

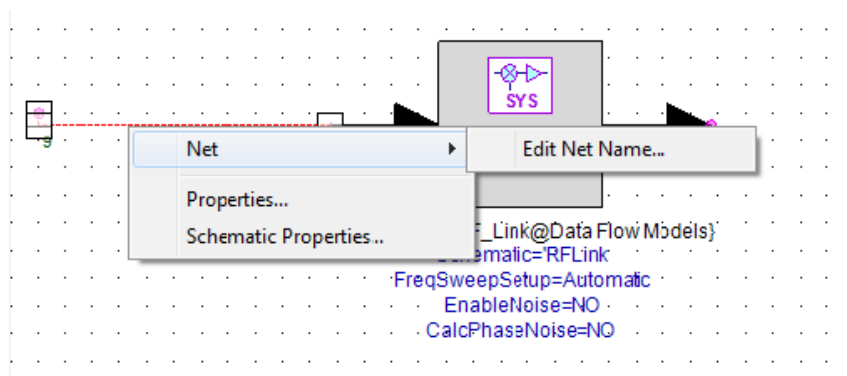


Рис. 9.31. Создайте соединение, выделите цепь и дайте ей имя.

Дайте соединению то же имя, что и выходному соединению компонента плотности шума ("IF_in"). Это соединит оба компонента между собой. Теперь компонент плотности шума и RF_Link соединены.

6. Создайте систему демодуляции. В сущности, эта система зеркальна уже созданной системе модуляции.
7. Добавьте после RF_Link анализатор спектра. Вам может понадобиться изменить "Current Library"

(текущая библиотека) в селекторе компонентов на Algorithm Design (конструктор алгоритмов). Мы будем наблюдать снижение мощности после передачи. Скопируйте соответствующие компоненты из Step5_Modulation, и их параметры тоже сохранятся (рис. 9.32).

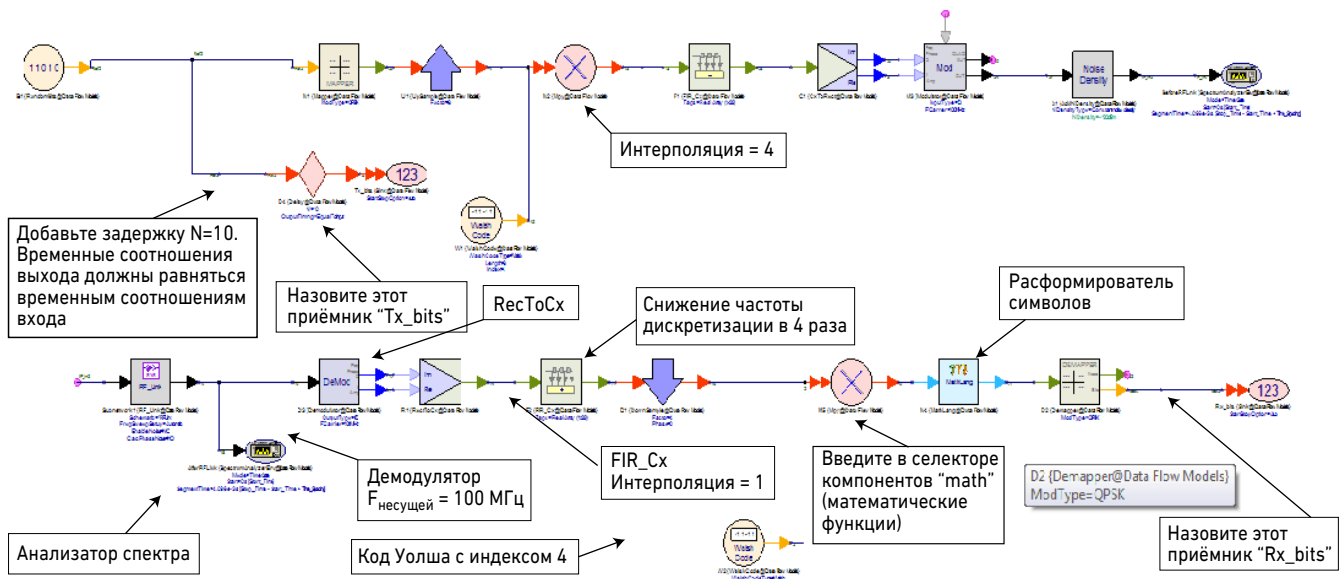


Рис. 9.32. Добавление анализатора спектра.

Модель Math Lang использует для обработки входных данных язык математических уравнений и создаёт выходные данные. Мы можем использовать это для поддержания нужной частоты дискретизации при расформировании символов.

9. Дважды щёлкните на компоненте Math Lang и на вкладке Equations (уравнения) скопируйте то, что показано на рис. 9.33 ($\text{output}=\text{sum}(\text{input})/8$).

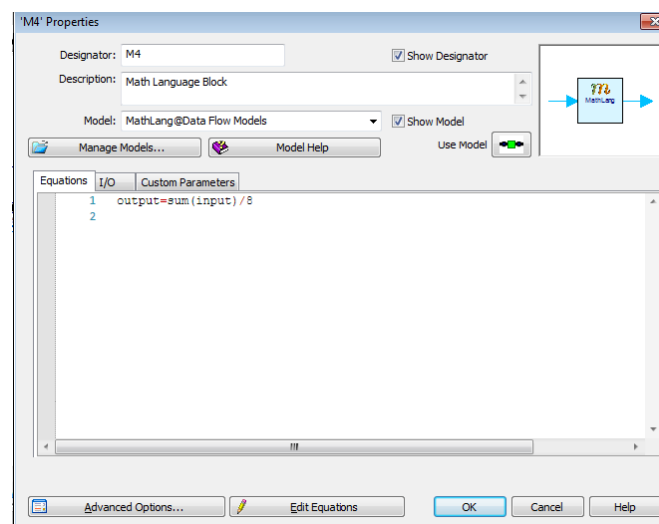


Рис. 9.33. Скопируйте информацию на вкладке Equations.

На вкладке "I/O" (ввод/вывод) установите число входных портов равным 8 (число выходных портов оставьте равным 1).

10. Щелкните на OK.

11. Запустите моделирование, следя за тем, чтобы анализ потока данных получил корректные значения (рис. 9.34).

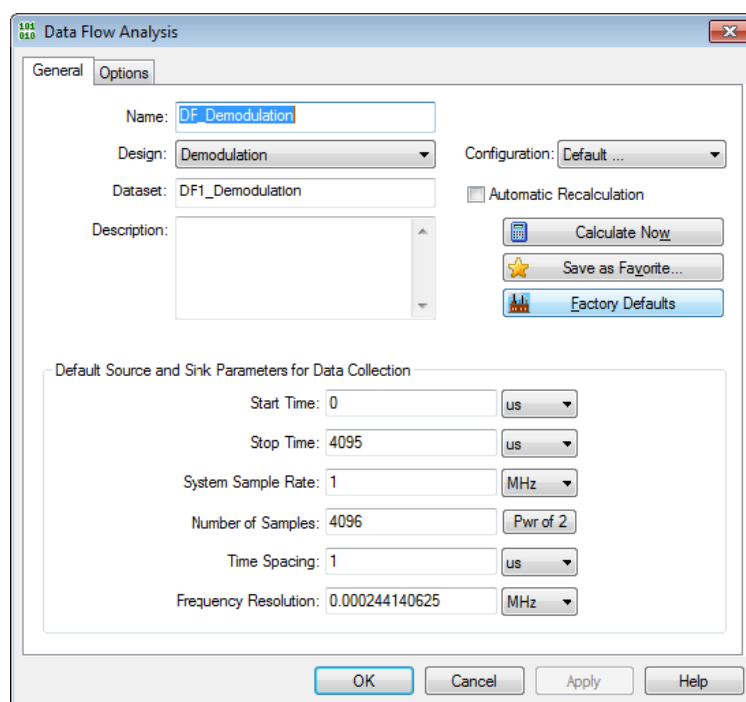


Рис. 9.34. Используйте для моделирования показанные здесь значения.

Давайте рассмотрим мощность на анализаторе спектра до и после канала RFLink, по которому передаётся поток данных.

12. Щёлкните на наборе данных и создайте диаграмму спектра мощности сигнала до RFLink. Дважды щёлкните на наборе данных "DF1_Demodulation" и, щёлкнув правой кнопкой на 'S6_Power', выберите Add to the Graph > New Graph Series Wizard > General. Дважды нажмите на OK (рис. 9.35).

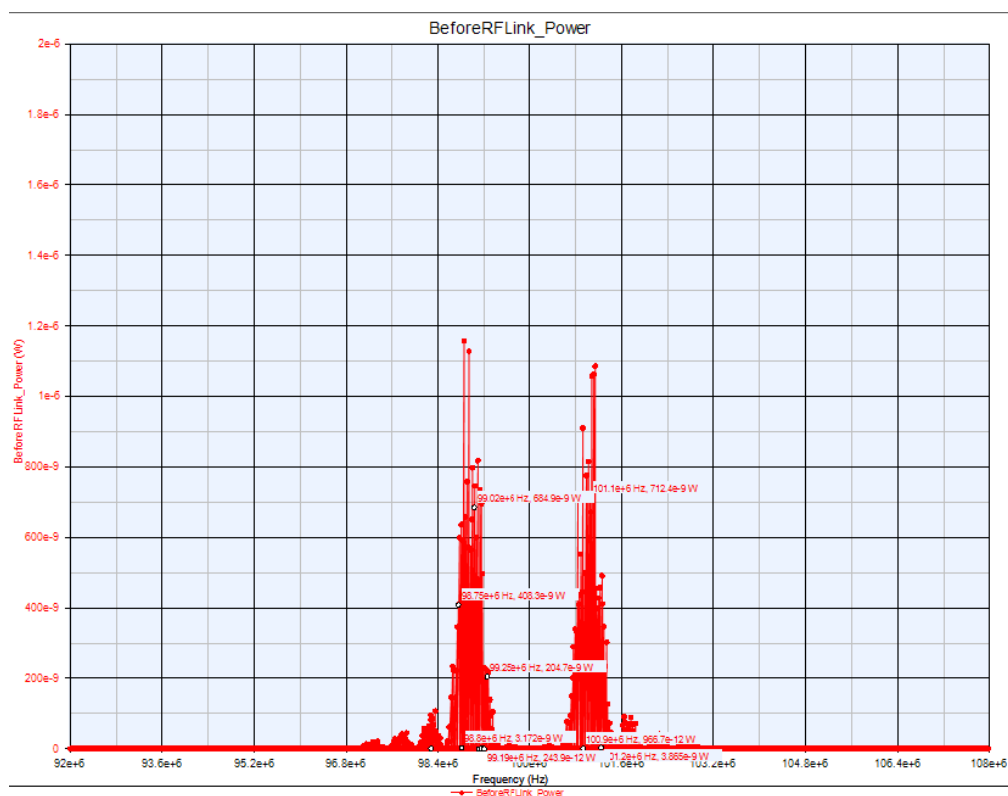


Рис. 9.35. Спектр мощности до RFLink.

13. Теперь давайте рассмотрим спектр мощности после прохождения потока данных по ВЧ каналу (рис. 9.36).

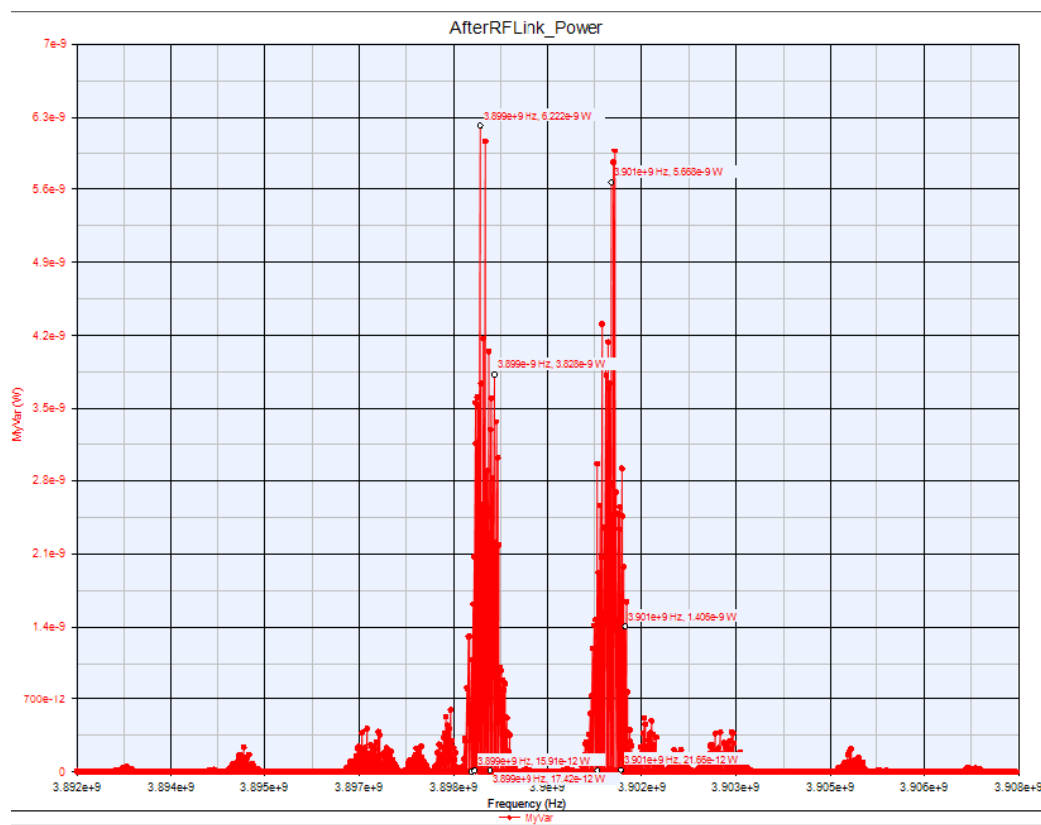


Рис. 9.36. Спектр мощности после RFLink.

Как видите, мощность на частоте 10 МГц, после того, как несущая промодулирована нашими данными, равна примерно $1,18 \times 10^{-6}$ Вт. После прохождения сигнала через ВЧ канал мощность падает примерно до $6,38 \times 10^{-9}$ Вт. Эта потеря мощности имитирует реальные условия распространения сигнала по эфиру между передающей и приёмной антеннами. При этом мощность сигнала может теряться из-за деревьев, зданий и других объектов ландшафта. Кроме того, мы видим боковые полосы. Можно смело утверждать, что они не влияют на данные на стороне приёма, поскольку код Уолша реконструирует информацию, превышающую уровень шумов.

Теперь давайте исследуем отдельные биты и посмотрим, насколько хороша связь.

14. В наборе данных вместо диаграммы откройте табличное представление переданных битов и сравните их с принятыми битами (Rx_bits и Tx_bits), как показано на рис. 9.37.

Если вам нужны пошаговые инструкции, вернитесь к лабораторной работе 3, где мы создавали таблицу для просмотра кодов Уолша.

Index	Rx_hits	Tx_hits
1	0	0
2	1	0
3	1	0
4	0	0
5	1	0
6	0	0
7	1	0
8	0	0
9	0	0
10	1	0
11	0	0
12	1	1
13	0	0
14	0	0
15	0	0
16	0	0
17	0	0
18	0	0
19	0	0
20	0	0
21	0	0
22	0	0
23	0	0
24	0	0
25	0	0
26	0	0
27	1	1
28	0	0
29	0	0
30	0	0

Рис. 9.37. Сравнение переданных и принятых битов.

По принятым битам можно заключить, что система работает хорошо.

9.9 Лабораторная работа 8. Полная система

В этой лабораторной работе мы протестируем характеристики системы, рассчитав коэффициент битовых ошибок (BER).

1. Добавьте новую папку с именем "Step8_CompletedSystem".
2. В папке создайте схему "CompletedSystem" (полная система), создайте набор данных "DF1_CompletedSystem" и анализ потока данных "DF_CompletedSystem" (установите число выборок 4096).
3. Скопируйте компоненты схемы, созданной на этапе 7, и вставьте их в новую пустую схему.

Давайте сопоставим два наших сигнала. Это позволит сравнить сигналы и оценить степень их подобия. В нашей системе CDMA мы сравним отправленный код с принятым кодом, полученным с помощью генератора кодов Уолша. Если коды подобны, то приёмник их декодирует и воспроизведёт исходный переданный сигнал.

4. Установите в свободном месте схемы кросс-коррелятор, для чего введите в селекторе компонентов текст "cross" и выберите компонент кросс-коррелятора.
5. Добавьте два приёмника – один с задержкой (назовите его "Delay_Estimate"), а второй с именем "Cross_Estimate".

6. Добавьте к каждому входу по соединению и измените их имена. Вход Y назовите "Test" (тест), а вход X – "Ref" (опорный). Для этого дважды щёлкните на соединении и измените стандартное имя (рис. 9.38).
7. Соответственно назовите выходное соединение генератора случайных битов "Ref", а выходное соединение системы – "Test" (рис. 9.38).

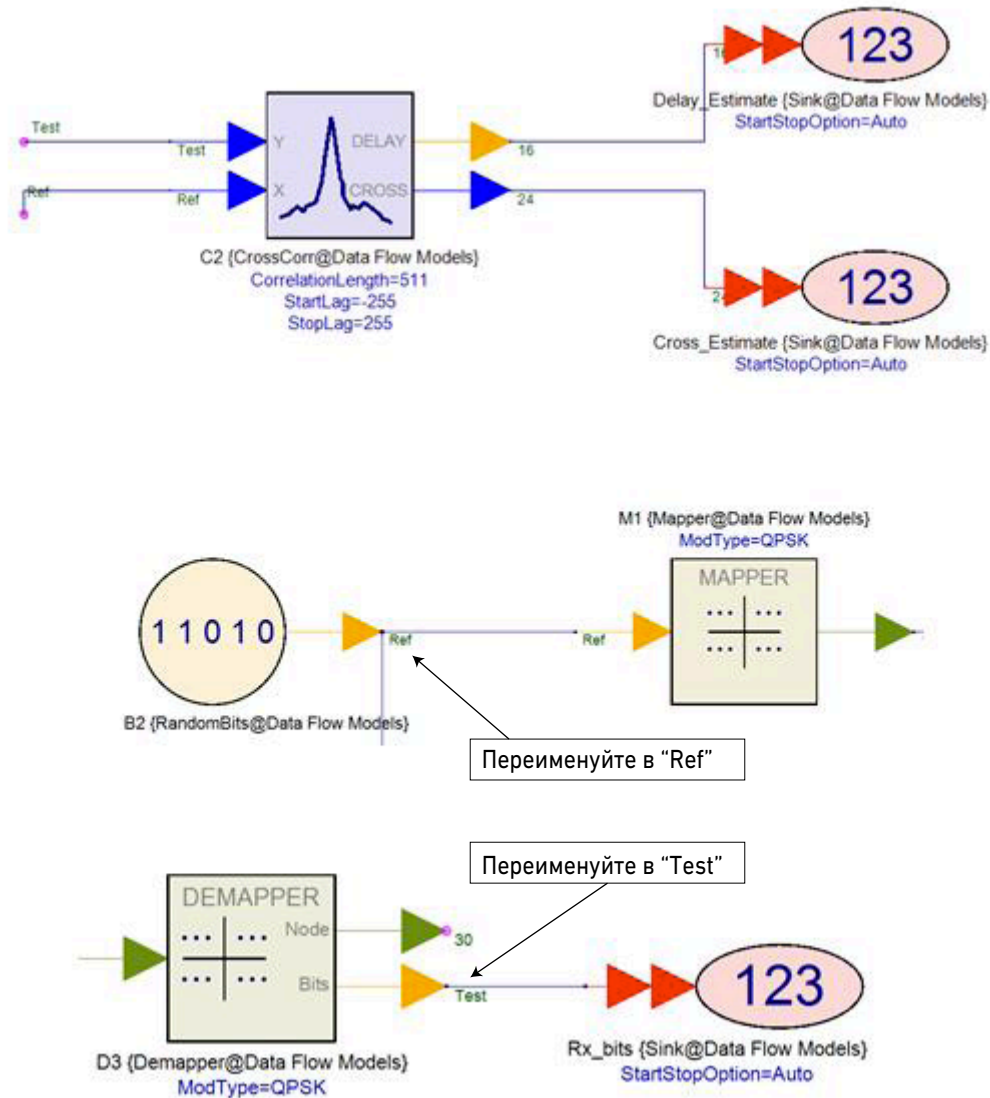


Рис. 9.38. Добавление соединения к каждому входу и изменение их имён.

8. Запустите моделирование.
9. Постройте график задержки, в виде графика кросс-корреляции (Dataset > Delay_Estimate > New Series Graph Wizard > Cross Correlation).

10. Теперь постройте график перекрёстной оценки (рис. 9.39).

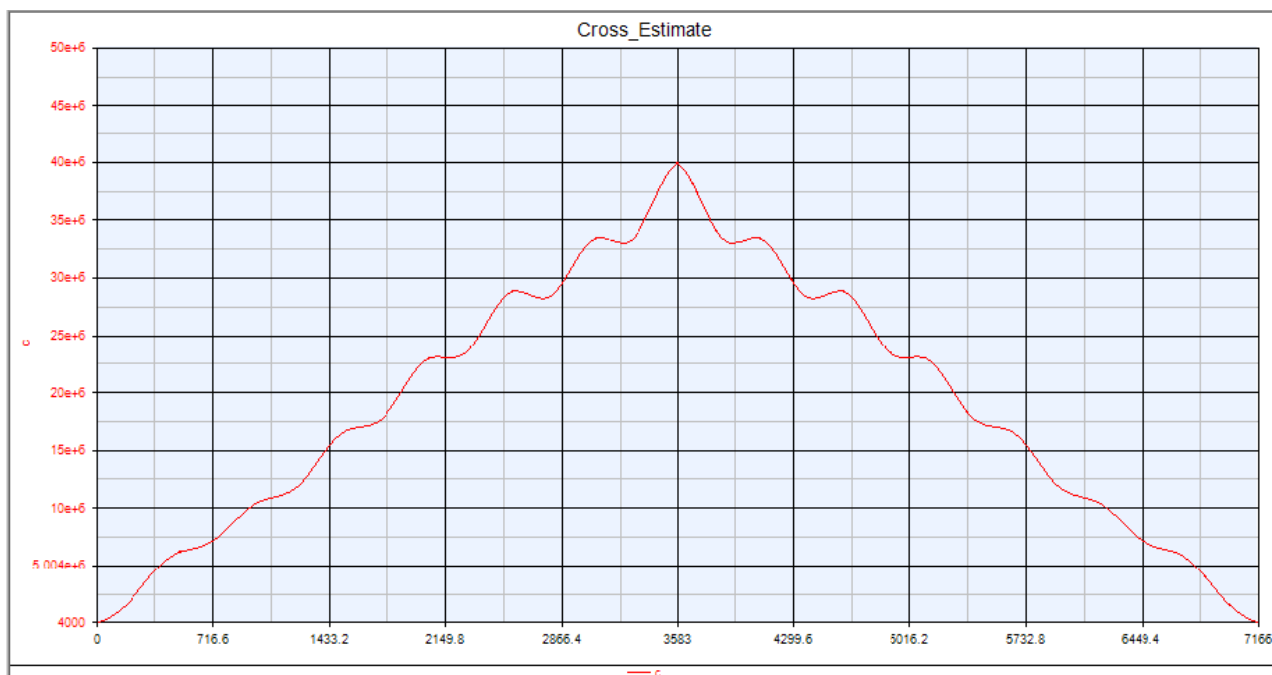


Рис. 9.39. График перекрёстной оценки.

Мы можем представить, что переданный сигнал и сигнал, сохранённый в приёмнике (с помощью генератора кода Уолша), скользят один по другому, и в это время коды становятся все более совпадающими, и пики кросс-корреляции образуют график, напоминающий треугольник. Пик представляет смещение, при котором коды почти идеально совпадают.

Теперь давайте проверим характеристики системы, применив метод BER. BER используется для оценки параметров системы связи, поскольку он показывает интенсивность появления ошибок в системе передачи. BER определяется уравнением Е.9.1:

$$BER = \text{Число ошибочных битов} / \text{Общее число переданных битов} \quad \text{Е.9.1}$$

Если среда между передатчиком и приёмником обладает хорошими характеристиками, то отношение сигнала к шуму будет большим, и BER будет малым. Если система обладает большим уровнем шумов, и методы кодирования выбраны неправильно – в нашем случае схема кодирования Уолша, – то принимаемые биты будут отличаться от исходно переданных битов, что приведёт к росту BER.

1. Найдите в схеме свободное место.
2. Введите в селекторе компонентов “BER” и вставьте в проект соответствующий компонент. Параметр SampleStart модели BER_FER должен быть установлен в 10, BitsPerFrame в 1000, а StartStopOption = Samples.
3. Добавьте по входу REF задержку N=10. Эта задержка обеспечивает синхронизацию битовых потоков, без которой значение BER будет неверным.
4. Добавьте к обоим входам соединения, чтобы мы могли дать им имена. Подключите вход TEST к сигналу TEST. Для этого можно назвать соединение “TEST”. Другому соединению на входе REF дайте имя “REF”. Это позволит измерителю BER рассчитать коэффициент битовых ошибок передаваемого сигнала и принимаемого сигнала, после декодирования его с помощью кода Уолша (рис. 9.40).
5. Запустите моделирование.

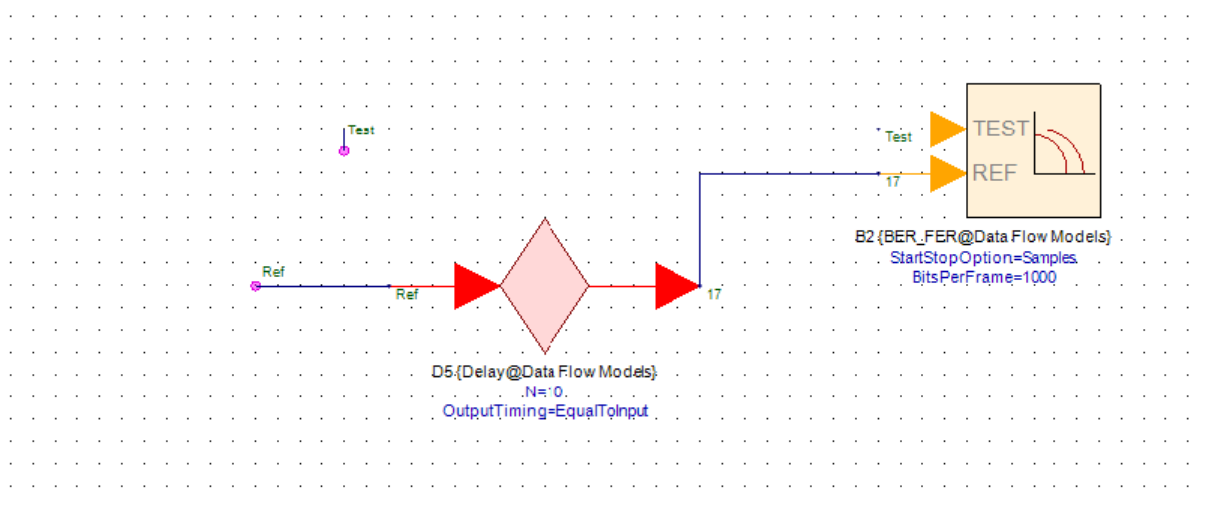


Рис. 9.40. Рассчитайте BER переданного и принятого сигнала с помощью измерителя BER.

Теперь давайте настроим уравнения BER и выведем наши результаты в текстовое окно.

6. Щёлкните в дереве рабочей среды на уравнении 1 (рис. 9.41). В поле кодирования введите следующий текст (B3_BER представляет переменную набора данных BER):

1. using('DF1_CompletedSystem');
2. BER = B3_BER;

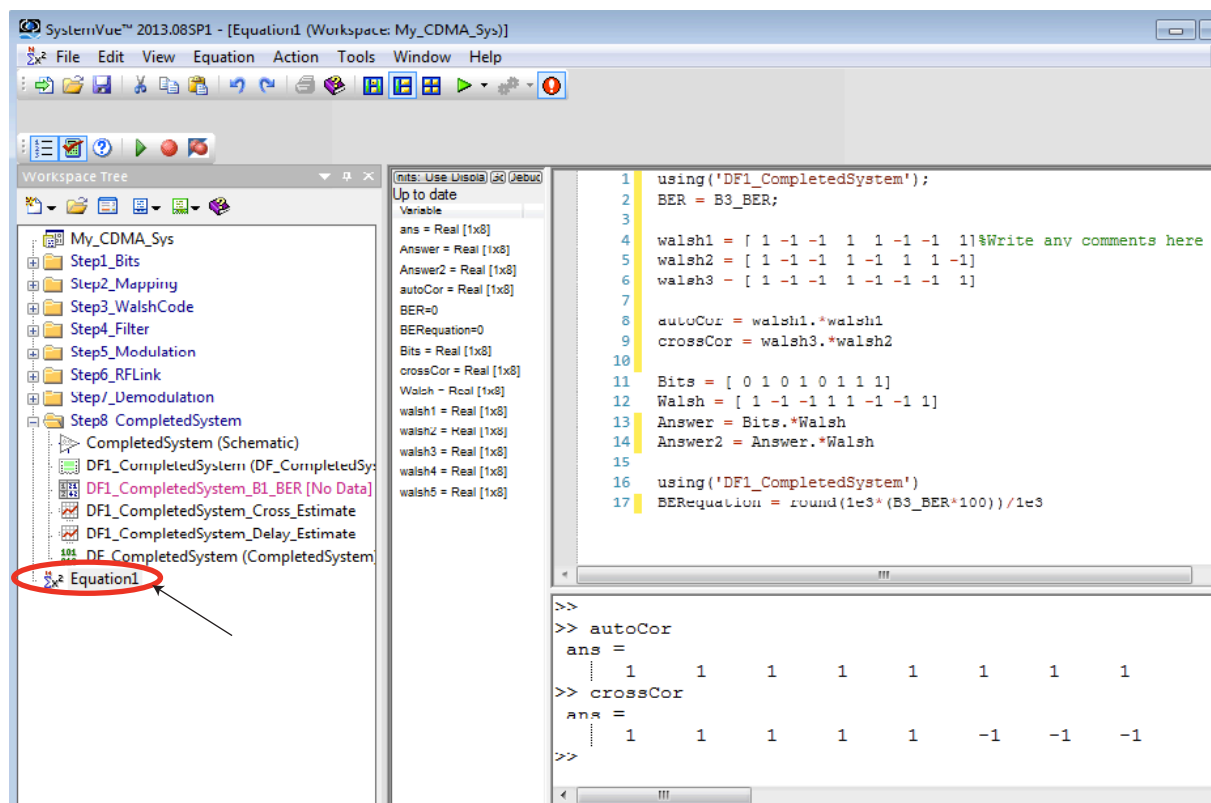


Рис. 9.41. Вставка уравнения BER.

7. Щёлкните на “Show/Hide Annotation Toolbar” (Показать/скрыть панель аннотаций) в верхнем углу приложения SystemVue (рис. 9.42).

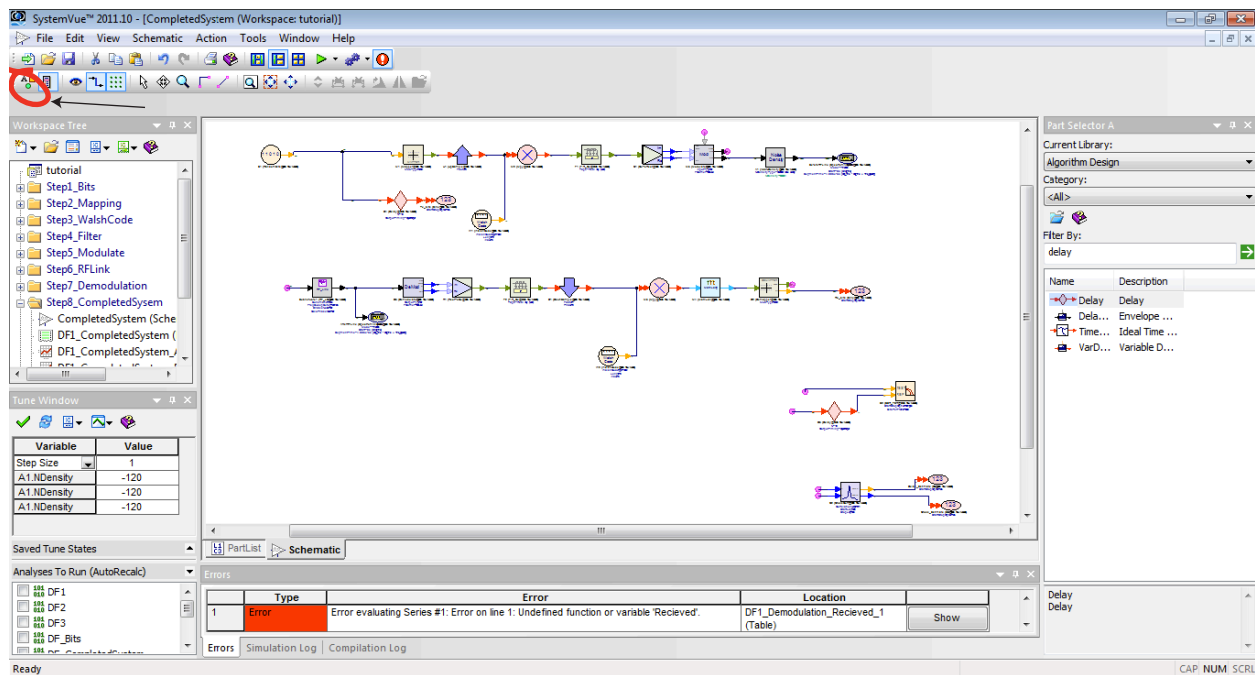


Рис. 9.42. Показать/скрыть панель аннотаций в SystemVue.

Появится панель инструментов. Щёлкните на квадратном текстовом значке, как показано на рис. 9.43.



Рис. 9.43. Квадратный текстовый значок.

Создайте два текстовых поля и расположите их бок о бок. Одно из них отображает наше значение BER, а второе – имя переменной (рис. 9.44).

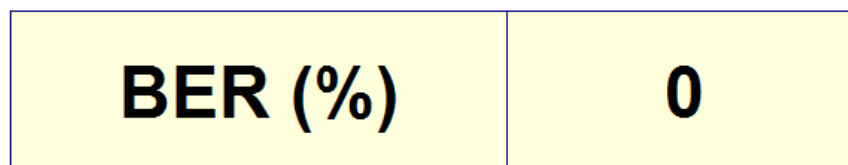


Рис. 9.44. Два расположенных бок о бок текстовых поля.

8. Дважды щёлкните на правом текстовом поле и введите “=BER”, как показано на рис. 9.45.

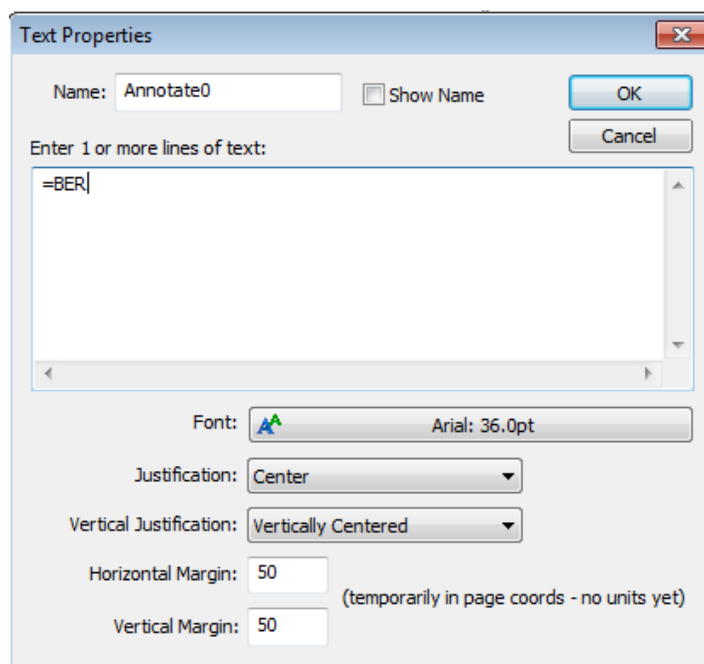


Рис. 9.45. Страница свойств текста.

Здесь вы можете изменить положение текста, его размер, шрифт и цвет.

9. Теперь дважды щёлкните в левом текстовом поле и введите текст “BER (%)”.

Если индекс генератора кода Уолша передатчика совпадает с индексом кода Уолша приёмника, то после щелчка на OK, BER должен равняться 0 %. Это показывает, что характеристики системы идеальны, и все переданные данные успешно декодированы кодом Уолша и приняты.

Давайте изменим индекс одного из кодов Уолша и посмотрим, что произойдёт.

10. Щёлкните на одном из генераторов кода Уолша и измените его индекс на любое значение до 8 (отличное от того, что было установлено).
11. Запустите моделирование.

Обратите внимание, что BER станет больше 0 %. Попробуйте изменить индекс Уолша, ещё раз запустите моделирование и наблюдайте изменения BER.

Мы завершили построение системы CDMA с распределённым спектром и протестировали её характеристики с помощью BER в САПР SystemVue.

В качестве последнего упражнения давайте используем уравнения SystemVue для наблюдения реальных битов, кодов Уолша и альтернативного способа отображения BER. В этом упражнении для обработки выходных данных будут использоваться уравнения на языке описания математических формул.

1. В дереве рабочей среды щёлкните правой кнопкой на “tutorial” (учебник) и выберите Add > Add Equation (Добавить > Добавить уравнение).

Давайте создадим некоторые коды Уолша.

2. В окне кода введите следующий текст, начиная со строки 1 (рис. 9.46).

```
1 walsh1 = [ 1 -1 -1 1 1 -1 -1 1]  %Здесь можно написать комментарии
2 walsh2 = [ 1 -1 -1 1 1 -1 1 -1]
3 walsh3 = [ 1 -1 -1 1 1 -1 -1 1]
```

Рис. 9.46. Знак "%" позволяет записывать комментарии

Обратите внимание, что каждый код Уолша имеет равное число единиц и единиц с минусом.

Обратите внимание, что переменные устанавливаются в левой части дерева.

3. Нажав Go (пуск), скомпилируйте код. Эту кнопку надо нажимать после создания новой переменной.
4. В окне команд (в нижней части экрана) введите любую переменную кода Уолша, например, Walsh1. Заметьте, что команды чувствительны к регистру букв.

```
>> walsh1
ans =
     1     -1     -1     1     1     -1     -1     1
>>
```

Рис. 9.47. Введите переменную кода Уолша, например, Walsh1.

Теперь давайте применим метод корреляции.

5. Примените автокорреляцию к Walsh1, введя в окно кода следующее уравнение
– autoCor = walsh1.*walsh1
6. Выполните кросс-корреляцию walsh3 и walsh2, введя в окно кода следующее уравнение
– crossCor = walsh3.*walsh2

Наблюдайте соответствующие значения в окне команд (рис. 9.41). Обратите внимание, что автокорреляция создаёт один объект из единиц. Здесь мы можем доказать, что автокорреляция, эквивалентная кросс-корреляции с таким же кодом, означает, что данные можно декодировать, и информацию можно извлечь (в связи с полным совпадением). Кросс-корреляция двух разных кодов Уолша даёт рассогласование, поскольку создаётся серия единиц и единиц с минусом. Из-за рассогласования данные декодировать нельзя. Таким образом, мы увидели и доказали, что для декодирования и получения исходных данных, генератор кода Уолша на стороне передатчика должен генерировать такой же код Уолша, что и на стороне приёмника.

Если создать битовую последовательность, то можно увидеть, что происходит с данными, когда они шифруются кодом Уолша (рис. 9.48).

```
Bits = [ 0 1 0 1 0 1 1 1]
Walsh = [ 1 -1 -1 1 1 -1 -1 1]
Answer = Bits.*Walsh
```

Рис. 9.48. Шифрование данных кодом Уолша.

Проверьте результирующий массив с помощью окна команд (рис. 9.49).

```
>> Answer
ans =
     0     -1     -0     1     0     -1     -1     1
>>
```

Рис. 9.49. Результирующий массив.

Если теперь мы возьмём этот код и сопоставим его с нашим кодом Уолша ("Walsh1"), мы сможем получить исходную битовую последовательность (рис. 9.50).

```
Bits = [ 0 1 0 1 0 1 1 1]
Walsh = [ 1 -1 -1 1 1 -1 -1 1]
Answer = Bits.*Walsh
Answer2 = Answer.*Walsh
```

Рис. 9.50. Исходная битовая последовательность.

Проверьте результирующий массив с помощью окна команд (рис. 9.51).

```
>> Answer
ans =
    0    1    0    1    0    1    1    1
>>
```

Рис. 9.51. Результирующий массив.

Используя уравнения, мы можем наблюдать результаты расчёта BER прямо в окне команд с помощью следующего кода:

```
using('DF_ompletedSystem')
BER = round(1e3*(B2_BER*100))/1e3
```

Это даст нам значение BER в процентах.

9.10 Заключение

Надеемся, упражнение по построению системы CDMA с распределённым спектром было полезным, и вы познакомились со всеми необходимыми инструментами для разработки, моделирования, отображения и анализа результатов – графиками, глазковыми диаграммами, спектрами, диаграммами мощности, графиками в частотной области и BER. Эта информация должна подготовить вас к созданию более сложных систем.

Дополнительные навыки работы с САПР SystemVue можно получить с помощью обширной библиотеки примеров, учебных видеороликов и интернет-форумов, доступ к которым предоставляется через меню справки.

Имеются и другие учебные и лабораторные материалы, которые познакомят вас с преимуществами использования SystemVue для проектирования. Посетите страницу EEsOF в интернете или обратитесь к нам для получения пробных версий и лицензий: <http://www.keysight.com/find/eesof>.

9.11 Приложение

Ниже приведена полезная информация и ссылки на ключевые темы, обсуждаемые в этом учебном пособии:

- Рекомендации по применению 1298 компании Keysight. Эти рекомендации по применению содержат введение в концепции цифровой модуляции, обсуждают разные типы цифровой модуляции и рассказывают, как работают цифровые приёмники и передатчики. Их можно найти по ссылке: <http://literature.cdn.keysight.com/litweb/pdf/5965-7160E.pdf>
- Дополнительная информация о SystemVue приведена по ссылкам:
 - Информация о продукте <http://www.keysight.com/find/eesof-systemvue>
 - Конфигурации продукта <http://www.keysight.com/find/eesof-systemvue-configs>
 - Запрос 30-дневной пробной версии <http://www.keysight.com/find/eesof-systemvue-evaluation>
 - Загрузки <http://www.keysight.com/find/eesof-systemvue-latest-downloads>
 - Полезные видео <http://www.keysight.com/find/eesof-systemvue-videos>
 - Форум технической поддержки <http://www.keysight.com/find/eesof-systemvue-forum>

Литература

3. К. Хсу, С. Рамасуббу, М. Ко, Дж. Л. Пино и С. С. Бхаттачария, “Эффективное моделирование графиков критически важных синхронных потоков данных”, материалы конференции по автоматизации проектирования, Сан-Франциско, Калифорния, июль 2006 г.
4. К. Хсу, Дж. Л. Пино, Ф. Ху, “Подход к моделированию и имитации физического уровня LTE, основанный на векторном потоке данных смешанного режима”, материалы конференции по автоматизации проектирования, Анахайм, Калифорния, июнь 2010 г.
5. Джеффри А. Велдон, “Высокопроизводительные КМОП передатчики для радиосвязи”, докторская диссертация, Калифорнийский университет в Беркли, 2005 г.
6. В. У. Лунг, Л. Е. Ларсон и П. Гудем, “Улучшенная архитектура передатчика с цифровой ПЧ для высокоинтегрированных мобильных терминалов W-CDMA”, сс. 1335-1339, 2003 г.
7. Бехзад Разави, “Архитектуры и цепи ВЧ передатчиков”, IEEE CICC, сс. 197-204, 1999 г.
8. Момита Мукхерджи, “Улучшенные полупроводниковые приборы, цепи и системы СВЧ и миллиметрового диапазона”, сс. 495-520, 2010 г.
9. Р. Свитек, С. Раман, “Постоянные смещения в приёмниках прямого преобразования: измерение характеристик и обсуждение”, IEEE Microwave Magazine, сс. 76-86, сентябрь 2005 г.
10. Асад А. Абида, “Трансиверы прямого преобразования для цифровой связи”, IEEE Journal of Solid-State Circuits, том 30, № 12, сс. 1399-1410 1995 г.
11. Кристофер А. ДеВрайс, Ральф Д. Мейсон, “Архитектура с субдискретизацией для маломощных приёмников”, IEEE Transactions on Circuits and Systems, том 55, сс. 304-308, 2008 г.
12. Дж. Х. Миккелсен, “Архитектуры входных цепей для КМОП радиоприёмников”, факультет телекоммуникаций, университет Ольборга, сс. 1-18, 1996 г.
13. *Основы измерения коэффициента шума в ВЧ и СВЧ диапазоне*, рекомендации по применению 57-1, Keysight Technologies, 2010 г.
14. Бернард Склар, “Цифровая связь: основы и применение”, второе издание, Prentice Hall PTR, 2001 г.
15. TRANZEO Wireless Technologies, “Анализ бюджета радиоканала: как рассчитать бюджет канала для радиосети”, официальный документ, 2010 г.
16. Intersil, “Учебник по основам анализа бюджета канала”, рекомендации по применению, AN9804.1, июнь 1998 г.
17. А. Хаджимири, Т. Х. Ли, “Общая теория фазового шума электрических генераторов”, IEEE Journal of Solid-State Circuits, том 33, № 2, февраль 1998 г., сс. 178-194.
18. Джон Пол, “Освобождение спектра и управление помехами”, сайт Международного союза электросвязи, 2006 г.
19. Куиженг Гу, “Проектирование ВЧ трансиверов для радиосвязи”, Springer, 2005 г.
20. Салваторе, Рагуса с соавторами, “Анализ относительного уровня мощности в соседнем канале для сигналов OFDM”, IEEE International Symposium on Signal Processing and Information Technology, 2005 г.

21. Матт Лой, "Понимание и улучшение чувствительности приёмников для беспроводных приложений", отдел радиосвязи, Texas Instruments, 1999 г.
22. С. Маас, "Нелинейные цепи ВЧ и СВЧ диапазона", второе издание, Artech House, 2003 г.
23. Г. Брид, "Базовые принципы электрически малых антенн", High Frequency Electronics Magazine, сс. 50-53, 2007 г.
24. А. Хурфар, В. Джемниджад, "Электромагнитное моделирование и анализ радиоантенн", IEEE Microwave Magazine, сс. 51-67, 2003 г.
25. Д. М. Позар, "Микрополосковые антенны", Proceedings of the IEEE, том 8, № 01, сс. 79-91, 1992 г.
26. Ф. Х. Рааб с соавторами, "Усилители мощности и передатчики ВЧ и СВЧ диапазона – части 1-4", High Frequency Electronics, 2003 г.
27. Г. Гонзалес, "Транзисторные усилители СВЧ диапазона: анализ и проектирование", Prentice Hall, второе издание, 1996 г.
28. С. Уиндер, "Проектирование аналоговых и цифровых фильтров", второе издание, Newnes, 2002 г.
29. Р. У. Реа, "Проектирование и компьютерное моделирование ВЧ фильтров", Noble Publishing, 1994 г.
30. Д. М. Позар, "Проектирование в диапазоне СВЧ", третье издание, John Wiley & Sons, 2005 г.
31. С. Маас, "Смесители СВЧ диапазона", второе издание, Artech House, 1993 г.
32. А. Гребенников, "Проектирование транзисторных генераторов ВЧ и СВЧ диапазона", John Wiley & Sons, 2007 г.
33. М. Одинец, "Проектирование генераторов ВЧ и СВЧ диапазона", Artech House, 2002 г.
34. И. Дж. Бал, "Широкополосные детекторы мощности", IEEE Micro Wave Magazine, сс. 82-86, 2007 г.
35. *Моделирование диодного детектора с помощью САПР ADS компании Keysight*, рекомендации по применению 1156, Keysight Technologies, 2007 г.
36. П. Е. Аллен, Д. Р. Холберг, "Проектирование аналоговых КМОП цепей", второе издание, Oxford University Press, 2002 г.
37. Д. Кестер, "Справочник по преобразованию данных", Newnes, 2005 г.
38. *Измерение параметров передатчиков и приёмников GSM/EDGE для базовых трансиверных станций и их компонентов*, рекомендации по применению 1312, Keysight Technologies.
39. К. Кундерт, "Принципы проектирования сверху вниз систем со смешанными сигналами", Designer's Guide Consulting Inc., 2006 г., сс. 1-31.
40. Г. Гилен, Р. Рутенбар, "Автоматизированное проектирование аналоговых и смешанных интегральных схем", IEEE Proceedings, том 88, 2000 г., сс. 1825-1852.
41. Фреверт с соавторами, "Моделирование и имитация для проектирования ВЧ систем", Springer, 2005 г.
42. Дж. Е. Чен, "Моделирование ВЧ систем", Designer's Guide Consulting Inc., 2000 г., сс. 1-41.

43. Партнёрский проект третьего поколения – Технические характеристики сети группового радиодоступа, “Наземный радиодоступ Evolved UMTS: радиопередача и приём абонентского оборудования (UE)”, выпуск 11, март 2012 г.
44. Партнёрский проект третьего поколения – Технические характеристики сети группового радиодоступа, “Наземный радиодоступ Evolved UMTS: радиопередача и приём базовых станций (BS)”, выпуск 11, март 2012 г.
45. *LTE и эволюция до 4G – Проблемы проектирования и измерения*, публикация Keysight Technologies, 2009 г.
46. Х. Холма, А. Тоскала, “LTE для UMTS – радиодоступ на основе OFDMA и SC-FDMA”, John Wiley & Sons, 2009 г.
47. С. Сесиа, И. Туфик, М. Бакер, “LTE – долговременное развитие UMTS: от теории к практике”, John Wiley & Sons, 2009 г.
48. Motorola Inc., “Долговременное развитие (LTE): обзор радиointерфейса LTE”, официальный документ, 2007 г.
49. Л. Тин, У. Ксу, “Входные цепи LTE сканируют три диапазона”, <http://mwrf.com/Article/ArticleID/23956/23956.html>, Microwaves & RF.
50. *Тестирование и диагностика цифровых ВЧ приёмников*, рекомендации по применению 1314, Keysight Technologies.
51. *Цифровая модуляция в системах связи – Введение, глава 7*, рекомендации по применению 1298, Keysight Technologies.
52. *Тестирование оптических систем следующего поколения с помощью имитируемых сигналов OFDM*, рекомендации по применению 5990-9972EN, Keysight Technologies.

*X-параметры являются товарным знаком и зарегистрированным товарным знаком компании Keysight Technologies в США, ЕС, Тихоокеанском регионе и других странах. Формат X-параметров и лежащие в их основе уравнения открыты и документированы. Дополнительная информация приведена по ссылке <http://www.keysight.com/find/eesof-x-parameters-info>.

WiMAX, Mobile WiMAX, WiMAX Forum и логотип WiMAX Forum, WiMAX Forum Certified и логотип WiMAX Forum Certified являются товарными знаками форума WiMAX в США.

myKeysight

myKeysight

www.keysight.com/find/mykeysight

Персонализированное представление интересующей вас информации.

www.keysight.com/find/eesof-systemvue

Российское отделение

Keysight Technologies

115054, Москва, Космодамианская
наб., 52, стр. 3

Тел.: +7 (495) 7973954

8 800 500 9286 (Звонок по России
бесплатный)

Факс: +7 (495) 7973902

e-mail: tmo_russia@keysight.com

www.keysight.ru

Сервисный Центр

Keysight Technologies в России

115054, Москва, Космодамианская
наб., 52, стр. 3

Тел.: +7 (495) 7973930

Факс: +7 (495) 7973901

e-mail: tmo_russia@keysight.com

Технические характеристики и описания
продуктов могут изменяться без предва-
рительного уведомления.